IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 14, 2021, accepted February 26, 2021, date of publication March 10, 2021, date of current version March 22, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3065195

An End-to-End Intelligent Fault Diagnosis Application for Rolling Bearing Based on MobileNet

WENBING YU[1] **AND PIN LV**[2]

[1]School of Higher Vocational Technology, Shanghai Dianji University, Shanghai 201306, China
[2]School of Electronic Information Engineering, Shanghai Dianji University, Shanghai 201306, China

Corresponding author: Wenbing Yu (yuwb@sdju.edu.cn)

This work was supported in part by the Intelligent Test Platform for Motor from Enterprise Application Project under Grant 20B228.

ABSTRACT To find out the hidden danger in the industrial production process in time, it is necessary to monitor the health condition of the key components of the mechanical system in operation. However, traditional fault diagnosis methods usually adopt manual feature extraction, which not only costs expensively and depends on prior knowledge. Therefore, it is of great significance to study the application of automatic fault identification based on the original vibration signals. Recently, existing studies have shown that most of fault diagnoses are implemented by using deep neural network. Although these methods have achieved satisfactory performances, there are obvious limitations in real applications, that is, the complexity of deep neural network requires a lot of hardware computing resources. This hinders the development of online fault diagnosis tools. To solve this problem, this paper proposes a fault diagnosis model based on lightweight convolutional neural network MobileNet, and realizes an end-to-end intelligent fault classification and diagnosis application. We evaluated the proposed method with the rolling bearing dataset from Western Reserve University. The best average precision, recall and F1 score of ten different bearing health conditions are about 96%, 82% and 88%, respectively. In addition, we also compare the accuracy of the rolling bearing fault diagnosis classification model under the standard ReLU and the improved ReLU. Experimental results show that both obtain good performances, but the improved ReLU reaches the over 96% accuracy more rapidly.

INDEX TERMS Fault diagnosis, rolling bearing, deep neural network, mobilenet.

I. INTRODUCTION

Bearings are a fundamental component of all kinds of rotary machinery, and its health conditions are directly related to the normal operation of rotating machinery. With the rise of industry 4.0, the application of fault diagnosis to ensure the reliability and safety of mechanical system has been paid more and more attention by industry and academia [1]–[3]. As we all know, it is the key of fault diagnosis to extract features that can represent fault information from original vibration signals. Based on this main line, fault diagnosis technology has experienced two development stages such as traditional methods [4]–[8] and deep learning based on methods [9]–[11].

The associate editor coordinating the review of this manuscript and approving it for publication was Her-Terng Yau.

As traditional fault diagnosis methods, feature extraction and classifier construction are independent, and signal processing technologies are often used to extract features. For example, Randall and Antoni [4] investigated that rolling bearing signals have the characteristics of randomness. By modeling it as pseudo cycle stationary signals, local defect pulse signals can be separated as fault features. Lu *et al.* [6] employed empirical mode decomposition to decompose a vibration signal into intrinsic mode functions, which represented the vibration signal characteristics of rolling bearings. Li *et al.* [7] transformed time-frequency signals into frequency features using Fast Fourier Transform, and then adopted a fuzzy clustering algorithm to realize fault diagnosis. Cui *et al.* [8] constructed two dictionaries, such as a pulse time-frequency dictionary and a modulation dictionary, to propose a method called DDMP (Double-Dictionary Matching Pursuit) to decompose and reconstruct rolling

bearing vibration signals. Liu and Xiang [12] extracted the high-frequency terms from the original vibration signals using a first-order kernel regression residual decomposition, and then purified these high-frequency terms containing fault information through a conversion technology. Zhen *et al*. [13] utilized the third-order statistics called Bispectrum to suppress the Gaussian noise in vibration signals, and effectively extracted fault features from non-stationary signals. Ming *et al*. [14] proposed to filter original vibration signals using cyclic Wiener filters, and then used envelope spectrum analysis to extract the most influential features for fault classification. Jiang *et al*. [15] investigated a semi-supervised kernel marginal Fisher analysis method, which directly extracted low dimensional features from the original high-dimensional vibration signals. Their researches showed that these low dimensional features effectively identified fault diagnosis when it took as the input of KNN (K Nearest Neighbor) classifier. Based on the above analysis, we can see that traditional fault diagnosis methods have achieved good application results, but there are two major limitations. (1) the signal processing technology used to extract features largely depends on the prior knowledge of experts. (2) the fault classifier is not universal because the extracted features are closely related to specific applications.

With the advent and development of deep learning, such problems have been solved to a certain extent. In sharp contrast with traditional fault diagnosis methods, the methods based on deep learning integrates feature extraction and fault classification, which fully demonstrates the ability of deep neural network to automatically learn features. At present, deep belief network [16], autoencoder [17], CNN (convolutional neural network) [18], [19] and RNN (recurrent neural network) [19]–[21] are the most widely used deep learning technologies in the field of fault diagnosis. In order to improve the accuracy of fault diagnosis, researchers have explored the data representation ability of deep belief network and auto encoder, the data transformation ability of CNN and the timing processing ability of RNN. For example, in the researches of fault diagnosis based on deep belief network and autoencoder, Deutsch and He [22] proposed to construct a deep belief network using a stacked version of an Restricted Boltzmann Machine, and then add a linear regression layer at the top of the deep belief network to realize the prediction of the remaining use life of bearings; Zhang *et al*. [23] took the fused frequency spectrums as the input of a deep belief network, and used these inputs to train the deep belief network model which recognized the degradation of ball screw. Based on the advantages of deep belief network and quantum inspired neural network, Gao *et al.* [24] constructed a quantum inspired neural network using linear superposition of deep belief network with quantum interval in the last hidden layer, which was used for fault detection of aircraft fuel system. When constructing the power system fault classification model based on multi-layer perceptron, oh *et al*. [25] transformed vibration sensor signals into vibration images, and then used a histogram of oriented gradients related to the vibration images as the input of deep belief network to learn the high-level features in the images. Sun *et al*. [26] adopted the idea of compressed sensing to construct the mapping relationship between all the information of faults and low dimensional spaces with appropriate compression ratio, and then obtained the compressed data containing the original fault signals, and finally used these compressed data to build a stacked sparse autoencoder to extract fault features. Sohaib *et al*. [27] proposed a two-layer fault diagnosis scheme, which used a hybrid feature set and a sparse stacked autoencoder to realize fault diagnosis of rolling bearings. Among them, the function of the mixed feature set is to isolate bearings under different health states, and the sparse stacked autoencoder is used to extract the intrinsic information in the mixed feature set.

