Journal of the Brazilian Society of Mechanical Sciences and Engineering (2023) 45:644
https://doi.org/10.1007/s40430-023-04537-8

TECHNICAL PAPER

A new deep learning model combining CNN for engine fault diagnosis

Eyup Sonmez[1] · **Sezgin Kacar**[2] · **Suleyman Uzun**[3]

Received: 23 April 2023 / Accepted: 11 October 2023 / Published online: 24 November 2023

Abstract

Real-time condition monitoring of electric motors and early diagnosis is of great importance for ensuring safe and reliable operation, preventing major accidents, and reducing production costs. Therefore, many intelligent fault diagnosis methods have been proposed. However, in industrial applications, the constantly changing loads of electric motors and the inevitable noise from the working environment cause a decrease in the performance of intelligent fault diagnosis methods. In this study, an effective and reliable deep learning model named the Combined One and Two-Dimensional Deep Convolutional Neural Network with Wide First-layer Kernels (WDD-CNN) is proposed for real-time condition monitoring and early fault diagnosis under noisy and changing operating conditions. The primary contribution of this study is the development of a fault diagnosis method that can operate in real-time to provide early detection of faults that may occur in electrical drive systems under operating conditions that are unpredictable and noisy. In addition, the proposed model works directly on raw signals, eliminating the complexity of preprocessing processes. The Case Western Reserve University (CWRU) dataset is used to test the performance and effectiveness of the proposed WDD-CNN model under different load conditions and for noise suppression. Additionally, the effectiveness of the model against data coming from a single sensor channel is also tested, and the results are recorded. The proposed method achieves 100% accuracy when tested with normal signals. Comparative results reveal that the WDD-CNN model outperforms other current state-of-the-art methods with an accuracy rate of 96.45% under different operating loads.

Keywords Intelligent fault diagnosis · Convolutional neural network · Raw signals · Dual pathway · Different operating conditions · Noisy environment

1 Introduction

Faults in electric motors are inevitable, and a single fault, such as a short or open circuit in one of the motor windings, can lead to significant problems. These problems encompass poor performance, unstable operation, increased noise and vibration, and unwanted torque fluctuations [1, 2]. Such problems can result in substantial costs and even pose risks to the environment and human safety [3]. Reliable and robust diagnostics of electric powertrains early on are essential to overcome such issues and ensure efficient, safe, and reliable operation [4]. Various diagnostic methods are available in the literature, including motor current signature analysis, mechanical vibration analysis, temperature measurement, infrared recognition, and chemical analysis [5]. These methods involve measuring signals such as stator currents, external magnetic flux densities, rotor position and speed, output torque, temperature, and case vibrations [6].

Traditional motor fault detection and diagnosis methods rely on the knowledge and experience of experts. However, data-driven artificial intelligence-based approaches are increasingly replacing expert experience. These approaches aim to reduce the periodic maintenance cycle of electric motors, widely used across diverse fields, and enhance the accuracy of fault diagnosis [7]. Data-driven intelligent fault

Technical Editor: Jarir Mahfoud.

Eyup Sonmez
d175045051@subu.edu.tr

[1] Department of Mechatronics Engineering, Faculty of Technology, Sakarya University of Applied Sciences, 54187 Sakarya, Turkey

[2] Department of Electrical and Electronics Engineering, Faculty of Technology, Sakarya University of Applied Sciences, 54187 Sakarya, Turkey

[3] Department of Computer Engineering, Faculty of Technology, Sakarya University of Applied Sciences, 54187 Sakarya, Turkey

diagnosis focuses on creating diagnostic models capable of automatically establishing relationships between collected data and engine robustness. In the data-driven intelligent diagnostic method, vibration data and current signals are widely used for motor fault diagnosis [8, 9]. These signals provide valuable information about the operational state of motors, allowing for the detection of various types of faults, including mechanical wear, electrical issues, and bearing failures. By analyzing the patterns and anomalies in these signals, machine learning and deep learning algorithms can effectively identify and classify motor faults, thereby enabling predictive maintenance and reducing unplanned downtime in industrial systems.

In the data-driven intelligent fault diagnosis process, traditional machine learning approaches consist of three main stages: feature extraction, feature selection and classification. Before the classification stage, features from both the frequency and time domains can be extracted separately from the signals collected by the sensors. Then, for these extracted features, the feature selection should be performed in the dimension reduction stage. These selected features are used as inputs for fault diagnosis through machine learning methods such as artificial neural networks (ANN) [10, 11], support vector machines (SVM) [12, 13] and k-nearest neighbor (k-NNR) [14, 15], classifiers.

Recent advancements in artificial intelligence and machine learning have led to the development of numerous data-driven intelligent diagnosis algorithms [16–19]. These algorithms enable the automatic identification of motor fault types and severities based on input data and training datasets. Asr et al. [20] proposed diagnosis of a new combined faults based on the Non-Naive Bayesian (NNBC) classifier, in which features are extracted from the resampled multi component signals using Empirical Mode Decomposition (EMD), and these features are used for intelligent fault diagnosis of the rotary mechanism. Georgoulas et al. [21] used the Symbolic Aggregate approXimation (SAX) framework to extract features from vibration signals and then, classify the faults using the k-NNR classifier. Piltan and Kim [22] proposed an intelligent digital twin integrated method based on a support vector machine for bearing anomaly detection and crack size identification and tested the impact of their proposed model with the Case Western Reserve University bearing dataset. Shahbaz and Amin [23] proposed a new hybrid fault-tolerant control system (HFTCS) with custom nonlinear controllers and used ANN for estimation of faulty sensor values in the observer model. Shifat and Hur [24] proposed a multiple sensor data fusion method combining vibration and current signals based on Principal Components Analysis (PCA) and ANN for reliable multi-fault diagnosis framework of a brushless DC (BLDC) motor.

Although intelligent fault diagnosis based on traditional machine learning methods has been widely used for years, these approaches have some disadvantages such as the time-consuming process of feature extraction and the requirement of expert experience [25].

Most of the data-driven intelligent diagnostic techniques in the literature have difficulty adapting to different operating conditions such as different motor loads and environmental noise. However, the performance of deep learning methods has been shown to be better than other techniques [26]. One of the most important advantages of applying deep learning methods is that, instead of manual feature extraction in traditional machine learning methods, features are automatically extracted from the raw signals collected from the sensors and motor fault diagnosis and classification can be made using these features. The ability of deep learning algorithms to automatically extract features from raw data removes complex feature engineering stages, but also makes these algorithms more suitable for use in many fields.

In particular, a lot of research has focused on deep learning methods due to the advantage of automatically processing raw signals, eliminating the need for time-consuming manual feature extraction. Deep learning methods eliminate the disadvantages of traditional machine learning with the ability to learn end-to-end by taking the raw signals collected from the sensors as input. Therefore, deep learning methods such as Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM) and Autoencoders (EA) architectures are used by many researchers today for intelligent fault detection. Zhang et al. [27] proposed an intelligent fault diagnosis method based on Convolutional Neural Network that converts raw signals into two-dimensional images and automatically extracts features from these images. They analyzed the effect of different sizes of input data and different load conditions on the fault diagnosis capability of this method. Shao et al. [28] used two-dimensional grayscale images as input vector for fault diagnosis. They obtained grayscale images by transforming current and vibration signals into time–frequency distribution (TFD) with wavelet transform and then, applying them to two separate deep convolutional neural networks, discussed the results. Wang et al. [29] proposed a multi-signal diagnostic network based on 1D-CNN model for permanent magnet synchronous motors (PMSM). The advantage of the proposed method is that feature extraction modules can extract multi-scale features from complex conditions. In addition, they proposed a class feature map to automatically determine the frequency component. With the experimental results, they tried to reveal whether the proposed model could effectively diagnose three different engine states—healthy state, demagnetization fault state, and bearing fault state. Wang et al. [30] proposed a multi-resolution and multi-sensor fusion network (MRSFN) consisting of a 1D CNN and LSTM for motor fault diagnosis. They discussed both the feature extraction efficiency of the proposed architecture and the extent to which it adapts to varying motor speed. Zhang et al. [31] proposed a method named Deep Convolutional Neural Networks with Wide First-layer (WDCNN) processing directly on raw temporal signals without the need for complex preprocessing, to address the fault diagnosis problem. In their method, they addressed the classification accuracy against noise and working load changes using wide kernels in the first convolution layer for characterizing the input signal and suppressing high-frequency noise. Zhuang et al. [32] proposed an intelligent diagnostic method named stacked residual dilated convolutional neural network (SRDCNN) with automatic feature extraction and high performance in noisy environments. The proposed method offers a deep learning framework that integrates Dilated Convolution, Input Gate Structure of LSTM, and Residual Network. Zhang et al. [33] developed a method named Neural Networks with Training Interference (TICNN) for solving the problem of bad performance caused by different working loads and environmental noises. They added with the additive white Gaussian noise to the original signals and used six convolution and pooling layers to overcome the problem. They also configured the first-layer kernel to be 64×1 wide in order to better filter out the noise. They used the CWRU database as the data set to test the proposed model, and they achieved an average accuracy of 96.1%. Jia et al. [34] presented the Deep Normalized Convolutional Neural Networks (DN-CNN) model to overcome the problem that classical CNNs have low accuracy rates with few data, and it has been shown that the model has high performance with few data. They also proposed the neuron activation maximization (NAM) algorithm to better understand the feature learning process. Karim et al. [35] aimed to enhance the performance of the Fully Convolutional Neural Networks (FCN) network by augmenting the FCN with their proposed Long Short-Term Memory Fully Convolutional Network (LSTM-FCN) and Attention LSTM-FCN methods. The proposed model has been tested with the University of California Riverside (UCR) Benchmark datasets [36]. The model has been compared with different algorithms and it has been shown that the performance has improved significantly. Zhang et al. [37] presented an enhanced CNN model that uses time–frequency images as inputs for bearing fault diagnosis. With the proposed model, short-time Fourier transform theory and the scaled exponential linear unit (SELU) function are introduced for overcoming problems such as the inability of single time or frequency domain analysis methods to extract features effectively and the ReLU function greatly affected by the learning rate. Shenfield and Howarth [38] proposed dual-path neural network with a wide first-kernel capable of operating on raw temporal signals by adding a recurrent neural network (RNN)-based pathway in parallel to the deep convolutional neural network pathway. The performance of the proposed model revealed by testing with data acquired both under different operating conditions and under noisy conditions.

