ICIC Express Letters
Volume **16**, Number **4**, April **2022**

ICIC International ©2022 ISSN 1881-803X
pp. 371–379

KERAS-BASED 3D-CONVOLUTIONAL NEURAL NETWORK BEARING FAULT DIAGNOSIS

Xingyu Han[1], Zhihua Hu[1,*], Miao Huang[1] and Lang Wang[2]

[1]College of Intelligent Manufacturing and Control Engineering
Shanghai Polytechnic University
No. 2360, Jinhai Road, Pudong District, Shanghai 201209, P. R. China
han_xy9526@163.com; huangmiao@sspu.edu.cn; *Corresponding author: zhhu@sspu.edu.cn

[2]School of Information Science and Engineering
NingboTech University
No. 1, South Qianhu Road, Ningbo 315100, P. R. China
wanglang@nbt.edu.cn

Received August 2021; accepted October 2021

Abstract. *This paper proposes a 3D-convolutional neural network (3D-CNN) fault diagnosis algorithm based on the Keras framework for the diagnosis of rotor bearing output end faults. Feature extraction and diagnostic analysis can be carried out based on 3D data. The model takes the acceleration data collected by the 3D acceleration sensor as the input of the diagnosis model, completes the feature extraction by the 3D-CNN algorithm, then diagnoses the fault types and classifies the output by the convolutional neural network (CNN). The fault types are classified by CNN Softmax classifier. The experimental results show that the diagnostic accuracy using this fault diagnosis model achieves 99%. Unlike the traditional CNN algorithm, this method can put three data sets directly into the CNN as the input layer for operation, which eases the feature extraction of data sets. Compared with the traditional 1D and 2D CNN-based feature extraction fault diagnosis algorithms, the Keras-based 3D-CNN diagnostic model has higher accuracy for analyzing and diagnosing.*
Keywords: Machine learning, Deep learning, 3D-convolutional neural network, Fault diagnosis, Data processing

1. **Introduction.** Rolling bearings, as an integral part of today's industrial production [1], are widely used in the connecting structures of industrial machine [2]. They are important parts in lots of mechanical equipment, and act to friction reduction, load transfer, and lubrication [3]. According to relevant statistics, bearing failure occupies a large proportion of the many causes of failure in mechanical equipment [4-6]. The traditional detection methods can only be found when bearing fault causes equipment fault, and the fault caused by weak signal detection still needs machine learning to be involved [7]. Therefore, bearing fault diagnosis and analysis based on machine learning has great significance [8].

Nowadays, the existing methods for fault diagnosis in the field of machine learning are initially for data collected from a single sensor to diagnose the related faults [9], but these methods have bad robust performance and the diagnostic accuracy requires high data requirements [10,11]. Therefore, to overcome this drawback, many fault diagnosis algorithms based on machine learning aim at detecting multiple sensors simultaneously. The fault data is collected by different sensors, and then the collected data is diagnosed by feature fusion or other experimental methods [12]. However, this method leads to strong coupling of the data collected by different sensors, which in turn makes it difficult to determine the type of bearing faults accurately and efficiently [6].

DOI: 10.24507/icicel.16.04.371

Most of the current models for bearing fault analysis are still one-dimensional or two-dimensional convolutional neural networks (CNN). One-dimensional CNN is a machine learning fault diagnosis model for data collected by a single sensor. And two-dimensional CNN requires two-dimensional data as input for fault diagnosis. Two-dimensional data is obtained by adding a time dimension or by lifting one-dimensional data [13-15]. The sensors of most mechanical devices are not only single-dimensional signal data, but there is not much research on machine learning diagnosis model for multidimensional fault analysis model.

In summary, this paper uses 3D-CNN to establish a multidimensional machine learning fault analysis method. Firstly, the collected three data signals are pre-processed, and the original data are firstly processed by one-hot labeling algorithm. Secondly, 3D-CNN fault diagnosis analysis algorithm based on Keras framework is established, including convolutional layer, pooling layer, fully-connected layer, and output layer. Finally, the fault diagnosis model based on Kreas outputs the prediction accuracy and draws the loss curve according to the degree of data set deviating from its label. The 3D-CNN algorithm based on Keras has the following advantages. 1) It can directly put the preprocessed 3D data into the established model without dimension reduction of multidimensional data. 2) There is no need to extract low-dimensional independent variables from multidimensional coupling data. 3) Up to now, it is the most convenient to build 3D-CNN algorithm model under Keras framework.