For the methods based on CNN, Wei *et al*. [28] studied the influence of activation functions on the performance of CNN, and proposed an improved model named ReLU-CNN for mechanical fault diagnosis. The experimental results showed that the proposed model has good performance and fast convergence. Xia *et al*. [29] used the structural characteristics of CNN to skillfully implement the sensor information fusion during fault diagnosis. The input of CNN was all the information collected by multiple sensors, which was represented by matrix. Abdeljaber *et al*. [30] proposed an adaptive method based on 1-D CNN to realize structure health monitoring based on vibration signals. Guo *et al*. [31] constructed a model named deep convolution transfer learning network, which consists of two modules. One is a health recognition module; the other is a domain adaptation module. The health recognition module is completed using 1-DCNN, and the domain adaptation module is convenient for 1-DCNN to learn domain invariant features by maximizing the domain recognition error and minimizing the probability distribution distance of domain recognition. The effectiveness of the method is verified in six migration fault diagnosis experiments. Based on CNN and multi-layer perceptron, Li *et al*. [32] trained some basic models with a large number of source data, then migrated the basic models to the target data of different workloads and different mechanical parts, and finally realized the fault diagnosis model trained by mechanical part A to predict the fault of mechanical Part B. Li *et al.* [19] introduced attention mechanism in the combination of CNN and LSTM (Long Short Term Memory) to achieve mechanical fault diagnosis, which can more effectively locate the typical fault features in the original data.

In the aspect of fault diagnosis of RNN, Yuan *et al*. [33] studied the performance of four different recurrent neural network models applied to aero-engine fault diagnosis tasks, including standard RNN, GRU (Gate Recurrent Unit), AdaBoost LSTM and standard LSTM. The experimental results showed that the standard LSTM outperformed the other three models. Zhao *et al*. [34] constructed a tool wear monitoring health model based on LSTM, and predicted the corresponding tool wear by encoding the original sensor signals. In addition, a convolutional bidirectional long short

term memory network was designed based on afore model, where the convolution network was used to extract local features of time series, and the bidirectional LSTM model was used to encode the extracted features. Finally, a stacked full connection network and a linear regression layer are used to predict tool wear. The experimental results showed that the proposed method had good prediction performances. Malhotra *et al*. [35] proposed a LSTM Encoder-Decoder (LSTM-ED) scheme to predict the residual life of mechanical components. By training LSTM-ED to reconstruct the time series of the health state of the mechanical system, the health index values of the mechanical components were estimated using the reconstruction error, and the residual life of the mechanical components was predicted by the health index.

Although the above existing methods achieved better performances in fault diagnosis, the heavily dependence of deep learning on the hardware platform and the intensive computing process are not considered. In fact, the application of these methods is very difficult under the limited computing resources and storage space, which hinders the use and popularity of the intelligent fault diagnosis system from being severe working conditions. Therefore, the MobileNet [36], which focus on the lightweight CNN network in mobile or embedded devices, has begun to receive attentions. The MobileNet can balance the latency and accuracy of the network. Since 2017, it has been researched and applied in some fields such as image recognition [37], health care [38], automatic driving [39] and so on. Inspired by these applications, we investigate how to build a fault diagnosis model based on MobileNet from the perspective of end-to-end application, and deploy an application that starts with model creation and terminates at the web application.

In this study, we propose an end-to-end intelligent classification solution for fault diagnosis based on MobileNet which is a lightweight CNN. Users can run our proposed model on the browser without installing any software, and get the classification results with over 95% in real time. To the best of our knowledge, this is the first end-to-end solution which performs fault diagnosis recognition and achieves the state-of-the-art performance.

In summary, in this paper we make the following contributions:

1) We propose to study fault diagnosis classification using a lightweight CNN, which aims to build an end-to-end fault diagnosis classification application.

2) We build a fault diagnosis classification model based on MobileNet. An improved activation function ReLU was investigated for the performance of our built model.

3) We conduct two kinds of experiments on a public rolling bearing dataset from Western Reserve University. One is to test the performance of the proposed model; the other is to implement the end-to-end application of the proposed model. Experimental results on two tasks prove the effectiveness and efficiency of the proposed approach, which provides a possible way for enterprises to realize online monitoring under the severe working conditions.

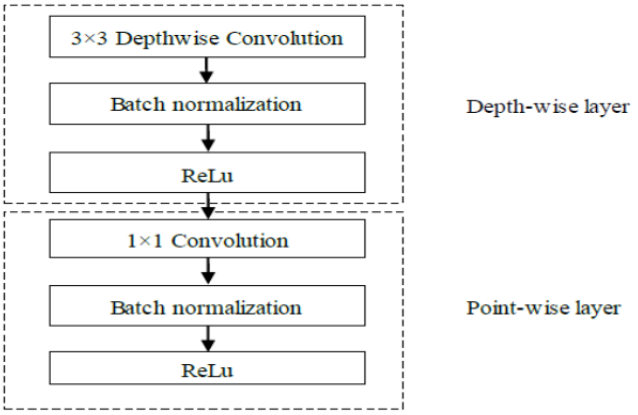

FIGURE 1. The basic convolutional structure of MobileNet.

The rest of the paper is organized as follows. We describe the principles of MobileNet in Section 2. Section 3 presents the network structure and training of fault diagnosis classification model based on MobileNet. Section 4 describes the experimental dataset in detail. Section 5 analyzes experimental results from different angles. Section 6 demonstrates the end-to-end solution for the proposed model and Section 7 concludes the paper.

