Research Article

Development of Deep Convolutional Neural Network with Adaptive Batch Normalization Algorithm for Bearing Fault Diagnosis

Chao Fu,1 Qing Lv,1 and Hsiung-Cheng Lin2

1Hebei Normal University, Shijiazhuang 050024, China
2National Chin-Yi University of Technology, Taichung 41170, Taiwan

Correspondence should be addressed to Hsiung-Cheng Lin; hclin@ncut.edu.tw

Received 29 May 2020; Revised 9 August 2020; Accepted 5 September 2020; Published 18 September 2020

Copyright © 2020 Chao Fu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

It is crucial to carry out the fault diagnosis of rotating machinery by extracting the features that contain fault information. Many previous works using a deep convolutional neural network (CNN) have achieved excellent performance in finding fault information from feature extraction of detected signals. However, they may suffer from time-consuming and low versatility. In this paper, a CNN integrated with the adaptive batch normalization (ABN) algorithm (ABN-CNN) is developed to avoid high computing resource requirements of such complex networks. It uses a large-scale convolution kernel at the grassroots level and a multidimensional $3 \times 1$ small convolution nuclear. Therefore, a fast convergence and high recognition accuracy under noise and load variation environment can be achieved for bearing fault diagnosis. The performance results verify that the proposed model is superior to Support Vector Machine with Fast Fourier Transform (FFT-SVM) and Multilayer Perceptron with Fast Fourier Transform (FFT-MLP) models and Deep Neural Network with Fast Fourier Transform (FFT-DNN).

1. Introduction

The characteristics of vibration values are a crucial issue in a designed machine. A lack of balance of its rotating parts and elements can result in noncoaxiality of shafts [1]. Noise and vibrations can even occur in complex structures of ships and result in noticeable obstacle in ship construction [2]. In such a phenomenon, the vibration measurement can help identify dynamic state changes in a rotational system [3]. Accordingly, it is known that fault diagnosis of bearings is based on vibration signals, and a set of features is extracted to classify for rotating shaft fault prediction. The key to the success of fault diagnosis is determined by the choice of feature extractor and classifier [4].

The current research on intelligent algorithms for either diagnosis or prediction has been attracting more attention in past years. However, the feature extractor and classifier of neural networks may not suit the load change for maintaining a highly accurate outcome [5–8]. For this reason, a combination of different neural network methods attempted to achieve a highly possible solution in a variety of disciplines. For example, trademark image retrieval was implemented using a combination of deep CNNs. A pretrained process and the trademark image retrieval task were carried out by fine tuning the network using two different databases [9]. Another case is that recurrent neural network (RNN) was used to learn inner spectral correlations, while CNN was focused on saliency features and spatial relevance [10]. Recently, a hybrid pattern recognition method was proposed for short-term wind power forecasting. The time series of produced power is estimated through combination of preprocessing, feature selection, and regression steps [11].

To solve problems of fault diagnosis of bearings, the most ideal way may combine two aspects of feature extraction and classification into one model without loss of information. The convolutional neural network (CNN) algorithm has the characteristic of “end-to-end.” It means that it can complete the whole process of feature extraction, feature dimension...
reduction, and classifier classification through a neural network process [12–14]. The CNN is a multilayer neural network, including a filtering stage and a classification stage [15–18]. The filtering stage is used to extract the characteristics of the input signal, and the classification level classifies the learned features, where the two-level network parameters can be obtained through a joint training.

In recent years, two-dimensional convolutional neural networks containing stacked 3 × 3 convolution kernels, e.g., VGGNet, ResNet, and Google’s Inception V4, have been reported [19–21]. This type of CNN model can deepen the network learning [22]. At the same time, it can achieve a larger receptive field with fewer parameters, thereby suppressing overfitting. However, for a one-dimensional vibration signal, the structure of the two-tiered 3 × 1 convolution, at the expense of six weights, only acquired the receptive field of 5 × 1 so that its application is limited.

2. The Proposed Model

2.1. WFK-CNN Algorithm. The model of the Deep Convolutional Neural Networks with Wide First-layer Kernel (WFK-CNN) is characterized by large-scale convolution kernels at the grassroots level, followed by 3 × 1 small convolution kernels in the convolution layer. The convolution kernel size of all the convolution layers except the primary layer was set as 3 × 1. After each convolution operation, batch normalization (BN) is performed, and then maximum pooling of 2 × 1 is carried out. The structure of WFK-CNN is shown in Figure 1.