2. **Problem Statement and Preliminaries.**

2.1. **Keras framework.** Keras is an open source artificial neural network library written in Python that can be used as a high-level application programming interface (API) for Tensorflow, computational network toolkit (CNTK), and Theano to design, evaluate, apply, and visualize deep learning models. To use Keras, you only need to import the relevant model interfaces and call the encapsulated functions. Compared to Tensorflow, it is easier to call central processing unit (CPU) or graphics processing unit (GPU) for model computation, and has more convenient function calling capabilities.

2.2. **3D-convolutional neural network.** CNN is a deep learning network framework with a wide range of applications in the field of machine learning. 3D convolution is performed by stacking multiple dimensions of data to form a cube and then applying 3D convolution kernels in the cube to performing the relevant operations.

The 3D-CNN is not very different from the traditional CNN in terms of architecture. It has the same structure of input layer, convolutional layer, pooling layer, fully connected layer and output layer, etc. The schematic diagram of the network structure is shown in Figure 1. However, the dimension of all layers is three-dimensional, the input layer is three-dimensional data, and the corresponding convolutional kernel is also a three-dimensional convolutional kernel.

The theoretical basis of 3D-CNN is not much different from that of CNN, and both of them perform convolutional operation on the input data and output the feature map by the convolutional kernel in the CNN. The mathematical expressions are as follows. Formally, the value of unit at position (x, y, z) in the jth feature map in the ith layer, denoted as v_{ij}^{xyz}, is given by

$$v_{ij}^{xyz} = \tanh\left(b_{ij} + \sum_{m}\sum_{p=0}^{P_i-1}\sum_{q=0}^{Q_i-1}\sum_{r=0}^{R_i-1} w_{ijm}^{pqr} v_{(i-1)m}^{(x+p)(y+q)(z+r)}\right)$$

where $\tanh(\cdot)$ is the hyperbolic tangent function, b_{ij} is the bias of this feature map, m refers to the index of the set of feature maps in the $(i-1)$th layer connected to the current