II. PRINCIPLES OF MOBILENET

MobileNet is a lightweight CNN proposed by Google in 2017, which can be used in mobile terminals and improve the real-time performance of deep learning under limited hardware conditions [36]. Figure 1 shows the basic convolution structure of the MobileNet. A standard convolution operation can be decomposed into a depth-wise convolution and a point-wise convolution by using deep and separable convolution. The function of the depth-wise convolution layer is to filter the input channel, while the point-wise convolution layer is to combine the output of the depth-wise convolution layer linearly in order to obtain a new feature map. After each convolution operation, batch normalization algorithm and ReLU activation function are used to realize automatic adjustment of data distribution. Deep and separable convolution network can accelerate the training of MobileNet and greatly reduce the amount of computation. The reasons are as follows:

The standard convolution structure can be described as (1).

$$G_s = \sum_m F_{m,n} \cdot Input_m \tag{1}$$

where m and n are respectively the number of input channels and output channels. $F_{m,n}$ is the filter, $Input_m$ denotes the input, including feature map, which use the fill style of zero padding. If the size of $Input_m$ is $D_{Input} \times D_{Input}$, it is necessary to have n filters with m channels and the size of $D_{output} \times D_{output}$ before outputting feature maps of the size $D_{output} \times D_{output}$. So the computing cost of standard convolution is shown in (2).

$$\mathcal{C}_s = D_{Output} \times D_{Output} \times m \times n \times D_{Input} \times D_{Input} \tag{2}$$

TABLE 1. The network structure of fault diagnosis classification model based on MobileNet.

Layer type /stride	Filter shape	Input shape
Convolution /2	(3, 3, 1, 32)	(50, 50, 1)
Depth-wise convolution/1	(3, 3, 32)	(25, 25, 32)
Point-wise convolution/1	(1, 1, 32, 64)	(25, 25, 32)
Depth-wise convolution/2	(3, 3, 64)	(25, 25, 64)
Point-wise convolution/1	(1, 1, 64, 128)	(15, 15, 64)
Depth-wise convolution/1	(3, 3, 128)	(15, 15, 128)
Point-wise convolution/1	(1, 1, 128, 128)	(15, 15, 128)
Depth-wise convolution/2	(3, 3, 128)	(15, 15, 128)
Point-wise convolution/1	(1, 1, 128, 256)	(8, 8, 128)
Depth-wise convolution/1	(3, 3, 256)	(8, 8, 256)
Point-wise convolution/1	(1, 1, 256, 256)	(8, 8, 256)
Depth-wise convolution/2	(3, 3, 256)	(8, 8, 256)
Point-wise convolution/1	(1, 1, 256, 512)	(5, 5, 256)
5 * (Depth-wise convolution/1 + Point-wise convolution/1)	(3, 3, 512) (1, 1, 512, 512)	(5, 5, 512) (5, 5, 512)
Depth-wise convolution/2	(3, 3, 512)	(5, 5, 512)
Point-wise convolution/1	(1, 1, 512, 1024)	(3, 3, 1024)
Depth-wise convolution/1	(3, 3, 1024)	(3, 3, 1024)
Point-wise convolution/1	(1, 1, 1024, 1024)	(3, 3, 1024)
Average pooling /1	(3, 3)	(3, 3, 1024)
Fully connected /1	(1024, 1000)	(1, 1, 1024)
Softmax /1	None	(1, 1, 1000)

By contrast, the depth-wise convolution structure can be expressed as (3).

$$G_M = \sum \hat{F}_{1,m} \cdot Input_m \tag{3}$$

where $\hat{F}_{1,m}$ is the filter, $Input_m$ has the same meaning as (1). During the depth-wise convolution, there must have m filters with 1 channels and the size of $D_{output} \times D_{output}$. During the point-wise convolution, it is necessary to have n filters with m channels and the size of 1×1. In this case, the computing cost of the deep separable convolution structure is computed using (4).

$$\begin{aligned} \mathcal{C}_M = {} & D_{Output} \times D_{Output} \times m \times D_{Input} \times D_{Input} \\ & + m \times n \times D_{Input} \times D_{Input} \end{aligned} \tag{4}$$

Comparing with standard convolution operation, the computing cost is reduced by $\frac{1}{n} + \frac{1}{D^2_{output}}$.So the deep and separable convolutional structure enables the MobileNet to speed up the training process and greatly reduces the amount of calculation.

III. NETWORK STRUCTURE AND TRAINING OF FAULT DIAGNOSIS CLASSIFICATION MODEL BASED ON MOBILENET

The network structure of the fault diagnosis classification model based on MobileNet is shown in Table 1, where all different types of layers used, the filter shape of each layer, and the input shaped in each layer are described in great detail.

In this work, we selected 2500 data points as a sample. Therefore, the input of our proposed model is a one-channel raw vibration signals with 50×50. Firstly, Zero Padding is used to fill the corner and boundary information of the vibration signals to prevent the loss of boundary information during convolutions. Then, after using a 3×3 filter and a stride with 2, the output of standard convolution is executed to get 32 feature maps of size 25×25. Secondly, thirteen depth-wise convolution layers and point-wise convolution layers are stacked to form the main body of the fault diagnosis classification model. Finally, the convolution layer of the 13th point-wise is connected with the full connection layer to realize the classification of fault classification.

Each depth-wise convolution operation in the model uses a filter of size 3×3. After each depth-wise convolution and point-wise convolution, batch normalization and ReLU activation functions are performed successively to prevent gradient disappearance, adjust network parameters, and speed up network training. In order to prevent information loss during depth-wise convolution, Zero Padding was performed between the second and the third deep convolution layer, the fifth and the sixth deep convolution layer, the eleventh and the twelfth deep convolution layer, respectively. To prevent over-fitting in the process of model training, the dropout mechanism is adopted.