2.2. Batch Normalized (BN) Algorithm. The BN algorithm can reduce the internal covariate transfer, improve the training efficiency of the network, and enhance the generalization ability of the network [19–21]. In this paper, a batch normalized layer is added between the convolutional layer and the active layer and between the full-connected layer and the active layer. Its main operation step is to subtract the mean of the Mini-Batch from the input signal of the convolutional layer or the fully connected layer before dividing it by its standard deviation so that the training process can be accelerated. However, this may limit the input value to a narrower range, reducing the network implementation ability. Therefore, the value of normalization is multiplied by a scaling amount γ and an offset β for enhanced expression. When BN acts on the fully connected layer, the input of the lth BN layer be \( y^{(l)} \), and the batch normalization operation is expressed in equations (1)–(4):

\[
\bar{y}^{(l)} = \frac{y^{(l)} - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \tag{1}
\]

\[
z^{(l)} = y^{(l)} \bar{y}^{(l)} + \beta^{(l)} \tag{2}
\]

\[
\mu_B = E_B[y^{(l)}] = \frac{1}{m} \sum_{k=1}^{m} y_k^{(l)} \tag{3}
\]

\[
\sigma_B^2 = \text{Var}[y^{(l)}] = \frac{1}{m} \sum_{k=1}^{m} (y_k^{(l)} - \mu_B)^2 \tag{4}
\]

where \( y^{(l)} \) is the input of the lth BN layer, \( \bar{y}^{(l)} \) and \( \beta^{(l)} \) are BN layer scaling and offset, \( z^{(l)} \) is BN layer output, and \( \epsilon \) is the numerical stability constant term.

When BN acts on the convolutional layer, the batch normalization operation is represented as equations (5)–(8):

\[
\bar{y}^{(l,i,j)} = \frac{y^{(l,i,j)} - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} \tag{5}
\]

\[
z^{(l,i,j)} = y^{(l,i,j)} \bar{y}^{(l,i,j)} + \beta^{(l)} \tag{6}
\]

\[
\mu_B = E_B[y^{(l,i,j)}] = \frac{1}{m_p} \sum_{k=1}^{m} \sum_{j=1}^{p} y_k^{(l,i,j)} \tag{7}
\]

\[
\sigma_B^2 = \text{Var}[y^{(l,i,j)}] = \frac{1}{m_p} \sum_{k=1}^{m} \sum_{j=1}^{p} (y_k^{(l,i,j)} - \mu_B)^2 \tag{8}
\]

From the literature by Guo et al., 2018, \( \mu_B \) and \( \sigma_B^2 \) of each Mini-Batch are used to solve the differences of \( \mu_{\text{train}} \) and \( \sigma_{\text{train}}^2 \) of the entire training sample by unbiased estimation. This paper directly solves \( \mu_{\text{train}} \) and \( \sigma_{\text{train}}^2 \) of the entire training sample. Replace \( \mu_{\text{train}} \) and \( \sigma_{\text{train}}^2 \) with \( \mu_B \) and \( \sigma_B^2 \) during the test.

When BN acts on a fully connected layer, the loss function \( L \) on the neurons of the BN layer and the derivative of the hyper parameter are shown in the following equation:

\[
\frac{\partial L}{\partial y^{(l)}} = \frac{\partial L}{\partial \bar{y}^{(l)}} \frac{\partial \bar{y}^{(l)}}{\partial y^{(l)}} = \frac{\partial L}{\partial \bar{y}^{(l)}} \frac{1}{\sqrt{\sigma_B^2 + \epsilon}}
\]

\[
\frac{\partial L}{\partial \mu_B} = \frac{1}{m} \frac{\partial L}{\partial \bar{y}^{(l)}} \frac{\partial \bar{y}^{(l)}}{\partial \mu_B} + \frac{1}{m} \frac{\partial L}{\partial \sigma_B} \frac{\partial \sigma_B}{\partial \mu_B} + \frac{1}{m} \frac{\partial L}{\partial \mu_B}
\]

\[
\frac{\partial L}{\partial \mu_B} = \frac{1}{m} \frac{\partial L}{\partial \bar{y}^{(l)}} \frac{1}{\sqrt{\sigma_B^2 + \epsilon}} + \frac{\partial L}{\partial \sigma_B} \frac{\partial \sigma_B}{\partial \mu_B} + \frac{\partial L}{\partial \mu_B}
\]

\[
\frac{\partial L}{\partial \sigma_B} = \frac{1}{m} \frac{\partial L}{\partial \bar{y}^{(l)}} \frac{1}{\sqrt{\sigma_B^2 + \epsilon}} + \frac{\partial L}{\partial \mu_B} \frac{\partial \mu_B}{\partial \sigma_B} + \frac{\partial L}{\partial \sigma_B}
\]