The studies mentioned above include different feature extraction methods and methods using different variations of RNN-CNN architectures in order to increase the performance in classification. However, when these studies are examined, it is understood that they have disadvantages such as high computational cost, tendency to incorrectly model feature dependencies, loss of the continuous nature of time series data, low accuracy rates, and model complexity. In our study, unlike the mentioned studies, a new model with faster speed and higher accuracy is proposed, which combines the advantages of 1D-CNN and 2D-CNN architectures. Thanks to our proposed model, spatial and temporal features are used together. We show that the proposed model performs well directly on raw signals and under different operating conditions, without the need for manual feature extraction and noise filtering stages. In Table 1, the traditional machine learning and other deep learning methods mentioned in the literature are compared with the method we propose.

The architecture of our proposed model is inspired by the “dual-path recurrent neural network with a wide first-kernel and deep convolutional neural network pathway (RNN-WDCNN)” proposed by Shenfield and Howarth [38]. There are also different models that combine RNN and CNN architectures [35]. However, the RNN architectures used in these models require more computational resources and time due to their large and complex nature. Therefore, they require higher computational resources or longer training times when working with large datasets or complex models. In addition, RNN models are more complicated and have shortcomings in capturing spatio-temporal features [39]. On the other hand, two-dimensional convolutional neural network (2D-CNN) architectures are very effective in capturing spatial relationships of features. 2D-CNNs are equipped with convolution and pooling layers and use these layers to extract features from image data and reduce their complexity. 2D-CNNs are very effective in learning local features through convolution layers. By traversing different regions of the data, convolution layers capture the spatial structure of features. In addition, the computational complexity of one-dimensional convolutional neural network (1D-CNN) architectures is low. Thanks to its parallel computing capabilities, it can be trained quickly, and the prediction time is short. Convolution operations can be performed in parallel, making it able to work quickly even with large datasets or complex models [40].

In this study, we propose an effective and reliable novel deep learning model, which we call WDD-CNN, that can operate in noisy and variable operating conditions for real-time condition monitoring and early fault detection. The

Table 1 Comparison of traditional machine learning and other deep learning methods with our proposed method

	References	Algorithm	Purpose	Object	Highlight	Disadvantages
Traditional machine learning methods	[20]	NNBC	Fault diagnosis	Rotary machinery	Diagnosing combined faults using single fault state features as training data	Use of hand-crafted feature extraction
	[21]	k-NNR and SAX	Fault diagnosis	Rolling element bearings	Using Symbolic Sum Approximation (SAX) for feature extraction and bearing fault diagnosis	Use of hand-crafted feature extraction Difficulty in applying symbolic representation to more complex or natural data types Risk of losing complex features due to symbols
	[22]	SVM, Kalman fitler, The intelligent digital twin	Fault diagnosis	Rolling element bearings	Feature extraction from residual signals via the intelligent digital twin	Use of hand-crafted feature extraction Model complexity
	[23]	ANN	Fault detection	Internal combustion engine	Optimizing post-fault performance	Use of hand-crafted feature extraction Model complexity
	[24]	ANN and PCA	Fault diagnosis	Electrical motor	Presenting a fault detection framework considering multiple faults	Use of hand-crafted feature extraction High computational cost
Deep learning methods	[27]	CNN	Fault diagnosis	Electrical motor	A data preprocessing method that converts the vibration signal into two-dimensional grayscale images	Limited accuracy and performance in varying load conditions
	[28]	CNN	Fault diagnosis	Electrical motor	Fault diagnosis with multi-signal model	High computational cost
	[29]	CNN	Fault diagnosis	Rolling element bearings, Electrical motor	Diagnosing motor faults by extracting the stator current signal and torque signal of the motor	Model complexity
	[30]	CNN LSTM	Fault diagnosis	Electrical motor	A multi-level information fusion model for fault diagnosis	Model complexity
	[31]	CNN	Fault diagnosis	Rolling element bearings	Proposing a learning framework that works directly on raw temporal signals	Limited accuracy and performance in varying speed conditions
	[32]	CNN LSTM	Fault diagnosis	Rolling element bearings	Stacked residual dilated convolutional neural network for fault diagnosis	Model complexity

Table 1 (continued)

References	Algorithm	Purpose	Object	Highlight	Disadvantages
[33]	CNN	Fault diagnosis	Rolling element bearings	An end-to-end method that takes raw temporal signals as inputs for fault diagnosis	The need to improve performance and accuracy
[34]	CNN	Fault classification	Rolling element bearings	A framework for imbalanced fault classification of machinery	Lack of generalization to different load conditions
[35]	CNN LSTM	Time series classification	Time series	Optimizing fault diagnosis performance	High computational cost Lack of accuracy generalization
[37]	CNN STFT	Fault diagnosis	Rolling element bearings	Presenting a CNN model using time–frequency images as inputs for fault diagnosis	High computational cost Lack of generalization to different load conditions
[38]	CNN LSTM	Fault diagnosis	Rolling element bearings	Providing early detection of developing problems under variable operating conditions	High computational cost The need to improve performance and accuracy

proposed WDD-CNN model is a model that combines the advantages a one-dimensional and two-dimensional deep convolutional neural network pathway and has a wide first-kernel layer. The main contributions of this study are:

1. We propose a new deep learning model with a first wide kernel layer that combines one and two-dimensional convolutional neural network pathway.
2. With the model we propose, we use both spatial and temporal features and show its effect on performance.
3. We show that the proposed model performs well directly on raw signals without the need for manual feature extraction and noise filtering stages.
4. We show that the proposed model also performs well under different operating conditions.
5. We show that it achieves higher performance by comparing our model with different fault diagnosis models in the literature.

2 A brief introduction to convolutional neural networks

CNNs are an important type of deep neural networks that have been successfully applied to various classification problems [41]. CNNs consist of multiple filtering stages and classification stages that do feature extraction automatically. The filtering stage includes layers such as the convolution layer, nonlinear activation layer, batch normalization and pooling layer. The classification stage is a multilayer perceptron consisting of several fully connected (Fc) layers. Figure 1 shows an example of a simple CNN architecture.

CNNs are a class of deep neural networks that can extract and classify certain features from images and are widely used for analyzing visual images. These deep neural networks are very useful as it minimizes the need for expert experience by automatically detecting the features. CNNs have been originally proposed for two-dimensional image processing and very good results have been obtained with CNNs in real-world applications in this field [42–44]. However, one-dimensional convolutional neural networks are also used effectively in the fields of automatic speech recognition, document classification, machine translation, and classification of one-dimensional data such as time series [45, 46].

The basic functions in the layers of convolutional neural networks can be listed as follows:

- *The input layer* is the first layer of convolutional neural networks, and the input data is raw data without any preprocessing.

- *In the convolution layer*, the input data is convoluted with a set of weights called a convolutional filter, and high-level features (such as edges and curves in an image) are extracted from these input signals by applying a nonlinear activation function.
- *The pooling layer* acts as a down sampling operation that reduces the number of the extracted features and eliminates the redundant details. Thus, with the decrease in the number of parameters and features, the computational load is reduced.
- *The classification layer* is a multilayer perceptron consisting of several fully connected layers where the classification is made. The features must be prepared by converting to one-dimensional vectors with the Flatting layer before the fully connected layers.

As the input signals progress through the convolutional stages, the network learns more detailed features. Therefore, layers consist of more than one block that follows each other. Batch Normalization (BN) layer is used to reduce the shift of the internal covariance and speed up the training process of the deep neural network [47]. The BN layer is usually added just after the convolution layer or the fully connected layer and before the activation unit. The Dropout layer, on the other hand, is used to prevent overfitting in deep neural networks. The Dropout layer helps prevent overfitting by randomly setting the input units to 0 with a frequency of rate at each step during the training time, ultimately improving its generalization performance on unseen data.

3 Proposed WDD-CNN method for fault diagnosis

The proposed WDD-CNN model for real-time condition monitoring and early fault diagnosis is tested with the CWRU dataset, and the classification performance of the model is measured. The flow diagram of the model is shown in Fig. 2. Here, after the raw time series has been split into training and test data, the training data are subjected to data augmentation. Then, both training and test data are converted into one- and two-dimensional training samples. After the model is trained with the training samples, the test data are subjected to the classification process and the model is evaluated.