After building the classification model of fault diagnosis based on MobileNet, we choose Keras framework to train it in this study. The free GPU is provided by Google Colaboratory. The weight of model is initialized by the Gaussian distribution. In order to highlight the sensitivity of the model to the three types of rolling bearing faults (BF, IF, and OF), the initial weights of the three types of rolling bearing faults were set to be greater than that weights of normal rolling bearings. The initial batch size is set to 100 samples during training and validation. The weight of the model is updated by Adam optimizer with an initial learning rate of 0.01, the size of the validation set is the 10 percent of the training set and the classification cross-entropy is selected as the loss function. The metric of *Precision*, *Recall*, *F1-score and Accuracy* were used to evaluate the proposed model performance.

IV. DATA DESCRITPION

The original fault data of rolling bearing studied in this paper are collected by Case Western Reserve University (CWRU), which includes 4 health types and 10 different health conditions. The four health types are respectively normal, rolling ball fault (BF), inner fault (IF) and outer fault (OF), and the corresponding diameters of each fault degree are 0.007mm, 0.014mm and 0.021mm respectively. There are three data acquisition points of each fault state, including drive end, fan end and base accelerator end. To validate the proposed method, all raw vibration signals are randomly selected from the original datasets with load of 1, 2 and 3 horsepower and sampling frequency of 12 kHz to form three different datasets D1, D2 and D3 in this work. D1, D2 and D3 all contain 17500 training samples and 2500 test samples, and each sample contains 2500 data points, as shown in Table 1. In addition, D4, which is the union set of D1, D2, and D3, has

TABLE 2. Description of the rolling bearing fault datasets.

Datasets	Load(horsepower)	Number of samples	Fault type	Fault diameter(mm)	Alias	Label
D1/D2/D3/D4	1/2/3/1-3	20000/20000/20000/60000	N	0	Normal	1
D1/D2/D3/D4	1/2/3/1-3	20000/20000/20000/60000	BF	0.007	B_007	2
D1/D2/D3/D4	1/2/3/1-3	20000/20000/20000/60000	BF	0.014	B_014	3
D1/D2/D3/D4	1/2/3/1-3	20000/20000/20000/60000	BF	0.021	B_021	4
D1/D2/D3/D4	1/2/3/1-3	20000/20000/20000/60000	IF	0.007	I_007	5
D1/D2/D3/D4	1/2/3/1-3	20000/20000/20000/60000	IF	0.014	I_014	6
D1/D2/D3/D4	1/2/3/1-3	20000/20000/20000/60000	IF	0.021	I_021	7
D1/D2/D3/D4	1/2/3/1-3	20000/20000/20000/60000	OF	0.007	O_007	8
D1/D2/D3/D4	1/2/3/1-3	20000/20000/20000/60000	OF	0.014	O_014	9
D1/D2/D3/D4	1/2/3/1-3	20000/20000/20000/60000	OF	0.021	O_021	10

600000 samples under all the loads. The detailed information is shown in Table 1, and the vibration signals under the different health conditions are shown in Figure 2.

V. EXPERIMENTS AND RESULTS

A. EVALUATION METRICS

Fault diagnosis is generally regarded as a classification problem that is implemented by supervised learning. Following [40], we use *Precision*, *Recall*, *F1-score*, and *Accuracy* for taxonomy evaluation in this study. *Precision* is the ratio $tp / (tp + fp)$ where tp is the number of true positives and fp the number of false positives. *Precision* is intuitively the ability of the classifier not to label as positive a sample that is negative. *Recall* is the ratio $tp / (tp + fn)$ where tp is the number of true positives and fn the number of false negatives. *Recall* is intuitively the ability of the classifier to find all the positive samples. The *F1-score* can be interpreted as a weighted average of the precision and recall, where *F1- score* reaches its best value at 1 and worst score at 0. In multilabel classification, *Accuracy* means the set of labels predicted for a sample must exactly match the corresponding set of labels in its true labels.

B. RESULTS DISCUSSION

We design three experiments for the proposed model in order to find the optimal parameters. The experimental results are represented in this section.

1) BATCH SIZE

The batch size is the number of training instances observed before the optimizer performs a weight update. Table 3 shows performance of different values of batch size. It can be observed that the model achieves the best classification when batch size is equal to 400. When batch size is greater than 400, although the average convergence time of the model is shorter, other metrics show a downward trend.

2) DROPOUT

Dropout is a well-known form of regularization. We further study the influence of different dropout values on model.

TABLE 3. Performance of model w.r.t. different batch size.

Batch size	Average Converge time(s)	*Precision*	*Recall*	*F1-scroe*
100	696	0.796	0.696	0.743
200	366	0.837	0.728	0.779
300	362	0.849	0.66	0.743
400	335	0.947	0.825	0.882
500	324	0.853	0.776	0.813

TABLE 4. Performance of model w.r.t. different dropout value.

Dropout	Average Converge time(s)	*Precision*	*Recall*	*F1-scroe*
0.1	515	0.786	0.686	0.733
0.2	470	0.867	0.738	0.797
0.3	474	0.829	0.680	0.747
0.4	429	0.957	0.815	0.880
0.5	473	0.903	0.796	0.846

Table 4 presents the results. We can see that the value of dropout has a great impact on the performance of the model. When dropout equals 0.4, the model has the best performance in *Precision*, *Recall* and *F1-score*.

3) LOSS AND ACCURAY

In addition, in order to examine learning and performance of the model, we conduct some experiments to observe the change of the loss function and classification accuracy of the model when batch size and dropout are equal to 400 and 0.4, respectively. Figure 3 shows that the loss function curves of training set and validation set. It can be observed that the proposed model shows a good learning as the training accuracy increase with the number of iterations along with symmetric downward sloping of training loss curve. The small gap between training and validation curves represents a good-fit, indicating model can generalize well on fresh unseen fault.