When BN acts on the convolution layer, the loss function \( L \) on the neurons of the BN layer and the derivative of the hyper parameter are shown in the following equation:
2.3. WFK-CNN Parameter Standard. The size, step size, and the number of convolutional layers in the WFK-CNN model are chosen for the base convolution kernel. The core of the convolutional neural network is the receptive field, which is the range of perception of a neuron in its underlying network [23–27]. The profile of neuronal receptive field is shown in Figure 2.

For the WFK-CNN filtering stage to learn displacement-independent features, the size of the receptive field in the input signal of the last neuron in the pooling layer should be greater than one cycle. Suppose that the size of the receptive field at the last input neuron in the pooled layer is \( R^{(0)} \), \( T \) is the number of data points recorded by using the accelerometer of the rotation of the bearing, and \( L \) is the length of the input signal. Then, \( T \leq R^{(0)} \leq L \) can be used as a criterion for WFK-CNN parameter selection.

First, solve the relationship between the receptive field of the \( l \)th pooling layer and the receptive field at the \( l-1 \)th pooling layer, as shown in the following equation:

\[
R^{(l-1)} = S^{(l)} \left( P^{(l)} R^{(l)} - 1 \right) + W^{(l)},
\]

where \( S^{(l)} \) and \( W^{(l)} \) are the step size and convolution kernel scale for the first convolutional layer and \( P^{(l)} \) is the number of downsampling points for the first pooling layer.

When \( l > 1 \), \( S^{(1)} = 1, W^{(1)} = 3, \) and \( P^{(1)} = 2 \), equation (11) can be simplified to the following equation:

\[
R^{(l-1)} = 2R^{(l)} + 2.
\]

When \( l = n \) and \( R^{(n)} = 1 \), the size of the receptive field of the first pooled layer neuron in the first pooling layer is shown in the following equation:

\[
R^{(1)} = 2^{n-1} \times 3 - 2,
\]

where \( n \) is the number of convolutional layers.

Substituting equation (13) into equation (11), the receptive field \( R^{(0)} \) on the input last neuron of the pooled layer is obtained in the following equation:

\[
R^{R^{(0)}} = S^{(1)} \left( P^{(1)} R^{(1)} - 1 \right) + W^{(1)}
\]

\[
= 2S^{(1)} \left( 2^{n-1} \times 3 - 2 \right) + W^{(1)} - S^{(1)} = S^{(1)} \left( 2^n \times 3 - 4 \right).
\]

According to the design rule \( T \leq R^{(0)} \leq L \), the final design criteria can be determined as
\[ T \leq S^{(1)}(2^n \times 3 - 4) \leq L, \]

\[ S^{(1)} = L. \]  

The input signal length \( L = 2048 \), and the signal period \( T \approx 400 \). If there are 5 layers of convolution, then the convolutional layer of the grassroots convolution can only have 8 or 16 convolution steps. The number of neurons in the network will increase when the number of convolution kernels, the scale, and the number of network layers, or the step size of the convolution kernel, are reduced, which can increase the expressiveness of the network but at risk of overfitting. Other hyperparameters of the network need to be further adjusted in the experiment according to the amount of training data [28–34].

The parameters of WFK-CNN model used in this study are shown in Table 1. It has five convolution and pooling layers. The size of the convolution kernel at the base level is \( 64 \times 1 \), and the other convolution kernels are \( 3 \times 1 \). The number of neurons in the hidden layer is 100, and there are 10 outputs in the layer softmax, corresponding to 10 kinds of bearing states.

2.4. ABN-CNN Model. The ABN-CNN model is based on the integration of WFK-CNN algorithm and Adaptive Batch Normalization (Ad-BN) algorithm. Therefore, the domain adaptation ability can be effectively improved when the distribution of test samples is different from the distribution of training samples. The ABN-CNN model is briefly described in Table 2.