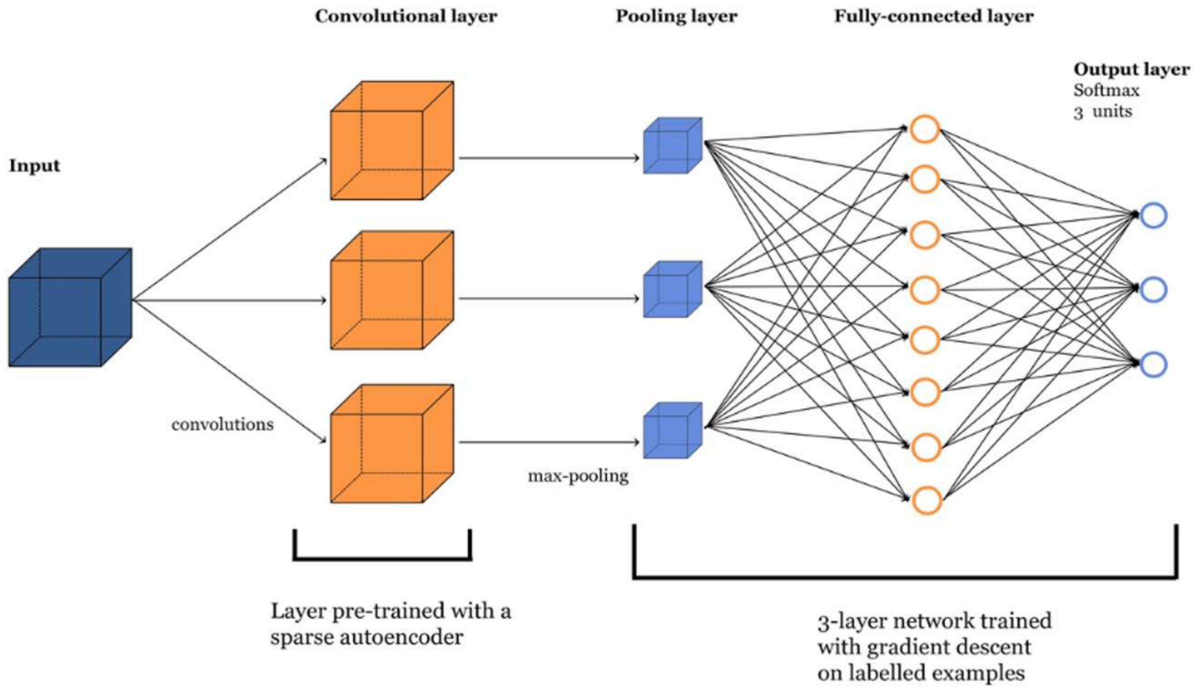


Figure 1. Schematic diagram of 3D-CNN

feature map, w_{ijm}^{pqr} is the value at the location (p, q, r) connected to the kth core, and P_i, Q_i, and R_i are the height, width, and length of the core, respectively.

Similar to the traditional CNN, the structural adjustments and the parameter settings of the machine learning layers can be chosen according to the model and data set.

3. **Keras-Based 3D-CNN Fault Diagnosis Model.** The implementation of 3D-CNN in Keras calls the keras.layers.convolutional.Conv3D function. Firstly, import the Conv3D, Dense, Maxpooling3D, Flatten, Dropout functions from keras.layers, and then the preprocessed bearing data is imported. Set the training parameters, such as the batch size, the training generations, the number of fault types or classifications, and the number of samples. The data are classified into a training set, a test set and a validation set by calling the functions in the preprocessing code with a preset scale, and then CNN functions such as Conv3D are called to build the 3D-CNN bearing fault detection model and model output training results.

The relevant parameters used in the Conv3D function are set and defined as follows.

Definition 3.1. *filters: The number of convolution kernels, i.e., the dimensionality of the output.*

Definition 3.2. *kernel_size: A single integer or a list of three integers as the length, width, and height of the convolution kernel. If it is a single integer, it means that the convolution kernel is of the same length in all spatial dimensions.*

Definition 3.3. *strides: A single integer or a list of three integers as the step size of the convolution. If it is a single integer, it means that the step length is the same in all spatial dimensions.*

Definition 3.4. *padding: Padding strategy, either valid or same-valid means that only valid convolution is performed, i.e., the boundary data is not processed. same-same means that the convolution result at the boundary is retained, which usually results in the output shape being the same as the input shape.*

Definition 3.5. *kernel_regularizer: Regularizer term imposed on the weights, for regularizer object.*

4. Experiments.

4.1. Data acquisition. The bearing failure data in this paper are collected from the HZXT-008 rotor bearing comprehensive failure simulation experiment platform, as shown in Figure 2. The experimental platform has the characteristics of simple structure, easy disassembly and assembly, and stable performance. It can be flexibly configured with sensors for vibration, speed, displacement, acceleration and other mechanical parameters measurement in order to measure the relevant data when the bearing is operating. It is equipped with data acquisition instruments and data analysis software to form a multi-purpose and comprehensive experimental system platform, providing a good experimental analysis environment for researchers engaged in the study of rotor dynamics and related courses.