4) IMPORVED ACTIVATION FUNCTION

Activation functions are crucial elements in deep learning neural networks. To observe the effect of activation function

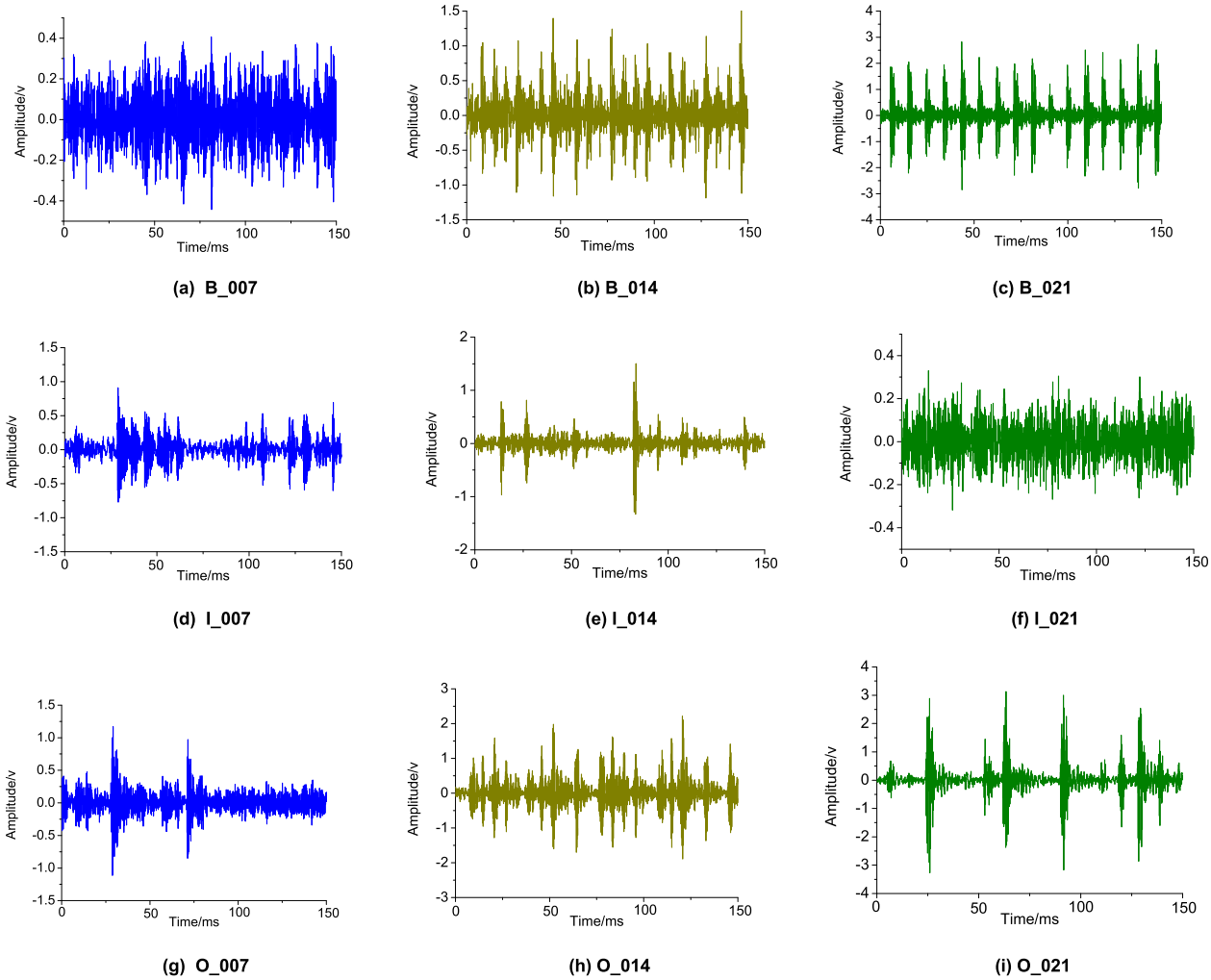

FIGURE 2. Vibration signals of bearings with different conditions from one sensor.

on the performance of the proposed model, we introduce tws improved ReLU activation functions shown in (4) and (6) to replace the standard ReLU activation function in Section 3. According to Figure 4, it is found that (1) when the coefficients of the improved ReLU activation function are 0.01 and 0.25, the accuracy of the model reaches 100% after about one epoch. When using the standard ReLU activation function, the accuracy of the model can only reach over 96% after about 18 epochs. This indicates that the performance of the proposed model can be improved significantly by introducing the improved ReLU activation function; (2) when the standard ReLU activation function is used in the proposed model, the loss convergence is about 18 epochs, but when the improved ReLU function is used, the loss of the model is very fast, about 5 epochs. Moreover, the function with coefficient of 0.25 converges faster than that with coefficient of 0.01.

$$f(x) = \max(0.01x, x) \tag{5}$$

$$f(x) = \max(0.25x, x) \tag{6}$$

VI. THE END-TO-END SOLUTION

In order to achieve the end-to-end intelligence diagnosis for rolling bearing faults classification, we use the TensorFlow.js machine learning framework in this paper. Developers can use the JavaScript language to model and train in the browser using various APIs provided by TensorFlow.js [41]. The core of these APIs is the concept of models and layers. The Layers API provided by Tensorflow.js better reflects the Kreas API. Therefore, Keras and TensorFlow.js can be converted to each other, and developers can easily load the pre-trained Keras model in Tensorflow.js.