The flowchart of ABN-CNN algorithm is shown in Figure 3. The training samples are used to train the WFK-CNN model until the training process is completed. If the distribution of the training signal and the test signal is not consistent, the test set is input to the WFK-CNN model, where only the data are forward propagating, and the mean variance of all BN layers is replaced by the mean variance of the test set, but other network parameters remain unchanged.

3. Performance Results

In this study, the data from the CWRU Rolling Bearing Data Center, which provides a world-recognized standard data set, were used for bearing fault diagnosis of the proposed model as well as the comparison with existing algorithms [35]. This ball bearing test data came from experiments conducted using a 2 hp Reliance Electric Motor. Accordingly, the actual test conditions of the motor can sufficiently support the proposed model for industrial applications.

3.1. Performance Analysis of Bearing Fault Diagnosis in Noise Environment. In this section, the anti-noise capability of ABN-CNN algorithm is analyzed. First, the noise signal is added to test samples to simulate noise pollution in industrial environment. The noise of the diagnostic signal is generally additive Gauss white noise. Therefore, the test signals contain different levels of additive white Gaussian noise.

Signal-to-noise ratio (SNR) is defined as the ratio of \( P_{signal} \) and \( P_{noise} \), represents the energy of signal and noise, respectively:

\[ \text{SNR}_{\text{dB}} = 10 \log_{10} \left( \frac{P_{signal}}{P_{noise}} \right). \]  

As shown in Figure 4, from top to bottom, the failure signal of the bearing from the inner ring without adding noise, the additive Gauss white noise signal, and the inner ring fault signal with noise are presented, respectively, where \( \text{SNR} = 0 \text{ dB} \). It can be seen that it is difficult to directly extract valid fault information from noise-containing signals.

3.2. Model Training and Testing. Based on the recognition rate, the training convergent curves with/without BN in the WFK-CNN model are shown in Figure 5. The curves of target function value with/without BN using logarithm is shown in Figure 6. The size of each Mini-Batch is set as 256, and the learning rate of Adam algorithm is set as 0.001.

3.3. The Influence of the Convolution Kernel Size on the Noise. The recognition rate of the influence of convolution kernel size on noise is studied in the WFK-CNN model and ABN-CNN model. In the test, the SNR is set to −4 dB to 10 dB, and the convolution kernel size ranges from 16 to 128.
As shown in Table 3, the WFK-CNN model can achieve a higher recognition rate in general when the convolution kernel is large in a noisy environment. For example, when the size of convolution kernel is 112 × 1, the recognition rate can reach more than 90% when the SNR is 0 dB, and when the convolution kernel size is 16 × 1, the recognition rate is as low as 55.46%. However, when the convolution kernel size is larger than 128 × 1, the recognition rate may decrease slightly. For example, the recognition rate of the convolution kernel 128 × 1 is 89.65%, which is slightly lower than the 89.93% of the 96 × 1 convolution kernel. In Table 4, when the SNR is >−4 dB, the recognition rate of the ABN-CNN model
is generally above 90%. Compared with the WFK-CNN model, the recognition rate is significantly improved, especially at lower SNR and smaller convolution kernel size. For example, when SNR = −4 dB, the WFK-CNN model with 64 × 1 convolution kernel size has only 51.89%, but the ABN-CNN model reaches up to 92.56%. As mentioned above, please note that the convolution kernel size influences the recognition rate considerably, particularly under small SNR situation. Accordingly, the large-scale convolution kernel should be appropriately selected when the noise is large, especially in WFK-CNN model.

3.4. Anti-Noise Comparison. The anti-noise comparison between ABN-CNN, WFK-CNN, FFT-SVM, FFT-MLP, and FFT-DNN models is shown in Figure 7. SVM uses a radial basis kernel function, and the number of neurons per layer of the FFT-DNN model is selected as 1025, 500, 200, 100, and 10, respectively. The convolution kernel size of the ABN-CNN and WFK-CNN models is set to 112 × 1.