3.1 The WDD-CNN model architecture

Researchers are still working to improve the fault diagnosis method of occurring faults in electric motors. In recent years, there has been an increasing interest in methods based on deep learning in fault diagnosis. These methods are generally categorized from two aspects: those that reshape one-dimensional input signals into two dimensions, such as time series data, and methods that operate on one-dimensional signals [41]. Fault classification is widely studied subfield in convolutional network-based fault diagnosis inspired by image classification. In image classification, 2D-CNN is widely used in fault diagnosis because it can readily exploit spatial patterns in common imagery. In addition, some advantages of 2D-CNN architecture for time series data can be listed as follows:

1. Capturing Spatial-Temporary Relationships: While time series data are usually represented in one dimension, 2D-CNN architecture is effective in capturing spatial–temporal relationships to understand the changes and

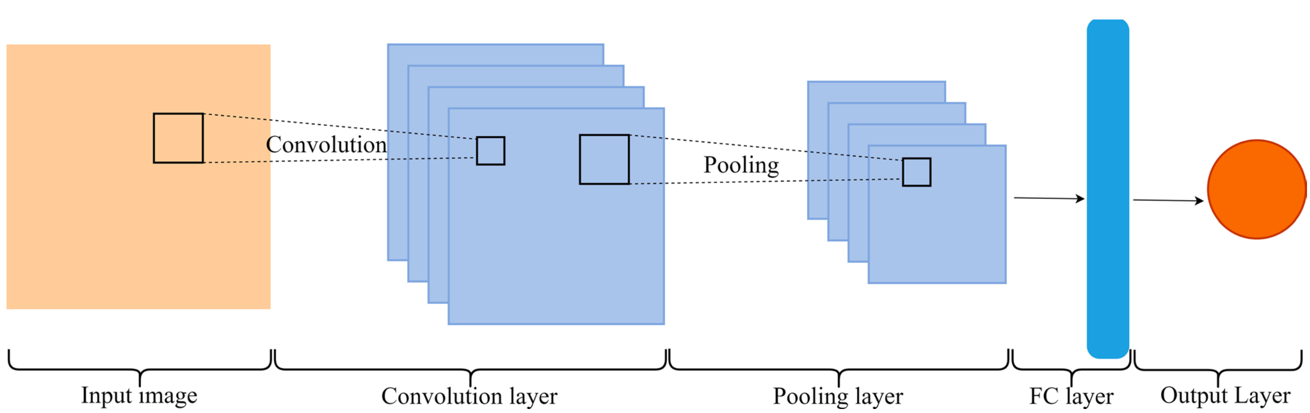


Fig. 1 Representation of a simple convolutional neural network (CNN)

patterns of time series data over time. Convolution layers capture patterns at different time stages and spatially analyze the features of the time series.

2. 2.Parallel Processing Capability: 2D-CNN architecture provides fast processing of time series data thanks to its parallel processing capability. Convolution operations can be performed in parallel, allowing it to work quickly even with large datasets or complex models.
3. Extraction of Feature Maps: 2D-CNN extracts important features from time series data and creates feature maps representing these features. Feature maps represent features at different time points and capture dynamics and relationships across time series.
4. Learning of Hierarchical Features: 2D-CNN models provide learning of hierarchical features in time series data. Lower layers can extract lower-level features (for example, edges), while upper layers can extract more complex features (for example, patterns). This provides a higher level of representation of time series data.
5. Reduction of Data Preprocessing Requirements: 2D-CNN architecture can reduce some data preprocessing stages. Instead of traditional methods (for example, Fourier transform) to obtain features, 2D-CNN can learn features directly.

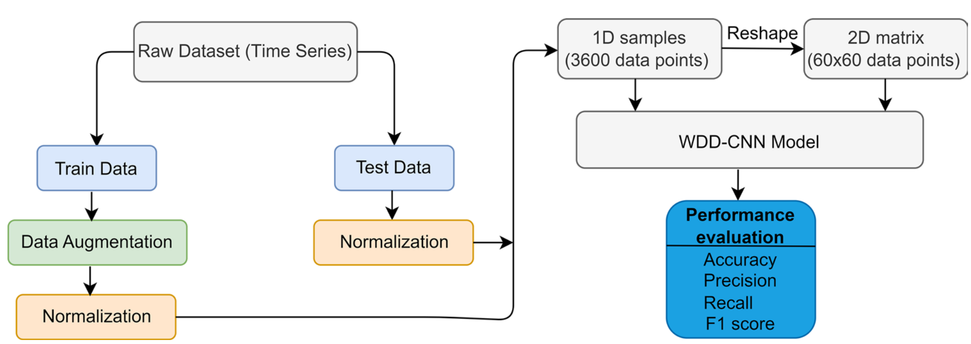
On the other hand, 1D-CNN models have many advantages when working effectively on 1D data structures such as time series data. Some of the advantages of 1D CNN can be listed as follows:

1. Local and Global Feature Extraction: 1D-CNN can extract local and global features at different scales effectively. Convolutional layers define local features as they hover over different regions of the data and then, obtain global features by combining these local features.
2. 2.Scaling Invariance: 1D-CNN models have scaling invariance. This reduces the sensitivity of the data to scale changes. For example, when the signal samples in the time series data have scale changes (expansion or compression), the 1D-CNN model can still capture the same features.
3. Reduced Number of Parameters: 1D-CNN models have fewer parameters compared to traditional RNN models. This provides faster learning of the model and less computational cost. It can also perform well even on smaller datasets.
4. Fast Training and Prediction Time: 1D-CNN can be trained quickly thanks to its parallel computational capabilities, and the estimation time is short. Convolution operations can be performed in parallel, allowing it to work quickly even with large datasets or complex models.
5. Feature Auto-Learning: Instead of manually defining features, 1D-CNN models can automatically discover features on the dataset.
6. 6.Overfitting Resistance: 1D-CNN models are generally more resistant to overfitting. Convolution layers capture the local structures of the data, thus increasing the generalization ability of the model. It can also be used to control overfitting with methods such as dropout and regularization.

In this study, we present a new dual-pathway WDD-CNN model combining 1D-CNN and 2D-CNN network for real-time condition monitoring and early fault diagnosis of electric motors under noisy and changing operating conditions by integrating the advantages of the two approaches aforementioned.

Each of the convolution blocks in both pathways (1D-CNN and 2D-CNN pathways) in the model architecture shown in Fig. 3 has a set of convolutional filters to extract meaningful features from the raw input signals. An activation function (ReLU) has been added to each block to include nonlinear features into the network. Additionally, there is a BN block for reducing internal covariance shift and improving the training performance by normalizing layer inputs between groups, and a max pooling block for accelerating the training process by reducing computational complexity. In the proposed model, in the first convolutional stage in both pathways, a much wider filter kernel is used for both suppressing high-frequency noise in the input signals and capturing distant dependencies. In other convolution

Fig. 2 Flow diagram of the fault diagnosis process



stages, smaller sized kernels are used for obtaining more useful features in the input signals and improving the performance of the network.

After the convolution layers, the data are prepared by converting it to a single array in the flattening layer and transferred to the fully connected layers for the learning. In addition, after the fully connected layer, dropout and BN operations are applied, respectively, to prevent overfitting and for providing a much stable and accelerated the training regime. Finally, the data coming out of the fully connected layers are subjected to the addition process through the Add layer, and the classification is performed by transferring to the output classification layer where the softmax function exists to be converted into a probability distribution between 0 and 1 corresponding to the motor fault classes. Softmax function is defined as follows:

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \text{for} i = 1, \ldots, K \tag{1}$$

In Eq. (1), $\vec{z}$ is the input vector of the softmax function, and all of z_i values are elements of the input vector of the softmax function. K is the number of classes in the classifier. The parameters of the convolution and pooling layers of both pathways in the proposed model are detailed in Table 2.

3.2 Data augmentation

Data augmentation is the process of artificially increasing the amount of data by generating new data points from the original data with a sort of minor geometric transformations in order to increase the diversity of the training set. Deep learning models require large amounts of data to train the model. However, in deep learning-based diagnostic applications, it is difficult to obtain large amounts of faulty data because it is impossible and costly for operating machines in faulty conditions for a long time. With the increase in parameters, a deep learning model can remember certain data points with an insufficient number of training data, leading to a generalizability problem with overfitting [48]. Due to the aforementioned problem, augmentation of training data is often used in fault diagnosis of a convolutional neural network [49]. Data augmentation allows the model to learn more features and variations by varying the samples in the data set. Data augmentation can help prevent overfitting when applied correctly. Data augmentation diversifies the data set, allowing the model to learn more features and variations, which increases the generalization ability of the model [50].

In this study, data augmentation is used for preventing the overfitting and increasing the classification accuracy. In