To load the pre-trained rolling bearing fault diagnosis classification model in Tensorflow.js, we have to convert it to the Tensorflow.js Layers format using the program statement *converters.save_keras_model(model, 'tfjs_dir')*. The parameter *tfjs_dir* represents the directory where the file is output. The converted Tensorflow.js Layers contains the *model.json* file and the directory of a set of weighted file fragments in binary format. The *model.json* file contains the structure of the pre-trained rolling bearing fault diagnosis classification

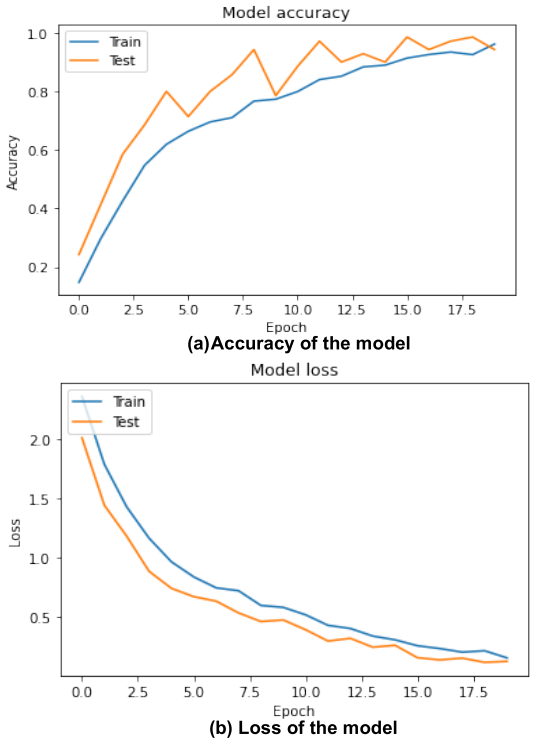

FIGURE 3. Performance of the rolling bearing fault diagnosis classification based on MobileNet.

model and its corresponding weight file. After the rolling bearing fault diagnosis classification model is converted into the format of Tensorflow.js, the model information can be obtained using JavaScript.

The pseudo code of using the pre-trained rolling bearing fault diagnosis classification model to implement the intelligent diagnosis in browser is as follows:

Input: skin lesion images
Procedure:

1. import *@tensorflow/tfjs* library
2. extracting the input size of the rolling bearing fault sample containing 2500 data points
3. normalize the input
4. load *model.json* file
5. obtain the predicted s rolling bearing fault and its probability
6. rank the probability in descending

Output: top3 rolling bearing fault diagnosis

Finally, the prediction results are returned to the client via JavaScript. This end-to-end application begins with model building, ending with a live web application, and all operations are carried out on the user side, greatly convenient to use for enterprises in the severe working conditions.

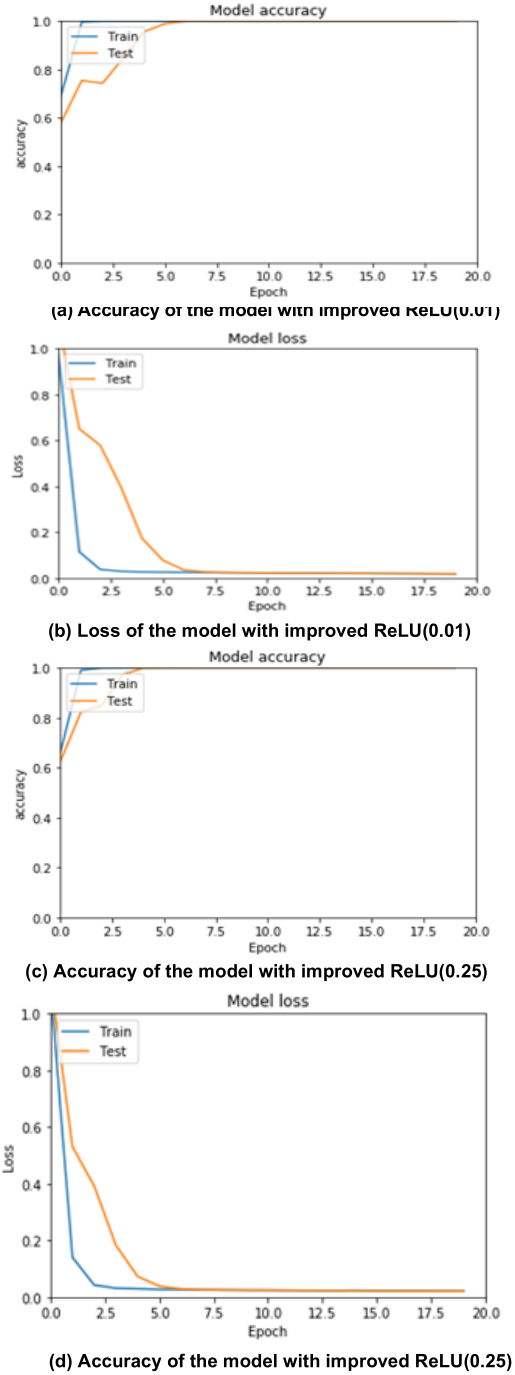

FIGURE 4. Performance of the rolling bearing fault diagnosis classification based on MobileNet with improved ReLU.

In addition, we randomly selected some rolling bearing fault samples from the test set and record the diagnosis time for each fault sample based on the configuration in Table 5, The experiment results of the diagnosis time are listed ten different fault types as shown in Table 6. It can be observed that the longest average diagnosis time is 2.2ms and the shortest average diagnosis time is 1.4ms. It means that the application can provide user with a real-time diagnosis result.

TABLE 5. The configurations of the proposed model application.

Configuration of Web Server	CPU/Memory	1 Core / 1G
	Single Web space	200MB
	Operating system	CentOS 6.5 / 64 bits
Configuration of local networking	CPU	2.6GHz Intel Core i7
	Memory	16GB DDR4
	Operating system	macOS Mojave 10.14.5
	Browser	Google Chrome 76.0.3809.100
	Operators in Test Environment	China Unicom / 4G Network
	Network Speed in Test Environment	Download Speed: 9.97Mbps Upload speed: 8.74 Mbps

TABLE 6. The diagnosis time for randomly selected fault samples.