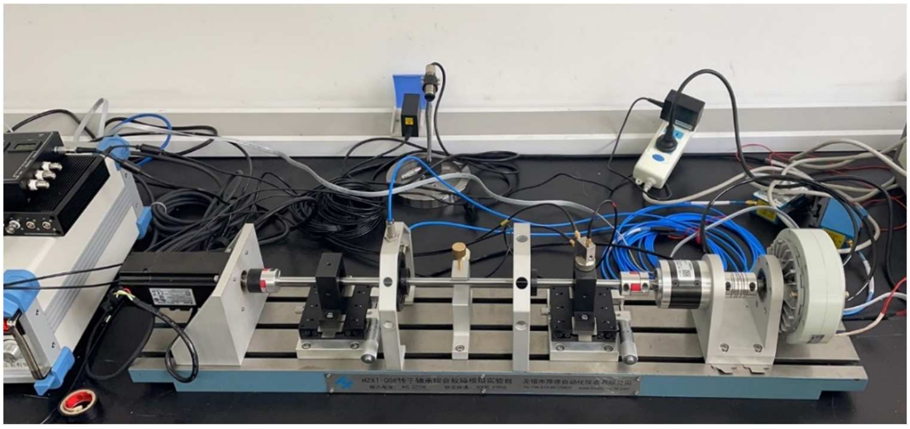
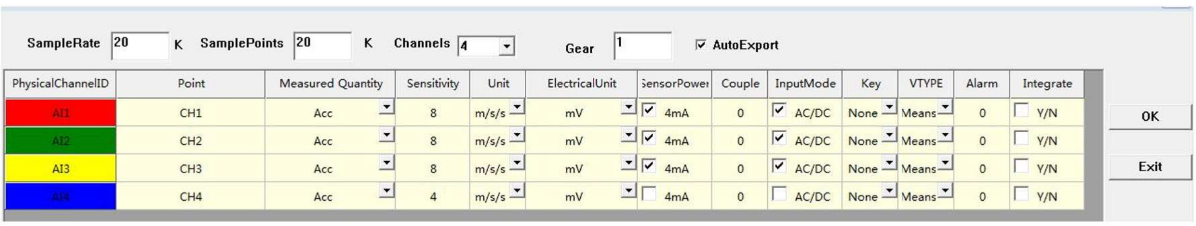


FIGURE 2. HZXT-008 rotor bearing fault simulation bench

The experimental data collected in this paper are mainly the experimental data measured by the three-axis acceleration sensor installed on the experimental platform, and the raw data sets of three different bearing faults were collected. The bearing faults are pitting outer ring faults, pitting inner ring faults, and pitting rolling faults. The relevant data were collected through the data acquisition equipped with the bearing experimental platform, and the data acquisition software was set up as shown in Figure 3. The bearing speed is 1300 rpm, the sampling frequency is 20 KHz, and the sample time for each fault is 10 min.



SampleRate 20 K SamplePoints 20 K Channels 4 Gear 1 AutoExport

PhysicalChannelID	Point	Measured Quantity	Sensitivity	Unit	ElectricalUnit	SensorPower	Couple	InputMode	Key	VTYPE	Alarm	Integrate
AI1	CH1	Acc	8	m/s/s	mV	4mA	0	AC/DC	None	Means	0	Y/N
AI2	CH2	Acc	8	m/s/s	mV	4mA	0	AC/DC	None	Means	0	Y/N
AI3	CH3	Acc	8	m/s/s	mV	4mA	0	AC/DC	None	Means	0	Y/N
AI4	CH4	Acc	4	m/s/s	mV	4mA	0	AC/DC	None	Means	0	Y/N

OK

Exit

FIGURE 3. Fault simulation software settings

The bearing fault type labels, bearing fault locations and bearing fault types are shown in Table 1; the sensor types, sensor mounting positions and sensor detection directions are shown in Table 2.

Table 1. Bearing failure types

Fault labels	Fault locations	Fault types
EA	Rolling body	Pitting
EB	Outer ring	Pitting
EC	Inner circle	Pitting

Table 2. Sensor types and mounting locations

Sensors	Sensor mounting position	Detection direction
Acceleration sensor	Faulty bearing connection	X
Acceleration sensor	Faulty bearing connection	Y
Acceleration sensor	Faulty bearing connection	Z

4.2. **Data processing.**

4.2.1. *Raw data set pre-processing.* The bearing failure data collection system is mentioned earlier, and the experimental data are collected through the experimental data collection software to support the experiment. The data collected by the experimental data collection software is saved as multiple csv files, and the data collected for different faults should be merged and sorted for pre-processing operations firstly. Figure 4 shows the result of the experimental original file merging and sorting operations. As shown in the figure, the experimental data of three different fault types are merged and organized into three different files, and each file has three dimensions of the experimental data.

```
F:\anaconda3\envs\keras\python.exe F:/PyCharm_Files/keras_bearing_fault_diagnosis-master
 /preprocess3D.py
416 files have been merged in fault tag EA, and the dimension of fault data is (8519684, 3)
298 files have been merged in fault tag EB, and the dimension of fault data is (6103044, 3)
351 files have been merged in fault tag EC, and the dimension of fault data is (7188484, 3)
```