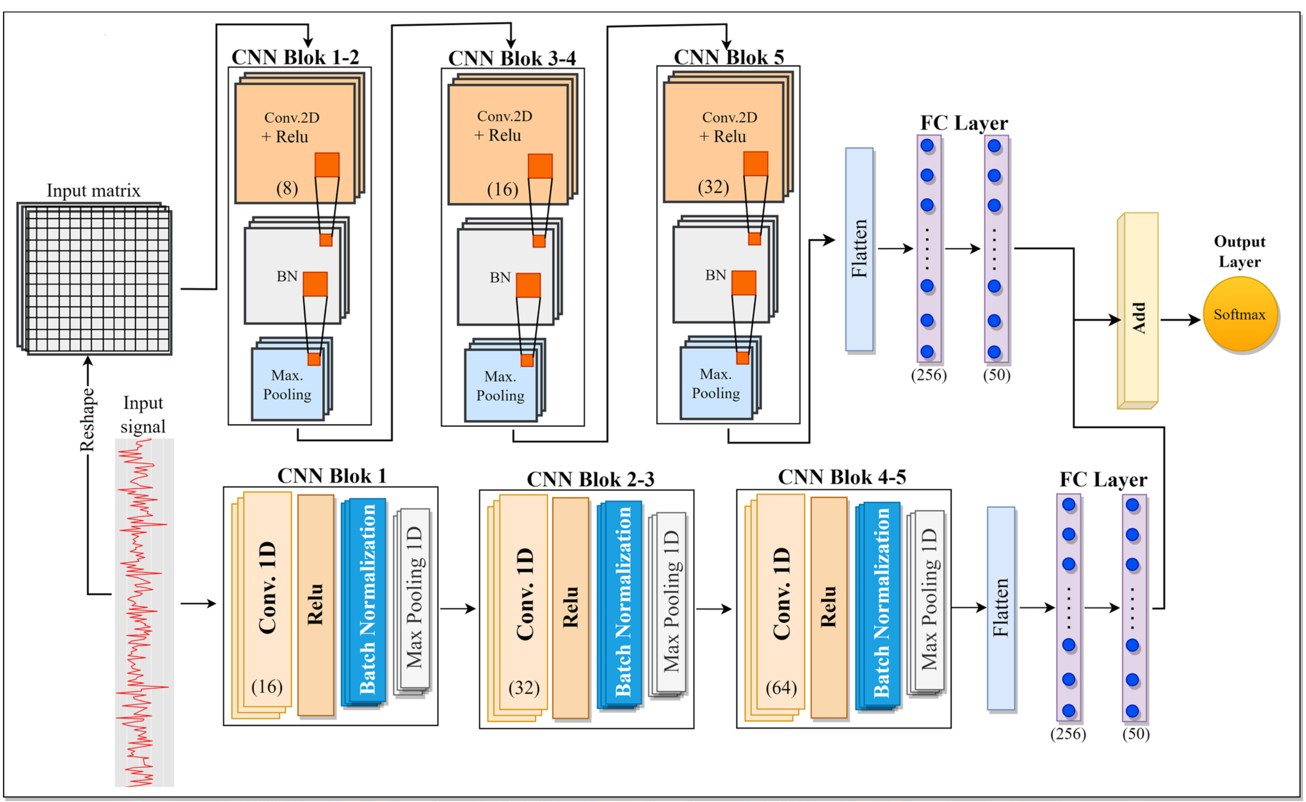


Fig. 3 Architecture of the proposed WDD-CNN model

the data augmentation process, only the amount of available training data has been increased, but the test data have been left in its original form in order to obtain more realistic results. A sliding window approach, which is an effective and easy method for data augmentation, is adopted. The overlapping sliding windows used during data augmentation ensure that each training sample obtained contains variations of the original signal. The advantages of using data augmentation in time series with the sliding window method can be listed as expanding the data set, providing diversity and variation, highlighting important features, increasing computational efficiency and preventing overfitting.

As shown in Fig. 4, in the sliding window approach, the data are increased by overlap depending on the specified step size. With the sliding window method, approximately 5568 training samples, each consisting of 3600 data points, can be created by using 64 step sizes in a data containing 360,000 data points, for example. This increases the number of samples in each fault type approximately 55 times with data augmentation using the sliding window method.

3.3 Signal-to-2D matrix conversion method

The Signal-to-2D Matrix Conversion Method is a process used in the WDD-CNN model to convert the raw time series data into 2D matrices. This conversion is essential to capture the spatial relationships and dependencies among the data points in the time series and effectively utilize both 1D and 2D representations in the model.

Here is a theoretical description of the Signal-to-2D Matrix Conversion Method and the methods used:

3.3.1 Preprocessing

Before conversion, the raw time series data are normalized to scale all features in the signal to the similar range. This is done to improve its performance and stability and to ensure that all features are treated fairly.

3.3.2 Segmentation

The preprocessed time series data are segmented into training and test samples to ensure unbiased evaluation of the model performance.

3.3.3 Signal-to-1D array conversion

As shown in Fig. 5, each segment of the time series data is converted into a 1D array of length N^2. To achieve this, the sequential data points in the time series are stacked together, forming a long 1D array. The length (N^2) of the 1D arrays is created for the model input is determined to include the failure effect in the engine.

3.3.4 1D to 2D matrix transformation

As shown in Fig. 5, after obtaining the 1D arrays, they are reshaped into $N \times N$ matrices, where N represents the length of each side of the square matrix. This

Table 2 Details of the proposed WDD-CNN model

Convolutional pathway 1					Convolutional pathway 2			
No	Layer type	Kernel size	Stride	Kernel number	Layer type	Kernel size	Stride	Kernel number
C1	1D Convolution	512×1	2×1	16	2D Convolution	15×15	2×2	8
C2	Max Pooling 1D	2×1	2×1	16	Max Pooling 2D	2×2	2×2	8
C3	1D Convolution	32×1	2×1	32	2D Convolution	5×5	2×2	8
C4	Max Pooling 1D	2×1	2×1	32	Max Pooling 2D	2×2	2×2	8
C5	1D Convolution	32×1	2×1	32	2D Convolution	5×5	2×2	16
C6	Max Pooling 1D	2×1	2×1	32	Max Pooling 2D	2×2	2×2	16
C7	1D Convolution	64×1	2×1	64	2D Convolution	3×3	2×2	16
C8	Max Pooling 1D	2×1	2×1	64	Max Pooling 2D	2×2	2×2	16
C9	1D Convolution	64×1	2×1	64	2D Convolution	3×3	2×2	32
C10	Max Pooling 1D	2×1	2×1	64	Max Pooling 2D	2×2	2×2	32
C11	Fully connected	256		1	Fully connected	256		1
C12	Fully connected	50		1	Fully connected	50		1

Output			
No	Layer type	Kernel size	Number of kernel
O1	Softmax	10	1

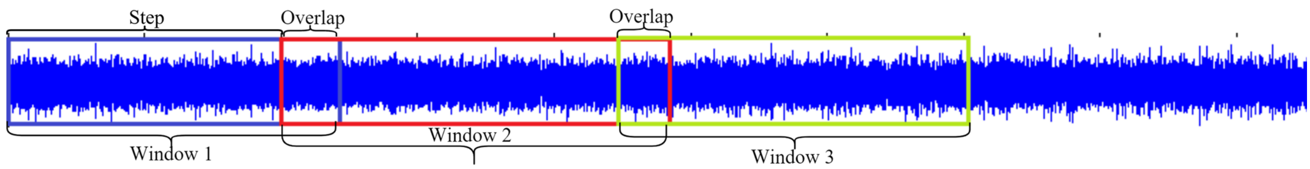


Fig. 4 Training data augmentation using overlap

transformation converts the 1D representation of the sequential data into a 2D representation, allowing the model to capture spatial patterns and dependencies.

3.3.5 Dual-pathway model

The transformed 2D matrices and the original 1D arrays are used as input data for the two pathways of the WDD-CNN model. This dual-pathway architecture enables the model to leverage both 1D and 2D representations, enhancing its ability to learn complex features from the time series data.

The methods used in this conversion process, such as reshaping the 1D arrays into 2D matrices, are standard techniques in data manipulation and representation in deep learning models. By applying this Signal-to-2D Matrix Conversion Method, the WDD-CNN model can effectively handle the complex spatial and temporal dependencies present in the vibration signals, resulting in improved performance in real-time condition monitoring and fault diagnosis tasks.

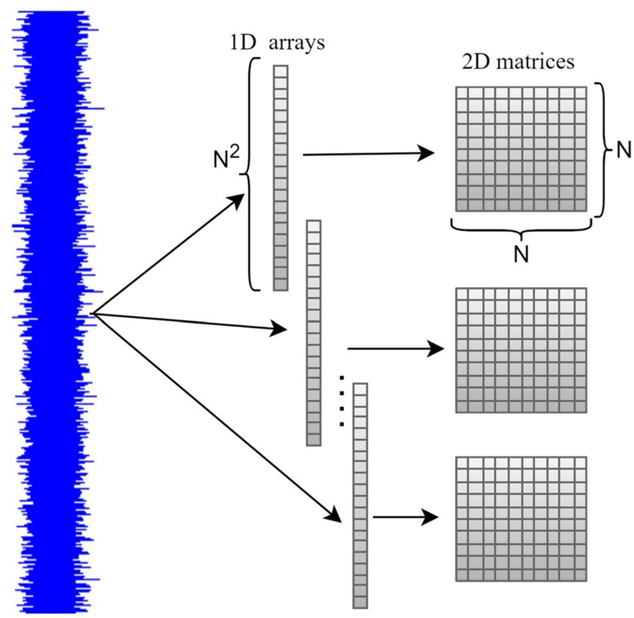


Fig. 5 Signal-to-2D matrix conversion

3.4 Training of the WDD-CNN model

There are many hyperparameters in CNN architectures that affect the performance of the model. The optimum hyperparameter values applied to the proposed model in this study are determined using the random search method [51]. The hyperparameter values applied for training are shown in Table 3. In addition, computer hardware with NVIDIA Gforce RTX2080Ti graphics card, 32 GB memory and 2.94GHz i5-1400F processor is used for model training.

The proposed WDD-CNN architecture is designed to use both 1D and 2D input data together. The main difference between 1 and 2D CNNs is that traditional 2D-CNN uses two-dimensional kernel and two-dimensional feature maps whereas 1D-CNN uses one-dimensional kernel and one-dimensional feature maps. Glorot uniform initialization method is used in all convolution kernels of the proposed model for weight initialization [52]. We have used the Adam optimization algorithm with an initial learning rate of $1e-3$ and a final learning rate of $1e-4$ to update the model weights during training [53]. Glorot uniform initialization method ($U[-\text{limit}, \text{limit}]$) is calculated as a random number with a uniform probability distribution in the range $[-\text{limit}, \text{limit}]$. Here, Glorot uniform weight initialization is expressed as in Eq. (2), where n_{in} and n_{out} are the number of input and output units in the weight tensor, respectively.

$$W = U\left[\frac{-\sqrt{6}}{\sqrt{n_{\text{in}} + n_{\text{out}}}}, \frac{\sqrt{6}}{\sqrt{n_{\text{in}} + n_{\text{out}}}}\right] \tag{2}$$

Adaptive Moment Estimation (Adam) is a method that calculates the adaptive learning rate for each parameter. Adam uses the exponential moving average v_t of the past squared gradients (g_t^2) to scale the learning rate, and the exponential moving average m_t of the past gradients instead of the gradient itself (g_t) to leverage momentum. Exponential moving averages (v_t and m_t) are defined as follows:

$$m_t = \beta_1 m_{t-1} + \left(1 - \beta_1\right) g_t \tag{3}$$

$$v_t = \beta_2 v_{t-1} + \left(1 - \beta_2\right) g_t^2 \tag{4}$$

Where β_1 and β_2 are exponential decay rates. m_{t-1} and v_{t-1} indicate the previous exponential moving averages. The vectors of the moving averages (m_t andv_t) are initialized with zeros in the first iteration. The estimators (m_t andv_t) tend to zero at the first-time steps and especially when the decay rates are small (i.e., β1 and β2 are close to 1). Therefore, it is necessary to correct the estimators. Bias correction is as in Eq. (5), respectively. With t in β_1^t and β_2^t, we denote power of $\beta 1$ and $\beta 2$:

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t} \text{and} \hat{v}_t = \frac{v_t}{1-\beta_2^t} \tag{5}$$

Finally, by replacing m_t ve v_t with the terms $\hat{m}_t$ ve $\hat{v}_t$, Adam weight update formula is obtained as follows:

$$\theta_{t+1} = \theta_t - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \varepsilon} \tag{6}$$

where η is the step size, θ_t is the previous weight, and ε is a very small positive constant such as 10^{-8} that is used to correct the division by zero error.