Fault type	Diagnosis time of randomly selected fault sample					Average diagnosis time
	first	second	third	fourth	fifth	
Normal	2ms	2ms	2ms	2ms	2ms	2ms
B_007	2ms	2ms	2ms	2ms	2ms	2ms
B_014	2ms	2ms	2ms	2ms	2ms	2ms
B_021	1ms	1ms	1ms	2ms	2ms	1.4ms
I_007	2ms	1ms	2ms	2ms	1ms	1.6ms
I_014	2ms	2ms	2ms	2ms	2ms	2ms
I_021	3ms	1ms	2ms	2ms	2ms	2ms
O_007	2ms	1ms	2ms	2ms	1ms	2ms
O_014	2ms	2ms	2ms	2ms	1ms	1.8ms
O_021	2ms	2ms	2ms	2ms	3ms	2.2ms

VII. CONCLUSION

In this paper, we built a rolling bearing diagnosis classification model based on MobileNet and performed experiments on tens types of faults. The results show that the proposed model has an ability to achieve intelligent classification in real time. Based on experiments, we find that although the standard ReLU activation function can obtain a good performance on our proposed model, the improved ReLU is better. Therefore, the activation function is still critical to the classification performance of the model. In addition, we deployed an end-to-end solution based on the proposed the rolling bearing diagnosis classification model combining with web applications.

On the other hand, we reiterate that the effective acquisition and collation of high-quality rolling bearing faults is a key prerequisite for the successful application of artificial intelligence technology in the manufacture. With the increase of fault data and the improvement of neural network structure, the classification model of fault diagnosis based on MobileNet will continue to improve its performance. We believe that in the future, enterprises can use this solution to check the health state of the rolling bearings under the severe working conditions.

REFERENCES

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

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

[3] S. Jeschke, C. Brecher, H. Song, and D. B. Rawat, *Industrial Internet of Things*. Cham, Switzerland: Springer, 2017.

[4] R. B. Randall and J. Antoni, "Rolling element bearing diagnostics: A tutorial," *Mech. Syst. Signal Process*, vol. 25, no. 2, pp. 485–520, 2011.

[5] D. Tuia, G. Camps-Valls, G. Matasci, and M. Kanevski, "Rolling bearing fault feature extraction technique and method," *Coal Mine Machinery*, vol. 2, no. 10, pp. II-65–II-68, 2010.

[6] L. Lu, J. Yan, and C. W. D. Silva, "Dominant feature selection for the fault diagnosis of rotary machines using modified genetic algorithm and empirical mode decomposition," *J. Sound Vibrat.*, vol. 344, pp. 464–483, May 2015.

[7] C. Li, J. V. D. Oliveira, M. Cerrada, F. Pacheco, D. Cabrera, V. Sanchez, and G. Zurita, "Observer-biased bearing condition monitoring: From fault detection to multi-fault classification," *Eng. Appl. Artif. Intell.*, vol. 50, pp. 287–301, Apr. 2016.

[8] L. Cui, X. Gong, J. Zhang, and H. Wang, "Double-dictionary matching pursuit for fault extent evaluation of rolling bearing based on the Lempel-Ziv complexity," *J. Sound Vibrat.*, vol. 385, pp. 372–388, Dec. 2016.

[9] T. Chen, Z. Wang, X. Yang, and K. Jiang, "A deep capsule neural network with stochastic delta rule for bearing fault diagnosis on raw vibration signals," *Measurement*, vol. 148, Dec. 2019, Art. no. 106857.

[10] D.-T. Hoang and H.-J. Kang, "A survey on deep learning based bearing fault diagnosis," *Neurocomputing*, vol. 335, pp. 327–335, Mar. 2019.

[11] S. Khan and T. Yairi, "A review on the application of deep learning in system health management," *Mech. Syst. Signal Process.*, vol. 107, pp. 241–265, Jul. 2018.

[12] H. Liu and J. Xiang, "Kernel regression residual decomposition-based synchroextracting transform to detect faults in mechanical systems," *ISA Trans.*, vol. 87, pp. 251–263, Apr. 2019.

[13] D. Zhen, J. Guo, Y. Xu, H. Zhang, and F. Gu, "A novel fault detection method for rolling bearings based on non-stationary vibration signature analysis," *Sensors*, vol. 19, no. 18, p. 3994, Sep. 2019.

[14] Y. Ming, J. Chen, and G. Dong, "Weak fault feature extraction of rolling bearing based on cyclic Wiener filter and envelope spectrum," *Mech. Syst. Signal Process.*, vol. 25, no. 5, pp. 1773–1785, Jul. 2011.

[15] L. Jiang, J. Xuan, and T. Shi, "Feature extraction based on semi-supervised kernel marginal Fisher analysis and its application in bearing fault diagnosis," *Mech. Syst. Signal Process.*, vol. 41, nos. 1–2, pp. 113–126, Dec. 2013.

[16] H. Wang, H. Wang, G. Jiang, J. Li, and Y. Wang, "Early fault detection of wind turbines based on operational condition clustering and optimized deep belief network modeling," *Energies*, vol. 12, no. 6, p. 984, Mar. 2019.

[17] Y. Qi, C. Shen, D. Wang, J. Shi, X. Jiang, and Z. Zhu, "Stacked sparse auto encoder based deep network for fault diagnosis of rotating machinery," *IEEE Access*, vol. 5, pp. 15066–15079, 2017.

[18] O. Janssens, V. Slavkovikj, B. Vervisch, K. Stockman, M. Loccuer, S. Verstockt, R. Van DeWalle, and S. Van Hoecke, "Convolutional neural network based fault detection for rotating machinery," *J. Sound Vibrat.*, vol. 377, pp. 331–345, Sep. 2016.

[19] X. Li, W. Zhang, and Q. Ding, "Understanding and improving deep learning-based rolling bearing fault diagnosis with attention mechanism," *Signal Process.*, vol. 161, pp. 136–154, Aug. 2019.

[20] J. Lei, C. Liu, and D. Jiang, "Fault diagnosis of wind turbine based on long short-term memory networks," *Renew. Energy*, vol. 133, pp. 422–432, Apr. 2019.

[21] H. Liu, J. Zhou, Y. Zheng, W. Jiang, and Y. Zhang, "Fault diagnosis of rolling bearings with recurrent neural network-based autoencoders," *ISA Trans.*, vol. 77, pp. 167–178, Jun. 2018.