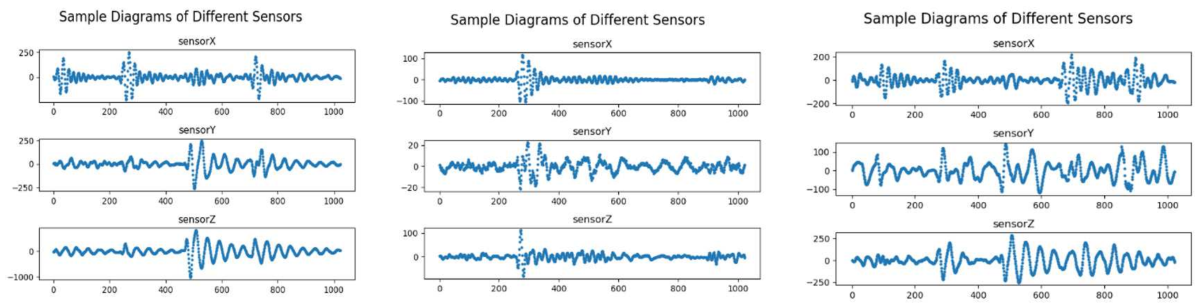
Figure 4. Pre-processing results of the original data of the fault

4.2.2. *Fault dataset pre-processing.* For the raw bearing failure data obtained by the above processes, the pre-processing functions are defined, such as partitioning the data set into training set, validation set, test set and labeling the different data sets by one-hot algorithm according to a certain proportion of functional functions. The defined functions are verified by importing the bearing failure datasets.

The partial visualization of the collected dataset is shown in Figure 5. Figure 5(a) represents the three-dimensional dataset of the pitting rolling body failure part, Figure 5(b) represents the three-dimensional dataset of the pitting inner ring failure part, and Figure 5(c) represents the three-dimensional dataset of the pitting outer ring failure part with different colors indicating different dimensional data.

4.3. **Experimental procedure.**

4.3.1. *Experimental steps.* Firstly, set the training parameters: the batch size is 128, the number of iterations is 15, the classification target is 3, each class of fault samples is 2000, each sample dimension is $(1024, 3)$, and the proportion of training set, test set and verification set is $(0.7, 0.2, 0.1)$. Then import the relevant functions in the preprocessing file, divide the failure data set according to the ratio, import the relevant machine learning functions of Keras framework, introduce two 3D convolutional layers, a fully connected layer, an output layer and set the parameters of function and layer. After that, we



(a) Pitting rolling body (b) Pitting inner circle partial (c) Pitting outer ring part