It can be seen that the recognition rate of FFT-DNN is higher than that of DNN-MLP, but the immunity of the two is relatively weak. For example, at SNR = 4 dB, the recognition rate of both models is less than 90%. In addition, the WFK-CNN model has higher noise and interference immunity than the FFT-DNN and DNN-MLP models, but it is slightly lower than the FFT-SVM. On the other hand, in various noise environments, the recognition rate of the ABN-CNN model reaches more than 90%, which is the strongest anti-noise ability of all models.

3.5. Fault Diagnosis Performance Analysis under Variable Load Condition. Figure 8 shows diagnostic signals with 0.014 inch normalized inner circle defects under different loads. As can be seen, the number of features in the vibration signal varies with different loads. In addition to the inconsistent amplitude, the period and phase of the fluctuations are also very different. This phenomenon makes it difficult to correctly classify the extracted features for fault recognition.

The proposed WFK-CNN (Ad-BN) model trained by different kinds of load data using 1 hp, 2 hp, and 3 hp motors has a great practical significance to diagnose the vibration signal particularly when the load changes. The detailed description of variable loads for training and testing is shown in Table 5. The data, e.g., A set, were set for training,
Table 3: Recognition rate of the WFK-CNN model under noise environment.

| Convolution kernel size | −4 (%) | −2 (%) | 0 (%) | 2 (%) | 4 (%) | 6 (%) | 8 (%) | 10 (%) |
|-------------------------|--------|--------|-------|-------|-------|-------|-------|--------|
| 16                      | 27.24  | 40.65  | 55.46 | 72.11 | 85.61 | 94.62 | 98.53 | 99.25  |
| 24                      | 35.15  | 52.84  | 70.32 | 84.83 | 94.32 | 98.53 | 99.66 | 99.85  |
| 32                      | 42.32  | 57.77  | 72.71 | 86.57 | 95.39 | 98.46 | 99.42 | 99.61  |
| 40                      | 46.73  | 63.09  | 77.60 | 90.16 | 97.03 | 99.18 | 99.61 | 99.86  |
| 48                      | 50.22  | 66.32  | 80.34 | 92.15 | 97.75 | 99.36 | 99.78 | 99.89  |
| 56                      | 51.78  | 67.75  | 80.79 | 92.32 | 97.80 | 99.19 | 99.72 | 99.72  |
| 64                      | 51.89  | 67.21  | 82.08 | 93.05 | 98.10 | 99.37 | 99.76 | 99.83  |
| 72                      | 53.83  | 68.67  | 82.41 | 92.95 | 97.88 | 99.44 | 99.69 | 99.80  |
| 80                      | 56.17  | 69.43  | 84.37 | 94.79 | 98.74 | 99.49 | 99.87 | 99.85  |
| 88                      | 56.13  | 71.78  | 85.43 | 95.09 | 98.42 | 99.33 | 99.79 | 99.84  |
| 96                      | 64.43  | 78.87  | 89.93 | 96.90 | 99.06 | 99.55 | 99.85 | 99.86  |
| 104                     | 62.77  | 79.31  | 90.44 | 97.68 | 99.36 | 99.69 | 99.84 | 99.83  |
| 112                     | 66.97  | 80.92  | 90.62 | 97.09 | 98.79 | 99.51 | 99.82 | 99.80  |
| 120                     | 61.66  | 77.70  | 90.37 | 97.30 | 99.05 | 99.60 | 99.80 | 99.85  |
| 128                     | 60.73  | 77.43  | 89.65 | 97.17 | 99.21 | 99.51 | 99.84 | 99.82  |
| The average             | 66.97  | 80.92  | 90.62 | 97.68 | 99.36 | 99.69 | 99.87 | 99.89  |

Table 4: Recognition rate of the ABN-CNN model under noise environment.