[22] J. Deutsch and D. He, "Using deep learning based approaches for bearing remaining useful life prediction," in *Proc. Annu. Conf. Prognostics Health Manage. Soc.*, 2016, pp. 292–298.

[23] L. Zhang, H. Gao, J. Wen, S. Li, and Q. Liu, "A deep learning-based recognition method for degradation monitoring of ball screw with multi-sensor data fusion," *Microelectron. Rel.*, vol. 75, pp. 215–222, Aug. 2017.

[24] Z. Gao, C. Ma, D. Song, and Y. Liu, "Deep quantum inspired neural network with application to aircraft fuel system fault diagnosis," *Neurocomputing*, vol. 238, pp. 13–23, May 2017.

[25] H. Oh, J. H. Jung, B. C. Jeon, and B. D. Youn, "Scalable and unsupervised feature engineering using vibration-imaging and deep learning for rotor system diagnosis," *IEEE Trans. Ind. Electron.*, vol. 65, no. 4, pp. 3539–3549, Apr. 2018.

[26] J. Sun, C. Yan, and J. Wen, "Intelligent bearing fault diagnosis method combining compressed data acquisition and deep learning," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 1, pp. 185–195, Jan. 2018.

[27] M. Sohaib, C.-H. Kim, and J.-M. Kim, "A hybrid feature model and deep-learning-based bearing fault diagnosis," *Sensors*, vol. 17, no. 12, p. 2876, Dec. 2017.
[28] W. You, C. Shen, D. Wang, L. Chen, X. Jiang, and Z. Zhu, "An intelligent deep feature learning method with improved activation functions for machine fault diagnosis," *IEEE Access*, vol. 8, pp. 1975–1985, 2020.
[29] M. Xia, T. Li, L. Xu, L. Liu, and C. W. D. Silva, "Fault diagnosis for rotating machinery using multiple sensors and convolutional neural networks," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 1, pp. 101–110, Feb. 2018.
[30] O. Abdeljaber, O. Avci, M. S. Kiranyaz, B. Boashash, H. Sodano, and D. J. Inman, "1-D CNNs for structural damage detection: Verification on a structural health monitoring benchmark data," *Neurocomputing*, vol. 275, pp. 1308–1317, Jan. 2018.
[31] L. Guo, Y. Lei, S. Xing, T. Yan, and N. Li, "Deep convolutional transfer learning network: A new method for intelligent fault diagnosis of machines with unlabeled data," *IEEE Trans. Ind. Electron.*, vol. 66, no. 9, pp. 7316–7325, Sep. 2019.
[32] X. Li, Y. Hu, M. Li, and J. Zheng, "Fault diagnostics between different type of components: A transfer learning approach," *Appl. Soft Comput.*, vol. 86, Jan. 2020, Art. no. 105950.
[33] M. Yuan, Y. Wu, and L. Lin, "Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network," in *Proc. IEEE Int. Conf. Aircr. Utility Syst. (AUS)*, Oct. 2016, pp. 135–140.
[34] R. Zhao, J. Wang, R. Yan, and K. Mao, "Machine health monitoring with LSTM networks," in *Proc. 10th Int. Conf. Sens. Technol. (ICST)*, Nov. 2016, pp. 1–6.
[35] P. Malhotra, V. Tv, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, "Multi-sensor prognostics using an unsupervised health index based on LSTM encoder-decoder," 2016, *arXiv:1608.06154*. [Online]. Available: http://arxiv.org/abs/1608.06154
[36] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: http://arxiv.org/abs/1704.04861
[37] H. Pan, Z. Pang, Y. Wang, Y. Wang, and L. Chen, "A new image recognition and classification method combining transfer learning algorithm and MobileNet model for welding defects," *IEEE Access*, vol. 8, pp. 119951–119960, 2020.
[38] S.-Y. Lu, S.-H. Wang, and Y.-D. Zhang, "A classification method for brain MRI via MobileNet and feedforward network with random weights," *Pattern Recognit. Lett.*, vol. 140, pp. 252–260, Dec. 2020.
[39] M. Hu, H. Guo, and X. Ji, "Automatic driving of end-to-end convolutional neural network based on MobileNet-V2 migration learning," in *Proc. 12th Int. Symp. Vis. Inf. Commun. Interact.*, Sep. 2019, pp. 36:1–36:4.
[40] Y. K. Lal, V. Kumar, M. Dhar, M. Shrivastava, and P. Koehn, "De-mixing sentiment from code-mixed text," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics, Student Res. Workshop*, 2019, pp. 371–377.
[41] D. Smilkov, N. Thorat, Y. Assogba, A. Yuan, N. Kreeger, P. Yu, K. Zhang, E. Nielsen, D. Soergel, S. Bileschi, M. Terry, S. Cai, S. N. Gupta, S. Sirajuddin, D. Sculley, R. Monga, G. Corrado, F. B. Viégas, C. Nicholson, and M. Wattenberg, "TensorFlow.Js: Machine learning for the Web and beyond," 2019, *arXiv:1901.05350*. [Online]. Available: http://arxiv.org/abs/1901.05350

WENBING YU received the M.S. degree from the Department of Photoelectric Engineering, Huazhong University of Science and Technology, China, in 2004. He is currently a Senior Engineer with the School of Higher Vocational Technology, Shanghai Dianji University. He has presided and participated in four scientific research projects, including NSFC General Project and Hubei Natural Science Foundation. He published more than 20 academic articles, including SCI or EI. His current research interests include intelligent computing, photonic crystal fiber sensors, and periodically poled lithium niobate all-optical communication devices.

PIN LV received the M.S. and Ph.D. degrees from the School of Computer Science and Technology, Wuhan University of Technology, Wuhan, China, in 2001 and 2014, respectively. She is currently, a Professor with the School of Electronics Information Engineering, Shanghai Dianji University. Her current research interests include data mining, machine learning, sentiment analysis, and artificial intelligence.