FIGURE 5. Partial data visualization chart

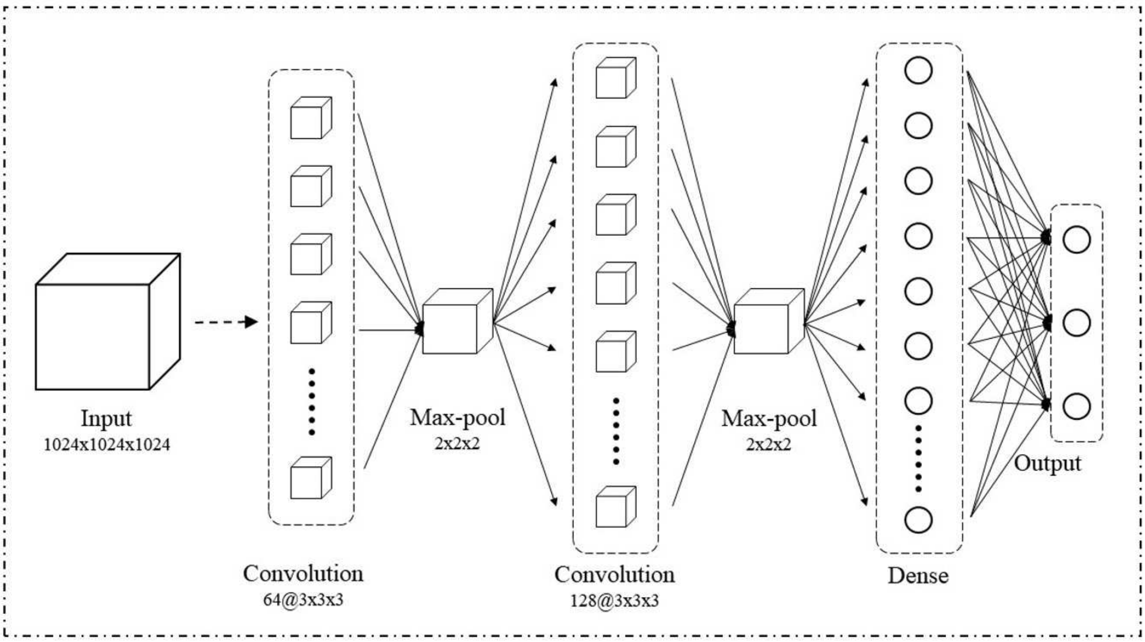


FIGURE 6. 3D-CNN algorithm model flow chart

compile the model, train it, and output the experimental results. The flow chart of 3D-CNN algorithm model is shown in Figure 6 and the experimental procedure is shown in Figure 7.

4.3.2. *Experimental results.* The experimental results are calculated by importing the relevant evaluation functions under the Keras framework. The experimental results are shown in Table 3.

It is proved that 3D-CNN has excellent prediction and classification effect for 3D data, and the accuracy rate on the test set reaches 99.16%. Loss curve is a measure of machine learning model, used to measure the prediction of the model deviate from its label. The loss curve of the model is shown in Figure 8.

Experimental results of 3D-CNN method compared with that of traditional CNNs are shown in Table 4. The traditional CNNs use the same dataset, but the data input is one dimension of the three-dimensional fault dataset or two dimension of the three-dimensional fault dataset, so there are three different prediction accuracy.

3D-CNN method has the highest prediction accuracy than CNNs with one-dimensional data input and two-dimensional data input. The prediction accuracy of traditional CNN is 76.67%, 84.5%, 83.16%, the prediction accuracy of 2D-CNN is 73.16%, 91%, 91.17% and the prediction accuracy of 3D-CNN is 99.16%.