| Convolution kernel size | −4 (%) | −2 (%) | 0 (%) | 2 (%) | 4 (%) | 6 (%) | 8 (%) | 10 (%) |
|-------------------------|--------|--------|-------|-------|-------|-------|-------|--------|
| 16                      | 81.87  | 90.34  | 95.57 | 98.45 | 99.04 | 99.45 | 99.77 | 99.74  |
| 24                      | 87.21  | 93.96  | 97.28 | 99.08 | 99.66 | 99.82 | 99.85 | 99.90  |
| 32                      | 89.83  | 95.16  | 97.96 | 99.23 | 99.52 | 99.65 | 99.73 | 99.88  |
| 40                      | 90.92  | 95.97  | 98.25 | 99.33 | 99.54 | 99.77 | 99.80 | 99.90  |
| 48                      | 91.69  | 96.39  | 98.43 | 99.47 | 99.70 | 99.81 | 99.83 | 99.85  |
| 56                      | 92.68  | 96.52  | 98.65 | 99.37 | 99.76 | 99.73 | 99.85 | 99.88  |
| 64                      | 92.56  | 96.83  | 98.81 | 99.56 | 99.65 | 99.87 | 99.88 | 99.95  |
| 72                      | 92.36  | 96.33  | 98.55 | 99.23 | 99.60 | 99.79 | 99.80 | 99.87  |
| 80                      | 92.35  | 96.80  | 98.63 | 99.44 | 99.61 | 99.76 | 99.89 | 99.83  |
| 88                      | 92.63  | 97.06  | 98.79 | 99.52 | 99.63 | 99.73 | 99.87 | 99.80  |
| 96                      | 92.65  | 97.10  | 98.79 | 99.61 | 99.62 | 99.82 | 99.84 | 99.81  |
| 104                     | 92.45  | 96.59  | 98.59 | 99.53 | 99.62 | 99.75 | 99.80 | 99.90  |
| 112                     | 91.84  | 96.21  | 98.64 | 99.42 | 99.69 | 99.75 | 99.90 | 99.90  |
| 120                     | 92.13  | 96.38  | 98.73 | 99.45 | 99.63 | 99.72 | 99.85 | 99.90  |
| 128                     | 91.90  | 96.62  | 98.60 | 99.44 | 99.55 | 99.71 | 99.84 | 99.84  |
| The average             | 92.68  | 97.10  | 98.79 | 99.61 | 99.76 | 99.87 | 99.90 | 99.95  |

Figure 7: Comparison of recognition rate between the ABN-CNN and existing algorithms in various noise environments.
and the other two loads, e.g., B and C sets, were used for testing, and so on.

3.6. Performance Analysis under Various Scenarios. As shown in Figure 9, the average recognition rate of FFT-SVM algorithm is less than 70% and FFT-MLP and FFT-DNN algorithms can reach about 80% recognition rate, while the WFK-CNN algorithm can achieve 90% average recognition rate. The ABN-CNN algorithm further improves the recognition rate up to 95.9% average. Especially, when set B (load for 2 hp) is used for training and set A (load for 1 hp) is used for testing, the recognition rate from FFT-SVM, FFT-MLP, and FFT-DNN models is 20% lower than that of the WFK-CNN model. On the other hand, the ABN-CNN has about average 6% higher recognition rate than the WFK-
CNN model. When set C (load for 3 hp) is trained and set A or B is tested, the recognition rate of ABN-CNN model increases more than 10% than the WFK-CNN model.

4. Conclusions

In this study, the proposed ABN-CNN model presents a relatively simple network using a large-scale convolution kernel with a small coiling convolution layer. Unlike the traditional Fourier transform, it is trained by the deep CNN with the adaptive batch normalization algorithm for bearing fault diagnosis. It can effectively reduce the difficulty in adjusting the parameters of the WFK-CNN model. When the SNR exceeds −4 dB, the proposed model can even reach a high recognition rate of more than 90% average using the one-dimensional bearing vibration signal. However, if the distribution of the test samples is significantly different from the training samples, the diagnostic performance efficiency may decrease. Also, noise interference and load change may affect the recognition rate. In a better situation, higher SNR can achieve recognition rates as high as 99%. In general, the experimental results confirm that the ABN-CNN algorithm can considerably improve the recognition rate of the WFK-CNN model and it is superior to FFT-SVM, FFT-MLP, and FFT-DNN algorithms. The future research can be further studied in increasing the recognition rate for low SNR and load variation under a noise circumstance. In addition, an axial piston pump that is widely used in various hydraulic pumps is one of the critical noise sources in the industry. Accordingly, it can be further studied on fault diagnosis of axial piston pumps using convolutional neural networks or deep belief networks via iterative learning processes.

Data Availability

The data that support the findings of this study are openly available in the CWRU Rolling Bearing Data Center at http://csegroups.case.edu/bearingdatacenter/home.

Conflicts of Interest

The authors declare no conflicts of interest.