In the proposed model, the categorical cross-entropy function has been used to calculate the cross-entropy loss between the target class probability distribution and the softmax output probability distribution estimated and to evaluate the training. In the categorical cross-entropy function, the closer the estimated value is to the ground truth, the less the loss. The cross-entropy loss between p and q is shown as follows:

$$\text{Loss} = H(p,q) = -\sum_{i=1}^{n} p_i \log(q_i) \tag{7}$$

where n shows the number of class. p_i and q_i are the original label value and softmax output value for i. class, respectively.

In order to prevent overfitting and improve the adaptability of the network, a standard dropout with 50% dropout rate has been applied at the end of each fully connected layer in both pathways of the model [54]. In addition, at the end of each convolution block in the 2D-CNN pathway, a total of five standard dropouts with dropout rate of 25% have been applied. The performances of the proposed model are evaluated in the next section.

4 Validation of the WDD-CNN model through comparative analysis

4.1 Data description

Bearing fault dataset from CWRU Bearing Data Center [55] is used to validate the proposed WDD-CNN model and compare its performance with existing studies. The CWRU bearing dataset is chosen because it is open source and widely used in the scientific literature [56]. The vibration data provided by CWRU is data collected from the bearing test platform shown in Fig. 6 (Image in Fig. 6 taken from itself website). The platform shown in Fig. 6 consists of a 2 HP motor, a shaft mounted torque transducer/encoder (center), dynamometer (right) and electronic control system (not shown). The test bearings support the motor shaft. The vibration data from the test platform are collected data for fault-free (normal) bearings, single-point drive-end and fan-end faults. The vibration data are collected at 12 and 48 kHz for drive-end bearing faults, and all fan-end bearing data are collected at 12 kHz sampling rates. The data are obtained from two different channels by measuring from two different locations near to and remote from the motor bearings for each type of fault. For the experiments to be carried out to form the dataset, four types of faults of varying severity, 0.007, 0.014, 0.021, and 0.028 inches in diameter, are formed using electro-discharge machining (EDM) on the drive and fan-end bearings. There are four different types of bearing faults including normal, ball fault, inner raceway fault and outer raceway fault. Faults ranging from 0.007 inches in diameter to 0.028 inches in diameter are

Table 3 Hyperparameter values applied for model training

Hyperparameter	Description	Value
Learning rate	Rate at which the model learns from data	0.001
Batch size	Number of samples used in each iteration	10
Epochs	Number of times the entire dataset is passed through the model	50
Activation function	Nonlinear function applied to the output of each neuron	ReLU
Optimizer	Algorithm used to update the model's parameters during training	Adam
Dropout rate (After each pooling layer)	Proportion of neurons randomly set to zero during training to prevent overfitting	0.25
Dropout rate (After FC Layer)		0.5
Weight initialization	Method used to set the initial values of the model's weights	Xavier/Glorot Initialization

introduced separately at the inner raceway, ball and outer raceway. Engine loads of 0hp, 1hp, 2hp and 3hp are applied by attaching defective bearings to the test platform separately, and data are collected by operating at four different fixed speeds (four different motor speeds of 1730, 1750, 1772 and 1797 rpm).

It is recommended to use a high sampling rate for data collection, since bearing faults are more evident at high frequencies [57]. For this reason, 48 kHz driver end bearing fault data are used in this study. It has been shown that the training and testing of the proposed model are performed only with the data collected from a single accelerometer (Drive-end accelerometer), and good results are obtained even with the data coming from a single channel. In Table 4, the fault conditions of the 48 kHz data set used in our study are shown in detail.

The data, collected from the shaft rotating between 1772 and 1730 rpm at operating conditions of the CWRU experimental set, with a sampling frequency of 48 kHz, contain between 1626 and 1665 data points per revolution. The training samples used to train the model in this study consist of 3600 data points to include the bearing fault impact. This means that there are at least two bearing fault impacts in each training sample.

The training samples used to train the proposed model in this study are obtained by augmenting the existing data using the sliding window method.

In our study, the sliding window method is used separately for each class by taking an equal amount of data points from each class (360,000 data points), that is, each input sequence is obtained under a single fault condition, so all augmented sequences are assigned to the same fault tag as the original input sequence. Thus, class imbalance is prevented. With data augmentation, 5625 training samples are obtained for each fault by using 64 step size and 3600 window length in the sliding window. These obtained training samples are prepared for the 1DCNN pathway of the proposed model. For the 2DCNN pathway of the model, each training sample consisting of 3600 data points is prepared by transforming into 60×60 matrices.

Each test sample is created without data augmentation using a window length of 3600, in accordance with the real-world application. Test samples consist of 100 samples for each fault. The number of each fault type for both training and test datasets under different loads (1hp, 2hp, 3hp) is evenly distributed as shown in Table 5. Each training is repeated ten times for different scenarios, and the results are obtained by taking the averages.

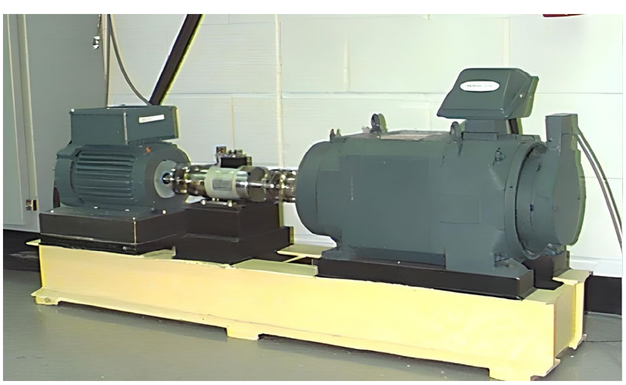


Fig. 6 CWRU Bearing Data Center test stand [55]

4.2 Performance metric for experiments

Accuracy, Precision, Recall and F1 score performance measures are used to evaluate the performance of the proposed WDD-CNN model under different operating loads. These measures are briefly described as follows:

Accuracy is a metric that represents the number of correctly classified data samples over the total number of data samples. In other words, it evaluates the capacity of the algorithm by measuring the ratio of correctly predicted observations to the total number of observations. However, if the accuracy dataset is balanced, it gives more intuitive results. Since the dataset is balanced in our fault classification, the accuracy the primary performance measure, but the results of other performance measures (Precision, Recall and F1 score) are also presented to provide more insight into the performance of the model. It is expressed mathematically as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{FP} + \text{TN}} \tag{8}$$

where TP, TN, FP, and FN represent number of true positive cases, number of true negative cases, number of false positive cases, and number of false negative cases, respectively.

Table 4 Description of bearing fault for CWRU datasets

Fault ID	Fault label	Fault type	Fault diameter (inch)
1	normal	Normal	0
2	b007	Ball fault	0.007
3	b014		0.014
4	b021		0.021
5	ir007	Inner race fault	0.007
6	ir014		0.014
7	ir021		0.021
8	or007c	Outer race fault	0.007
9	or014c		0.014
10	or021c		0.021

Precision is the ratio of correctly predicted positive cases to the total predicted positive cases. This metric highlights the true positive predictions out of all positive predictions. Precision value close to one indicates the low false positive rate. It is expressed mathematically as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{9}$$

Recall is the ratio of correctly predicted positive observations to the all observations belong to a class. Recall highlights the sensitivity of the algorithm, i.e., it defines how many of all true positives are captured by the model. It is expressed mathematically as follows:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{10}$$

$F1$ score, defined mathematically by Eq. (10), is the weighted average of Precision and Recall. F1-score combines Precision and Recall, providing a balance between them. Therefore, this score takes both false positives and false negatives into account:

$$F1 - \text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{11}$$

Precision, recall, and F1-score offer a viable alternative to the traditional accuracy metric and provide detailed information about the method being analyzed. Precision, recall, and F1 score metrics are calculated as macro-averaged, since multiple fault classes are taken into account, and our sample numbers are equal for each fault class. The macro-mean is obtained by calculating the arithmetic mean over all classes, after each metric is calculated separately per class.

4.3 Experimental results and discussion

The operating and load conditions of electric drive systems are variable. In addition, noise is inevitable in the operating conditions of electric motors in the real-world applications, and this noise adversely affects raw vibration signals. Therefore, the intelligent diagnostic systems are expected to have an effective and high performance under different load conditions and noisy environments. In this section, the performance of the proposed WDD-CNN model will be evaluated separately under these two situations (variable load and noisy environment conditions). The evaluation is carried out under the following limitations.

1. Since bearing failures are more pronounced at high frequencies, 48 kHz driver end bearing fault dataset are used.
2. Each training sample are adjusted to include the bearing fault impact.
3. Tests are carried out under different loads (1 hp, 2 hp, 3 hp) by equally distributing the number of each fault type for both training and test datasets.
4. Each training is repeated ten times for different scenarios and the results are obtained by taking the averages.
5. Training samples are augmented with the sliding window method, but the test data are left in its original form.
6. To avoid class imbalance, each input data are acquired under a single fault condition so that all augmented sequences are assigned the same fault tag as the original input data.
7. The assumptions and limitations mentioned are discussed in detail below.