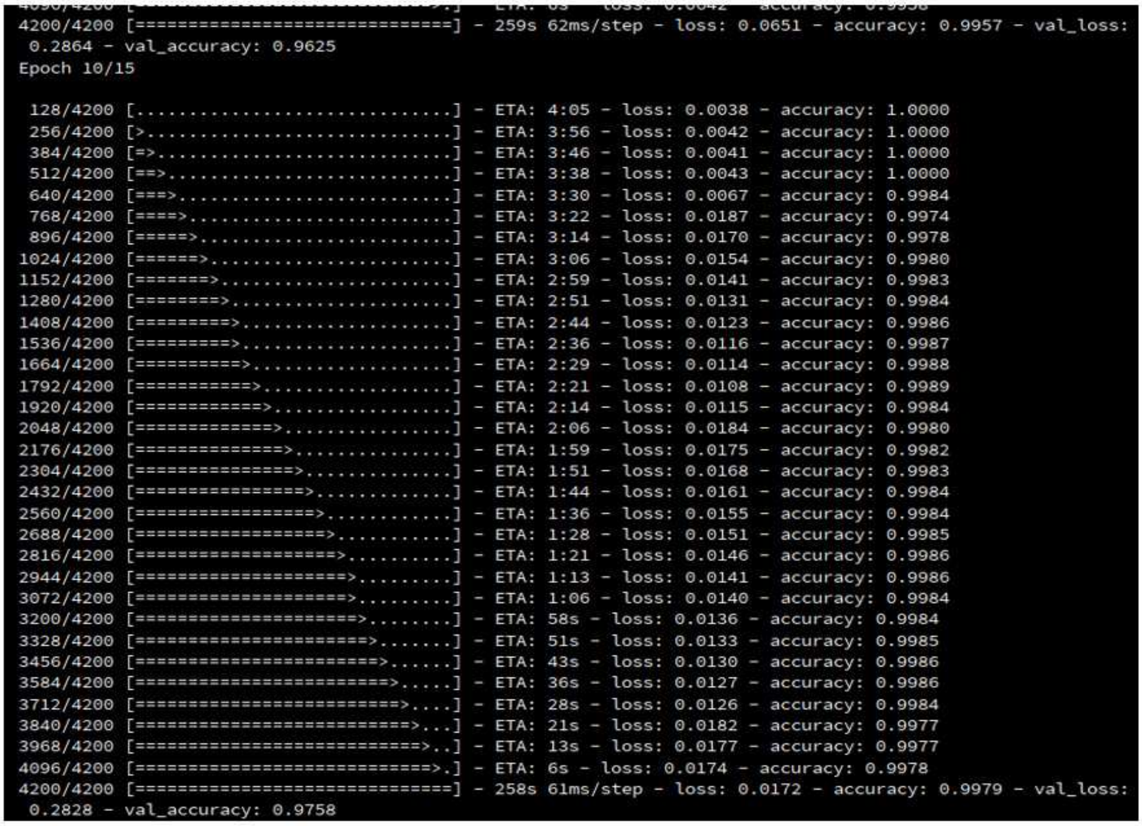


FIGURE 7. Model training process diagram

TABLE 3. Model training result

Validation with test sets	Validation results
Loss rate	6.76%
Accuracy	99.16%

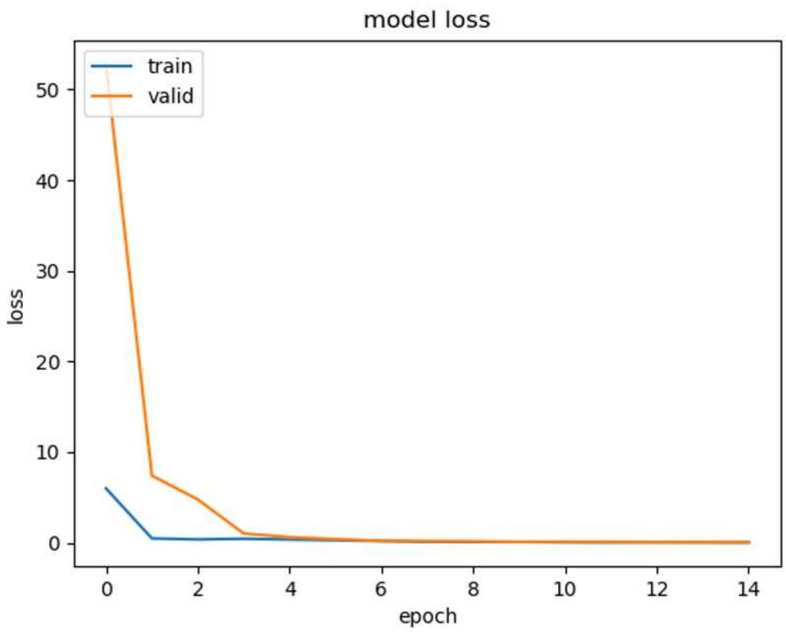


FIGURE 8. Model training loss graph

Table 4. Prediction accuracy of different machine learning models

Machine learning models	**Model prediction accuracy**		
CNN	X	Y	Z
	76.67%	84.5%	83.16%
2D-CNN	X&Y	Y&Z	X&Z
	73.16%	91%	91.17%
3D-CNN		99.16%	

5. **Conclusion.** In this paper, a 3D-CNN fault diagnosis algorithm based on Keras framework is proposed to analyze the output faults of rotor bearings. The experimental results show that 3D-CNN method has higher prediction accuracy than traditional CNN on multidimensional data sets. Multidimensional data sets can be operated and trained simultaneously. There is no need to predict and analyze single-dimensional data like traditional CNN. 3D-CNN method is a convenient and efficient algorithm on multi-dimensional data sets. 3D-CNN method is an extension of CNN algorithm. It is proved that 3D-CNN has excellent prediction and classification effect for 3D data.