Authors’ Contributions

Chao Fu developed the model. Qing Lv collected the data and carried out the performance analysis. Hsiung-Cheng Lin helped edit the manuscript. All the authors contributed to the writing of the final research paper.

Acknowledgments

This work was supported by the Key Fund of Hebei Provincial Education Department (Grant no. ZD2018028).

References

[1] A. Grzadziela, J. Musial, D. Musłewski, and M. Pająk, “A method for identification of non-coaxiality in engine shaft lines of a selected type of naval ships,” Polish Maritime Research, vol. 22, no. 1, pp. 65–71, 2015.
[2] L. Musłewski, M. Pająk, A. Grzadziela, and J. Musial, “Analysis of vibration time histories in the time domain for propulsion systems of minesweepers,” Journal of Vibration and Control, vol. 17, no. 3, pp. 1309–1316, 2013.
[3] M. Pająk, L. Musłewski, B. Landowski, and A. Grzadziela, “Fuzzy identification of the reliability state of the mine detecting ship propulsion system,” Polish Maritime Research, vol. 26, no. 1, pp. 55–64, 2019.
[4] D. Kolar, D. Lisják, and M. Pająk, “Rotating shaft fault prediction using convolutional neural network: a preliminary study,” Journal of KONES, vol. 26, no. 3, pp. 75–81, 2019.
[5] L. L. Li, P. Cheng, H. C. Lin, and H. Dong, “Short-term output power forecasting of photovoltaic systems based on the deep belief net,” Advances in Mechanical Engineering, vol. 9, no. 9, pp. 1–13, 2017.
[6] Q. Xuan, B. Fang, Y. Liu et al., “Automatic pearl classification machine based on a multistream convolutional neural network,” IEEE Transactions on Industrial Electronics, vol. 65, no. 8, pp. 6538–6547, 2018.
[7] H. Zhang, Z. Xie, H.-C. Lin, and S. Li, “Power capacity optimization in a photovoltaics-based microgrid using the improved artificial bee colony algorithm,” Applied Sciences, vol. 10, no. 9, p. 2990, 2020.
[8] Y. Ren, H. Li, and H.-C. Lin, “Optimization of feedforward neural networks using an improved flower pollination algorithm for short-term wind speed prediction,” Energies, vol. 12, no. 21, p. 4126, 2019.
[9] C. A. Pérez, P. A. Estévez, F. J. Galdames et al., “Trademark image retrieval using a combination of deep convolutional neural networks,” in Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), pp. 1–7, Rio de Janeiro, Brazil, July 2018.
[10] X. Mei, E. Pan, Y. Ma et al., “Spectral-spatial attention networks for hyperspectral image classification,” Remote Sensing, vol. 11, no. 8, p. 963, 2019.
[11] I. Sajedian, J. Kim, and J. Rho, “Finding the optical properties of plasmonic structures by image processing using a combination of convolutional neural networks and recurrent neural networks,” Microsystems & Nanoengineering, vol. 5, p. 27, 2019.
[12] F. Jia, Y. Lei, N. Lu, and S. Xing, “Deep normalized convolutional neural network for imbalanced fault classification of machinery and its understanding via visualization,” Mechanical Systems and Signal Processing, vol. 110, pp. 349–367, 2018.
[13] S. S. Kumar, D. M. Abraham, M. R. Jahneshahi, T. Isley, and J. Starr, “Automated defect classification in sewed closed circuit television inspections using deep convolutional neural networks,” Automation in Construction, vol. 91, pp. 273–283, 2018.
[14] L. Wen, X. Li, L. Gao, and Y. Zhang, “A new convolutional neural network-based data-driven fault diagnosis method,” IEEE Transactions on Industrial Electronics, vol. 65, no. 7, pp. 5990–5998, 2018.
[15] W. Xu and J. M. LeBeau, “A deep convolutional neural network to analyze position averaged convergent beam electron diffraction patterns,” Ultramicroscopy, vol. 188, pp. 59–69, 2018.
[16] S. Wang, I. W. Selesnick, G. Cai, Y. Feng, X. Sui, and X. Chen, “Nonconvex sparse regularization and convex optimization for bearing fault diagnosis,” IEEE Transactions on Industrial Electronics, vol. 65, no. 9, pp. 7332–7342, 2018.
[17] A. Sapena-Bano, J. Martinez-Roman, R. Puche-Panadero, M. Pineda-Sanchez, J. Perez-Cruz, and M. Riera-Guasp, “Induction machine model with space harmonics for fault diagnosis based on the convolution theorem,” *International Journal of Electrical Power & Energy Systems*, vol. 100, pp. 463–481, 2018.