Table 5 Numerical distribution of each fault type under different loads

	Dataset A (Motor Load:1hp)	Dataset B (Motor Load:2hp)	Dataset C (Motor Load:3hp)		Dataset A (Motor Load:1hp)	Dataset B (Motor Load:2hp)
Fault Id	Train samples	Test samples	Train samples	Test samples	Train samples	Test samples
1	5625	100	5625	100	5625	100
2	5625	100	5625	100	5625	100
3	5625	100	5625	100	5625	100
4	5625	100	5625	100	5625	100
5	5625	100	5625	100	5625	100
6	5625	100	5625	100	5625	100
7	5625	100	5625	100	5625	100
8	5625	100	5625	100	5625	100
9	5625	100	5625	100	5625	100
10	5625	100	5625	100	5625	100

4.3.1 The effect of data number on performance

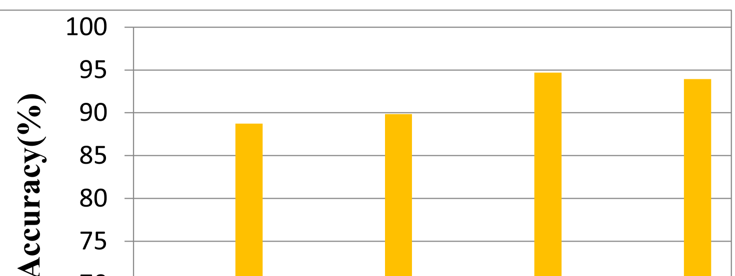
There are many parameters that affect the performance of the model in CNN architectures. The number of training data is a critical factor affecting the performance of the CNN model. More training data can improve the generalization of the model, resulting in better performance [31]. If too little training data are used, the model's learning capacity may be limited, and the risk of overfitting may be high. However, when a lot of training data are used, the learning process of the model may take longer, and the data processing cost may increase. Therefore, it is important to carefully determine the number of training data. The training data set should be created in a highly representative, diverse and balanced way. The number of training data is an important parameter to consider when evaluating the performance of the model. In order to investigate how much training data the WDD-CNN model can perform better in our study, our model under the hyperparameter conditions given in Table 3 is trained with the C dataset and tested with the A and B datasets. Different numbers of training data are obtained by augmenting with the step numbers of 64, 128, 256 and 512, respectively, using the sliding window method, but the test data are not augmented. Here, in determining the number of training data, care is taken to include the bearing fault effect of each training sample as mentioned in Sect. 4.1. The generated training dataset is a balanced dataset.

The number of each fault type in the training samples is the same. Ten runs are made to reduce the negative effects of the random initial values of the network. The recommended method is implemented with NVIDIA GForce RTX2080Ti graphics card and i5-1400F processor with 32 GB memory at 2.94 GHz. Results figured. It is shown in Fig. 7. Accuracy values in Fig. 7 are given by taking the average of the test results. At Fig. 7, it is clear that as the training samples increase, the standard deviation decreases while the accuracy increases. It is seen that the accuracy increases by 5.95% with the increase in the number of training samples. However, the average accuracy value obtained with the data augmented with the number of 128 steps in these tests is 0.75% higher than the average accuracy value obtained with the data augmented with the number of steps of 64. This shows that our model gives good results with less data. While our training sample numbers are 2532 and 5063 per fault, our accuracy rate is over 90%, showing that our model has high performance.

An important consideration in diagnostic methods is the time cost required to produce a classification inference. In real-world applications, the acceptable time for model fault inference may vary depending on the application's requirements and usage scenario. Especially, in real-time or interactive applications, low fault diagnosis time is important because the user may need a quick response. In this section, the fault inference times of our model have also been determined and given in Fig. 7. When these inference times are examined, it shows that the fault inference time (average 0.72 s) is acceptable for real-world applications of our model.

4.3.2 Performance under different working conditions

Experimental studies in this section are carried out to measure the performance of the proposed model against data obtained under different load conditions. Data from three different load conditions (1 hp, 2 hp and 3hp) are used for training the proposed model. Table 6 shows how to arrange the datasets. Dataset A, B and C in Table 6 are obtained from the collected signals at the load conditions of 1, 2 and 3 hp, respectively. As mentioned in Sect. 3.2, the training samples required for the training of the model are augmented with the sliding window method, but the data augmentation operation is not performed for the test samples, and the original data



	1	2	3	4
Window step	512	256	128	64
Number of training data	6320	12670	25320	50630
Time(sec.)	0.72	0.74	0.7	0.7
Accuracy(%)	88.75	89.85	94.7	93.95

Fig. 7 Numbers of training samples, training times and diagnosis results (C to AB)

are used. Experiments are carried out with 5625 training samples and 100 test samples, each containing 3600 data points for each fault class. The confusion matrices of the model are given in Fig. 8.

When Fig. 8 is investigated, the model trained with dataset B confuses the 21 inch inner race fault (ir021) in dataset C with the 21 inch rolling element fault (b021) at a rate of approximately 50%. When the model is trained with dataset C, it confuses 21 inch rolling element fault (b021) in dataset A with 14 inch outer race fault (or014) approximately 77%. Again, it is seen that the model trained with dataset C confuses the 21 inch inner race fault (ir021) in dataset A with14 inch outer race fault (or014) by 20%. This shows us that the 21 inch inner race bearing fault data in the A datasets and the 21 inch rolling element fault data in the C datasets are problematic datasets. It is seen that the model performs very well in detecting other types of faults. As a result, the proposed model works by combining the advantages of 1D-CNN and 2D-CNN architectures. This means that 1D-CNN is effective in extracting the features of one-dimensional data, while 2D-CNN is effective in capturing the features of multidimensional data. When the confusion matrices of our model, which is tested under variable load conditions, are examined, it shows that the model can capture both the spatial and temporal properties of the data very well by using this dual pathway.

The Accuracy metric alone may not be sufficient to fully evaluate classification performance. It should be supported by other performance metrics. For example, metrics such as precision, recall, F1 score can be used to further evaluate the model's performance on false positives, false negatives, true positives, and true negatives. Therefore, the accuracy metric can be used in conjunction with other performance metrics, allowing us to evaluate the model's performance more accurately and comprehensively.

In addition to our study, to provide more information on the performance of the proposed model, precision, recall (sensitivity) and F1-score results are also presented in Table 7. Precision shows how many of the values we predicted as positive are actually positive. Sensitivity, on the other hand, is a metric that shows how many of the values we need to positively predict, we predict positively. Therefore, precision gives information about the cost of false positives, while sensitivity provides information about the cost of false negatives.

Table 6 Dataset arrangement for load adaptation

Labeled data for training	Unlabeled data for test	
Dataset A (data of 1 hp)	Dataset B	Dataset C
Dataset B (data of 2 hp)	Dataset A	Dataset C
Dataset C (data of 3 hp)	Dataset A	Dataset B

A high precision value indicates that the model tends to reduce false positives. In balanced classes, precision can evaluate the model's ability to make true positive predictions. A high recall value indicates that the model tends to reduce false negatives. In balanced classes, recall can measure the samples that the model misses and evaluate the model's ability to catch true positives. The F1 score is a metric that represents the weighted average of precision and recall. When Table 5 is examined, it is seen that the proposed model has high performance in terms of precision, recall and F1 score, with average values of 97.57, 96.45 and 96.23%, respectively.

4.3.3 Comparison with the results of different methods

Comparative results of the adaptation performance of the proposed model in different load conditions with WDCNN [31], SRDCNN [32], TICNN [33], RNN-WDCNN [38], BiLSTM [51] and LINET (AdaBN) [58] models are detailed in Fig. 9. Table 8 shows at which sampling frequencies the collected raw signals are collected for obtaining the CWRU dataset used in the compared models.

In contrast to existing literature, our study deviates by utilizing vibration signals sampled at 48 kHz from the fan end and drive-end accelerometers, instead of the commonly used 12 kHz sampled data. Although data with a sampling frequency of 12 kHz are easier to diagnose, many bearing faults manifest at high frequencies, as noted by Smith and Randall [57]. Therefore, an effective diagnostic framework must be capable of working with data with a high sampling frequency. It is important that the data sampling frequency is sufficient so that faults at high frequencies can be detected accurately. In this case, the decision to use data with a sampling frequency of 48 kHz reflects a more efficient approach to detecting bearing faults present at higher frequencies. In this way, it is aimed to better detect the faults that exist at high frequencies and to provide an accurate diagnosis.

When the accuracy average values of the models are investigated and the sampling frequencies of the data they use are taken into consideration, it is clearly seen that the proposed model is superior to the other models. In addition, when the accuracy average values of the models are examined, and the sampling frequencies of the data run in the models are taken into account, it is clearly seen that the proposed WDD-CNN model is superior to other models with an accuracy rate of 96.45%.