Future plans include the implementation of more tests, covering more types of bearing tests under various working conditions and fault conditions. These tests will be used to further understand the limitations and boundaries of the proposed method in fault diagnosis. In addition, the computational efficiency of this method needs to be improved in the training process.

Acknowledgment. The work is supported by the National Natural Science Foundation of China under Grant 62003205, Zhejiang Provincial Natural Science Foundation of China under Grant LQ19F030005; Natural Science Foundation of Ningbo under Grant 2019A610092; Shanghai Natural Science Foundation under Grant 20ZR1421000.

REFERENCES

[1] M. B. Sun, Q. L. Ma, J. F. Lei and Y. L. Zhang, Research on fault diagnosis of gearbox based on HGWO-MSVM, *Combined Machine Tools and Automatic Machining Technology*, no.5, pp.95-98+102, 2018.

[2] R. Chen, J. Li, T. Shang and J. Zhang, Improved fireworks algorithm and probabilistic neural network for intelligent diagnosis of gearbox faults, *Journal of Agricultural Engineering*, vol.34, no.17, pp.192-198, 2018.

[3] J. Yang, A BP neural network method for automotive gearbox fault diagnosis and experimental validation, *Mechanical Transmission*, vol.43, no.1, pp.150-153, 2019.

[4] C. Zhou, J. Ma, J. Wu and X. Yuan, An adaptive VMD method based on improved GOA to extract early fault feature of rolling bearings, *International Journal of Innovative Computing, Information and Control*, vol.15, no.4, pp.1485-1505, 2019.

[5] Q. Zhang, *A Machine Learning Method for Fault Diagnosis of a Class of Shaft System Equipment*, Master Thesis, Henan University of Technology, 2020.

[6] J. Han, *Design and Implementation of a Random Forest and LightGBM Based Device Fault Detection Method*, Master Thesis, University of Chinese Academy of Sciences (Shenyang Institute of Computing Technology, Chinese Academy of Sciences), 2020.

[7] Y. Xie, *Research on Machine Learning-Based Fault Diagnosis of Machine Pumps*, Master Thesis, Xi'an University of Science and Technology, 2020.

[8] H. Sun, Z. Li, G. Sheng and X. Jiang, Research on discharge sound recognition based on machine learning and convolutional neural network, *High Voltage Apparatus*, vol.56, no.9, pp.107-113, DOI: 10.13296/j.1001-1609.hva.2020.09.016, 2020.

[9] X. Zhang and Y. Xie, Application of support vector machine parameter optimization method in gearbox fault diagnosis, *Ship Electronics Engineering*, vol.40, no.9, pp.146-149, 2020.

[10] S.-Q. Li and C.-J. Jiang, EEMD-CNN based rolling bearing fault diagnosis method, *Mechanical Strength*, vol.42, no.5, pp.1033-1038, 2020.

[11] L. Quan and Q. Che, Artificial intelligence-based short-circuit fault detection method for ship generators, *Ship Science and Technology*, vol.42, no.20, pp.61-63, 2020.

[12] N. Zhu, Design of electrical equipment fault diagnosis system based on machine learning, *Automation Applications*, no.10, pp.16-17, 2020.
[13] J. Gao, H. Han, Z. Ren and Y. Fan, Fault diagnosis for building chillers based on data self-production and deep convolutional neural network, *Journal of Building Engineering*, vol.34, DOI: 10.1016/j.jobe.2020.102043, 2021.
[14] P. Chen, Y. Li, K. Wang and M. J. Zuo, An automatic speed adaption neural network model for planetary gearbox fault diagnosis, *Measurement*, vol.171, DOI: 10.1016/j.measurement.2020.108784, 2021.
[15] Y. Zhang, T. Zhou, X. Huang, L. Cao and Q. Zhou, Fault diagnosis of rotating machinery based on recurrent neural networks, *Measurement*, vol.171, DOI: 10.1016/j.measurement.2020.108774, 2021.