[18] E. Khalastchi and M. Kalech, “A sensor-based approach for fault detection and diagnosis for robotic systems,” *Autonomous Robots*, vol. 42, no. 4, pp. 1–18, 2017.

[19] L. Li, D. Ma, and Z. Li, “Residual useful life estimation by a data-driven similarity-based approach,” *Quality and Reliability Engineering International*, vol. 33, no. 2, pp. 231–239, 2017.

[20] Y. Gao, M. Karimi, A. A. Kudreyko, and W. Song, “Spare optimistic based on improved ADMM and the minimum entropy de-convolution for the early weak fault diagnosis of bearings in marine systems,” *ISA Transactions*, vol. 78, pp. 6–29, 2018.

[21] H. Wang, J. Chen, and G. Dong, “Fault diagnosis of rolling bearing’s early weak fault based on minimum entropy de-convolution and fast Kurtogram algorithm,” *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 229, no. 16, pp. 2890–2907, 2015.

[22] X. Guo, L. Chen, and C. Shen, “Hierarchical adaptive deep convolution neural network and its application to bearing fault diagnosis,” *Measurement*, vol. 93, pp. 490–502, 2016.

[23] P. Baldi, P. Sadowski, and Z. Lu, “Learning in the machine: random backpropagation and the deep learning channel,” *Artificial Intelligence*, vol. 260, pp. 1–35, 2018.

[24] S. Guo, T. Yang, W. Gao, and C. Zhang, “A novel fault diagnosis method for rotating machinery based on a convolutional neural network,” *Sensors*, vol. 18, no. 5, pp. 5–13, 2018.

[25] F.-C. Chen and M. R. Jahanshahi, “NB-CNN: deep learning-based crack detection using convolutional neural network and naive bayes data fusion,” *IEEE Transactions on Industrial Electronics*, vol. 65, no. 5, pp. 4392–4400, 2018.

[26] N. Cai, G. Cen, J. Wu, F. Li, H. Wang, and X. Chen, “SMT solder joint inspection via a novel cascaded convolutional neural network,” *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 8, no. 4, pp. 670–677, 2018.

[27] S. Wang, J. Xiang, Y. Zhong, and Y. Zhou, “Convolutional neural network-based hidden Markov models for rolling element bearing fault identification,” *Knowledge-Based Systems*, vol. 144, pp. 65–76, 2018.

[28] J.-H. Zhong, P. K. Wong, and Z.-X. Yang, “Fault diagnosis of rotating machinery based on multiple probabilistic classifiers,” *Mechanical Systems and Signal Processing*, vol. 108, pp. 99–114, 2018.

[29] W. A. Smith and R. B. Randall, “Rolling element bearing diagnostics using the case western reserve university data: a benchmark study,” *Mechanical Systems and Signal Processing*, vol. 64–65, pp. 100–131, 2015.

[30] Y. Yang, P. Fu, and Y. He, “bearing fault automatic classification based on deep learning,” *IEEE Access*, vol. 6, pp. 71540–71554, 2018.

[31] M. Lei, G. Meng, and G. Dong, “Fault detection for vibration signals on rolling bearings based on the symplectic entropy method,” *Entropy*, vol. 19, no. 11, p. 607, 2017.

[32] Y. Yang and P. Fu, “Rolling-Element bearing fault data automatic clustering based on wavelet and deep neural network,” *Shock and Vibration*, vol. 2018, Article ID 3047830, 11 pages, 2018.

[33] S. Wang and J. Xiang, “A minimum entropy deconvolution-enhanced convolutional neural networks for fault diagnosis of axial piston pumps,” *Soft Computing*, vol. 24, no. 4, pp. 2983–2997, 2020.

[34] S. Wang, J. Xiang, Y. Zhong, and H. Tang, “A data indicator-based deep belief networks to detect multiple faults in axial piston pumps,” *Mechanical Systems and Signal Processing*, vol. 112, pp. 154–170, 2018.

[35] CWRU, Case Western Reserve University, “Bearing data center website,” Cleveland, OH, USA, http://csegroups.case.edu/bearingdatacenter/home.