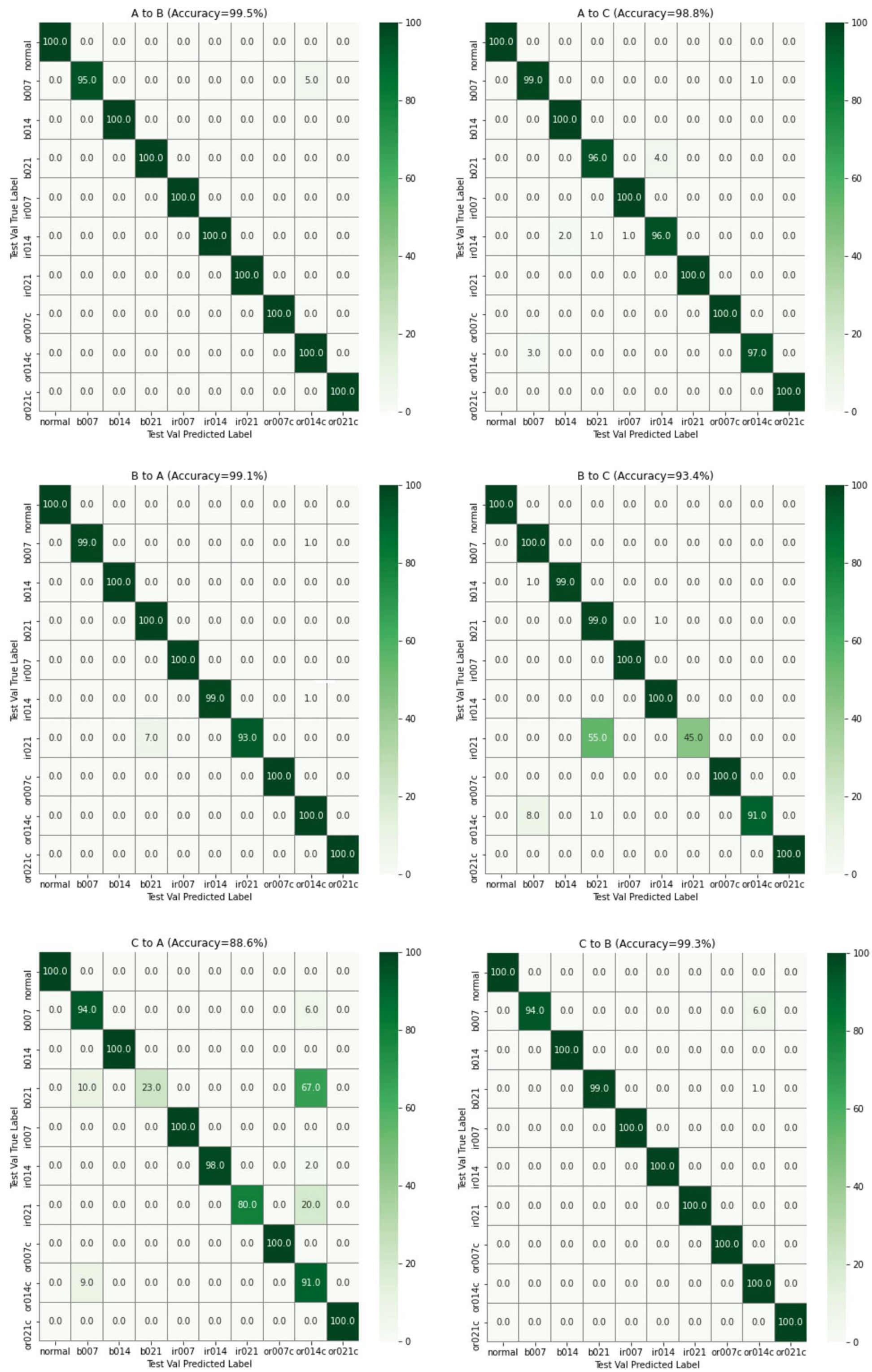


Fig. 8 Confusion matrices for the WDD-CNN model on the different load adaptation cases of test datasets

Table 7 Load adaptation precision, recall and F1 scores of the WDD-CNN

Metrics	A to B	A to C	B to A	B to C	C to A	C to B	Average
Precision	99.5	98.8	99.1	95.5	93.2	99.3	97.57
Recall	99.5	98.8	99.1	93.4	88.6	99.3	96.45
F1 Score	99.5	98.8	99.1	93.0	87.7	99.3	96.23

4.3.4 Performance under different amounts of noise

In this section, performance of the proposed WDD-CNN model to noisy environments has been investigated and verified. In real-world industrial production conditions, there are many sources of noise and interference with the collected vibration signals is inevitable. Therefore, the proposed model should work effectively in noisy environment and its performance should be verified. In the experimental study, datasets A, B and C are tested separately. 70% of the data in each data set is reserved for training by augmenting with the sliding window method mentioned in Sect. 3.2, and the remaining 30% data are reserved for testing in original form without augmentation. By adding white Gaussian noise to the original data reserved for testing, noisy test data with different signal-to-noise ratios (SNR) in the range of 0–10 dB are obtained. With the noisy test data obtained, our proposed WDD-CNN model is tested, and the results are presented in Table 9.

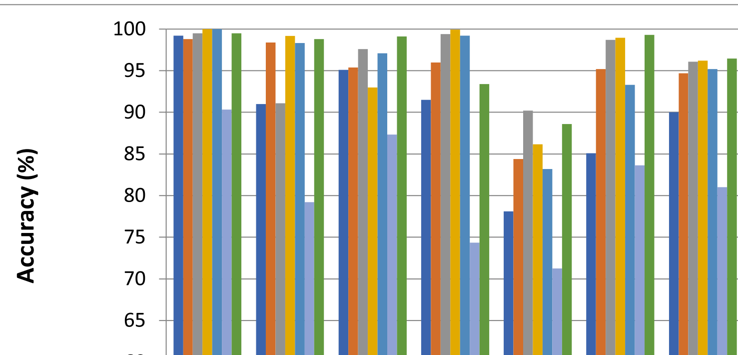
When Table 9 is investigated, it is seen that our model performs very well at low signal-to-noise ratios (at least 99% between 4 and 10 dB). It is obvious that the proposed model achieves very good results for Dataset A and Dataset C at noise ratios of 0 and 2 dB. However, it is seen that the performance has decreased slightly in Dataset B. When the average accuracy values are examined, it is seen that our model gives good results in noisy environments.

5 Conclusions and further work

In this study, an intelligent fault diagnosis model was proposed for condition monitoring and fault diagnosis of electrical motors operating under variable load conditions. The proposed model was introduced as a novel dual-pathway deep learning model that combines 1D and 2D convolutional layers. WDD-CNN model directly processed raw vibration signals without time-consuming manual feature extraction. The WDD-CNN model, tested using the CWRU bearing dataset and the results of which are given comparatively in Sect. 4.3, was found to provide superior performance with high classification accuracy.

In addition to the superior performance of the model under different load conditions, its robustness against noisy environments was also demonstrated in this study. With these results, it was exposed that good results could be obtained with data from a single sensor channel through a successful model. Consequently, the proposed WDD-CNN model proved to be an efficient solution for condition monitoring and intelligent diagnostics of electric motors operating in high-noise industrial environments.

However, testing the proposed model on artificial fault signals obtained by EDM method instead of real-world



	A to B	A to C	B to A	B to C	C to A	C to B	Avg.
WDCNN [31]	99.2	91	95.1	91.5	78.1	85.1	90
SRDCNN [32]	98.8	98.4	95.4	96	84.4	95.2	94.7
TICNN [33]	99.5	91.1	97.6	99.4	90.2	98.7	96.08
RNN-WDCNN [38]	100	99.17	93	99.92	86.16	98.94	96.2
LINET (AdaBN) [58]	100	98.31	97.09	99.21	83.19	93.31	95.19
BiLSTM [51]	90.32	79.21	87.32	74.35	71.25	83.65	81.02
WDD-CNN (Proposed Method)	99.5	98.8	99.1	93.4	88.6	99.3	96.45

Fig. 9 Accuracy comparison between WDD-CNN and five other different models (WDCNN, SRDCNN, TICNN, RNN-WDCNN, LINET (AdaBN) and BiLSTM) under varying load domains

Table 8 Sampling frequencies of the CWRU dataset used in the compared models

Models	WDCNN [31]	SRDCNN [32]	TICNN [33]	RNN-WDCNN [38]	BilSTM [51]	LINET (AdaBN) [58]	WDD-CNN
Sampling frequency of raw data	12 kHz	12 kHz	12 kHz	48 kHz	48 kHz	-	48 kHz

Table 9 Accuracy scores of the WDD-CNN under different amounts of noise

	0 dB	2 dB	4 dB	6 dB	8 dB	10 dB	Avg
Dataset A	89.7	99.7	100	100	100	100	98.22
Dataset B	83	91	99	99.7	99.7	99.7	95.34
Dataset C	89.3	96	99	100	100	100	97.38
Avg.	87.33	95.57	99.33	99.9	99.9	99.9	

faults and only for bearing fault can be seen as disadvantages and shortcomings. Using artificial fault signals may cause some limitations in reflecting real-world data. Real-world data may have more noise, variability, and complexity. Therefore, it is important to remember that testing based on real data is important to fully evaluate how your model will perform in real-world conditions. Additionally, our research focused solely on bearing faults. However, faults in electrical motors can often be very diverse and addressing just one type of fault may not cover the wider range of faults you may encounter in real-world applications.

In future work, we plan to integrate motor current signals alongside vibration signals to provide a more comprehensive approach to intelligent diagnostics and control. This will enable us to detect various types of faults in engines more precisely and accurately while enhancing overall engine performance. Instead of relying on the global bearing dataset, we will conduct tests for different types of electric motor faults using both normal and faulty data collected from our electric motor test setup, and we will evaluate performance of the proposed model. Our goal is for the proposed model to achieve a sensitive and reliable fault diagnosis capability for different types of electric motor faults.

Furthermore, we aim to test our real-time intelligent diagnostic model on real hardware setups, incorporating a wider range of data, including real-world failure conditions rather than artificial faults. This will further improve the classification accuracy, reliability, and generalizability of our model.

References

1. Amin AA, Hasan KM (2019) A review of fault tolerant control systems: advancements and applications. Measurement 143:58–68
2. Amin AA, Mahmood-ul-Hasan K (2019) Hybrid fault tolerant control for air–fuel ratio control of internal combustion gasoline engine using Kalman filters with advanced redundancy. Meas Control 52(5–6):473–492
3. Alsuwian T, Amin AA, Iqbal MS, Qadir MB, Almasabi S, Jalalah M (2022) Design of active fault-tolerant control system for air-fuel ratio control of internal combustion engine using nonlinear regression-based observer model. PLoS ONE 17(12):e0279101
4. Gutiérrez León P, García-Morales J, Escobar-Jiménez RF, Gómez-Aguilar JF, López-López G, Torres L (2018) Implementation of a fault tolerant system for the internal combustion engine's MAF sensor. Measurement 122:91–99
5. Nandi S, Toliyat HA, Li X (2005) Condition monitoring and fault diagnosis of electrical motors—a review. IEEE Trans Energy Convers 20(4):719–729
6. Benbouzid MEH (1998) A review of induction motors signature analysis as a medium for faults detection. IEEE Trans Electron Devices 47(5):984–993
7. Butt AS, ul Huda N, Amin AA (2023) Design of fault-tolerant control system for distributed energy resources based power network using Phasor Measurement Units. Meas Control 56(1–2):269–286
8. Li Y, Yang Y, Wang X, Liu B, Liang X (2018) Early fault diagnosis of rolling bearings based on hierarchical symbol dynamic entropy and binary tree support vector machine. J Sound Vib 428:72–86
9. Pires VF, Foito D, Martins JF, Pires AJ (2015) Detection of stator winding fault in induction motors using a motor square current signature analysis (MSCSA). In: 2015 IEEE 5th international conference on power engineering, energy and electrical drives (POWERENG), pp 507–512
10. Samanta B, Al-Balushi KR, Al-Araimi SA (2005) Artificial neural networks and genetic algorithm for bearing fault detection. Soft Comput 10(3):264–271
11. Zhang Z, Wang Y, Wang K (2012) Fault diagnosis and prognosis using wavelet packet decomposition, Fourier transform and artificial neural network. J Intell Manuf 24(6):1213–1227
12. Saidi L, Ben Ali J, Fnaiech F (2015) Application of higher order spectral features and support vector machines for bearing faults classification. ISA Trans 54:193–206
13. Senanayaka SL, Kandukuri ST, Khang HV, Robbersmyr KG (2017) Early detection and classification of bearing faults using support vector machine algorithm. In: 2017 IEEE workshop on electrical machines design, control and diagnosis (WEMDCD), pp 250–255
14. Gunerkar RS, Jalan AK, Belgamwar SU (2019) Fault diagnosis of rolling element bearing based on artificial neural network. J Mech Sci Technol 33(2):505–511
15. Tian J, Morillo C, Azarian MH, Pecht M (2016) Motor bearing fault detection using spectral kurtosis-based feature extraction coupled with K-nearest neighbor distance analysis. IEEE Trans Ind Electron 63(3):1793–1803
16. Stetco A, Dinmohammadi F, Zhao X, Robu V, Flynn D, Barnes M, Keane J, Nenadic G (2019) Machine learning methods for wind turbine condition monitoring: a review. Renew Energy 133:620–635
17. Duan L, Xie M, Wang J, Bai T (2018) Deep learning enabled intelligent fault diagnosis: overview and applications. J Intell Fuzzy Syst 35:5771–5784
18. Liu R, Yang B, Zio E, Chen X (2018) Artificial intelligence for fault diagnosis of rotating machinery: a review. Mech Syst Signal Process 108:33–47
19. Dineva A, Mosavi A, Gyimesi M, Vajda I, Nabipour N, Rabczuk T (2019) Fault diagnosis of rotating electrical machines using multi-label classification. Appl Sci 9(23):5086
20. Asr MY, Ettefagh MM, Hassannejad R, Razavi SN (2017) Diagnosis of combined faults in Rotary Machinery by Non-Naive Bayesian approach. Mech Syst Signal Process 85:56–70

21. Georgoulas G, Karvelis P, Loutas T, Stylios CD (2015) Rolling element bearings diagnostics using the Symbolic Aggregate approXimation. Mech Syst Signal Process 60–61:229–242
22. Piltan F, Kim JM (2021) Bearing anomaly recognition using an intelligent digital twin integrated with machine learning. Appl Sci 11(10):4602
23. Shahbaz MH, Amin AA (2023) Design of hybrid fault-tolerant control system for air-fuel ratio control of internal combustion engines using artificial neural network and sliding mode control against sensor faults. Adv Mech Eng. https://doi.org/10.1177/16878132231160729
24. Shifat TA, Hur J-W (2021) ANN assisted multi sensor information fusion for BLDC motor fault diagnosis. IEEE Access 9:9429–9441
25. Li C, Sanchez R-V, Zurita G, Cerrada M, Cabrera D, Vasquez RE (2015) Multimodal deep support vector classification with homologous features and its application to gearbox fault diagnosis. Neurocomputing 168:119–127
26. Kamilaris A, Prenafeta-Boldú FX (2018) Deep learning in agriculture: a survey. Comput Electron Agric 147:70–90
27. Zhang J, Sun Y, Guo L, Gao H, Hong X, Song H (2019) A new bearing fault diagnosis method based on modified convolutional neural networks. Chin J Aeronaut 33(2):439–447
28. Shao S, Yan R, Lu Y, Wang P, Gao R (2019) DCNN-based multisignal induction motor fault diagnosis. IEEE Trans Instrum Meas 69(6):2658–2669
29. Wang C-S, Kao I-H, Perng J-W (2021) Fault diagnosis and fault frequency determination of permanent magnet synchronous motor based on deep learning. Sensors 21(11):3608
30. Wang J, Fu P, Zhang L, Gao RX, Zhao R (2019) Multi-level information fusion for induction motor fault diagnosis. IEEE ASME Trans Mechatron 24(5):2139–2150
31. Zhang W, Peng G, Li C, Chen Y, Zhang Z (2017) A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals. Sensors 17(2):425
32. Zhuang Z, Lv H, Xu J, Huang Z, Qin W (2019) A deep learning method for bearing fault diagnosis through stacked residual dilated convolutions. Appl Sci 9(9):1823
33. Zhang W, Li C, Peng G, Chen Y, Zhang Z (2018) A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load. Mech Syst Signal Process 100:439–453
34. Jia F, Lei Y, Lu N, Xing S (2018) Deep normalized convolutional neural network for imbalanced fault classification of machinery and its understanding via visualization. Mech Syst Signal Process 110:349–367
35. Karim F, Majumdar S, Darabi H, Chen S (2018) LSTM fully convolutional networks for time series classification. IEEE Access 6:1662–1669
36. Chen Y, Keogh E, Hu B, Begum N, Bagnall A, Mueen A, Batista G (2015) The UCR Time Series Classification Archive. https://www.cs.ucr.edu/~eamonn/time_series_data/. Accessed 15 Sept 2022
37. Zhang Y, Xing K, Bai R, Sun D, Meng Z (2020) An enhanced convolutional neural network for bearing fault diagnosis based on time–frequency image. Measurement 157:107667
38. Shenfield A, Howarth M (2020) A novel deep learning model for the detection and identification of rolling element-bearing faults. Sensors 20(18):5112
39. Fang W, Chen Y, Xue Q (2021) Survey on research of RNN-based spatio-temporal sequence prediction algorithms. J Big Data 3:97
40. Kiranyaz S, Avci O, Abdeljaber O, Ince T, Gabbouj M, Inman DJ (2021) 1D convolutional neural networks and applications: a survey. Mech Syst Signal Process 151:107398
41. Jiao J, Zhao M, Lin J, Liang K (2020) A comprehensive review on convolutional neural network in machine fault diagnosis. Neurocomputing 417:36–63
42. Zhiyi H, Haidong S, Xiang Z, Yu Y, Junsheng C (2020) An intelligent fault diagnosis method for rotor-bearing system using small labeled infrared thermal images and enhanced CNN transferred from CAE. Adv Eng Inform 46:101150
43. Shang L, Yang Q, Wang J, Li S, Lei W (2018) Detection of rail surface defects based on CNN image recognition and classification. In: 2018 20th international conference on advanced communication technology (ICACT)
44. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR 2016), pp 770–778.
45. Conneau A, Schwenk H, Barrault L, Lecun Y (2017) Very deep convolutional networks for text classification. Assoc Comput Linguist 1:1107–1116
46. Gehring J, Auli M, Grangier D, Dauphin Y (2017) A convolutional encoder model for neural machine translation. Assoc Comput Linguist 1:123–135
47. Ioffe S, Szegedy C (2015) Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd international conference on machine learning, vol 37, pp 448–456
48. Zhang C, Bengio S, Hardt M, Recht B, Vinyals O (2021) Understanding deep learning (still) requires rethinking generalization. Commun ACM 64(3):107–115
49. Shorten C, Khoshgoftaar TM (2019) A survey on image data augmentation for deep learning. J Big Data 6(1):1–48
50. Xua M, Yoonb S, Fuentes A, Park DS (2023) A comprehensive survey of image augmentation techniques for deep learning. Pattern Recogn 137:109347
51. Bergstra J, Bengio Y (2012) Random search for hyper-parameter optimization. J Mach Learn Res 13:281–305
52. Glorot X, Bengio Y (2010) Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the 13th international conference on artificial intelligence and statistics, pp 249–256
53. Kingma D, Ba J (2014) Adam: a method for stochastic optimization. In: International conference on learning representations
54. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15(56):1929–1958
55. CWRU-CaseWestern Reserve University Bearing Data Center. https://engineering.case.edu/bearingdatacenter/download-data-file. Accessed 25 Aug 2022
56. Neupane D, Seok J (2020) Bearing fault detection and diagnosis using Case Western Reserve University dataset with deep learning approaches: a review. IEEE Access 8:93155–93178
57. Smith WA, Randall RB (2015) Rolling element bearing diagnostics using the Case Western Reserve University data: a benchmark study. Mech Syst Signal Process 64–65:100–131
58 Jin T, Yan C, Chen C, Yang Z, Tian H, Wang S (2021) Light neural network with fewer parameters based on CNN for fault diagnosis of rotating machinery. Measurement 181:109639