



Fault detection and diagnosis in electric motors using 1d convolutional neural networks with multi-channel vibration signals

Ronny Francis Ribeiro Junior^{a,*}, Isac Antônio dos Santos Areias^b, Mateus Mendes Campos^b, Carlos Eduardo Teixeira^c, Luiz Eduardo Borges da Silva^b, Guilherme Ferreira Gomes^a

^a Mechanical Engineering Institute, Federal University of Itajubá – UNIFEI, Itajubá, Brazil

^b Institute of Systems Engineering and Information Technology, Federal University of Itajubá – UNIFEI, Itajubá, Brazil

^c Gnarus Institute – Itajubá, Brazil

ARTICLE INFO

Keywords:

Vibration
Fault diagnosis
1D Convolution Neural Network
Multi-head deep neural network
Multi-sensor systems

ABSTRACT

Fault detection and diagnosis in time series data are becoming mainstream in most industrial applications since the increase of monitoring sensors in machinery. Traditional methods generally require pre-processing techniques before training; however, this task becomes very time-consuming with multiple sensors. Recently, deep learning methods have shown great results on time series data. This paper proposes a multi-head 1D Convolution Neural Network (1D CNN) to detect and diagnose six different types of faults in an electric motor using two accelerometers measuring in two different directions. This architecture was chosen due to each head can deal with each sensor individually, increasing feature extraction. The proposed method is verified through a series of experiments with seven different induced faults and operation conditions. The results show that the proposed architecture is very accurate for multi-sensor fault detection using vibration time series. Since the experiments are based on real electric motors and faults, these results are promising in real applications.

1. Introduction

Electric motors are a significant supply of energy and also the backbone of many industries. Therefore, it's vital to keep the upkeep of those machines up-to-date to ensure production and job security while they are up and running. Therefore, it's very vital to enhance maintenance techniques.

Various strategies have been used to diagnose the failure of rotating machinery, like oil debris analysis, temperature analysis, acoustic detection, analysis of electrical characteristics, and vibration signal analysis [1–11,41–43].

Oil debris analysis is a technique that evaluates particles in the lubricating oil of machines to predict wear on mechanical components. However, it is extremely difficult to do online monitoring using this technique, as it is necessary to extract oil samples and analyze them in the laboratory, and not all mechanical parts are in contact with lubricating oil [5,6]. Temperature analysis is a technique that uses resistive sensors or thermographic cameras to identify non-standard machine temperatures. However, it is extremely difficult to measure the defect locally through the technique, making the method one more indicator of

global failure [7,8]. Acoustic detection is a very efficient technique that uses acoustic sensors to detect vibration in the air. Although similar to vibration analysis, acoustic analysis is heavily influenced by noise from other equipment in a real application, making it extremely difficult to use [9].

In the same way, analysis of electrical characteristics is a technique that uses sensors to measure motor current and voltage to identify faults. With it, it is possible to identify a wide range of defects online and with little invasion of the machine. However, the technique cannot identify the bearing defect, which is the most common in electric motors [10,11]. Vibration signal analysis is a technique that uses accelerometers and vibrometers to capture machine vibration.

Even machines in good condition generate vibration, and these vibrations are closely linked to periodic events in the operation of the machine, such as rotating shafts, gear teeth, rotating electric fields, etc. The frequency at which these events repeat generates sparks from the source. Therefore, many diagnostic techniques are based on frequency analysis. Despite the many possible approaches, vibration signal analysis is very common and appropriate for the detection and identification of the most common motor failures, being the most suitable for detecting bearing defects [1,12–14]. Fig. 1 compares the techniques for analyzing

* Corresponding author.

E-mail address: ronnyjr@unifei.edu.br (R.F.R. Junior).

Nomenclature	
MCSA	Motor Current Signature Analysis
EPVA	Extended Park's Vector Approach
IPSA	Instantaneous Power Signature Analysis
ANN	Artificial Neural Network
DL	Deep Learning
CNN	Convolution Neural Network
ReLU	Rectified Linear Unit
Avg	Average
Conv	Convolution layer
FC layer	Fully Connected Layer
TP	True positive
TN	True negative
FP	False positive
FN	False negative
t-SNE	t-distributed stochastic neighbor embedding
CBM	Condition-based monitoring

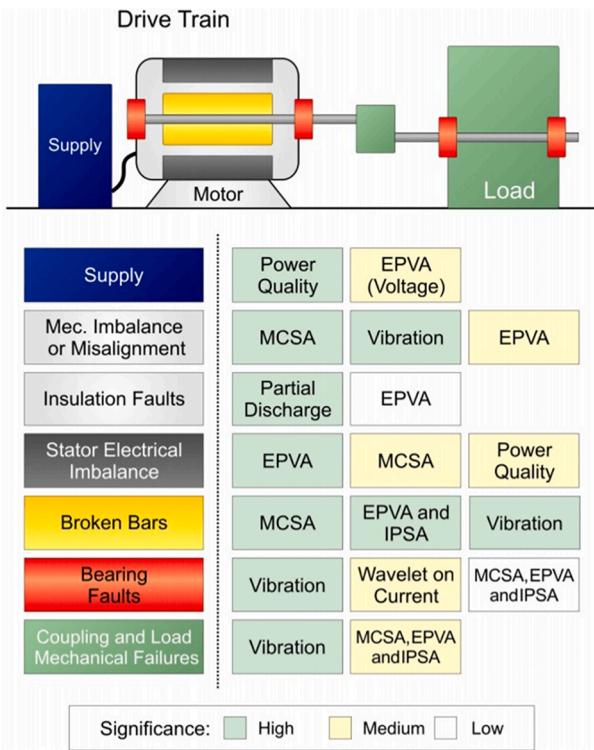


Fig. 1. Comparison between analysis of electrical characteristics and vibration analysis (legend: MCSA: Motor Current Signature Analysis, EPVA: Extended Park's Vector Approach, IPSA: Instantaneous Power Signature Analysis). adapted from [10]

electrical characteristics with vibration analysis (the two main techniques used to detect failures in electrical motors).

However, all of these methods depend on professional knowledge and are time-consuming. Because of the increased complexity of the machine and also the variety of sensors, these strategies are no longer effective.

The first attempt was to use an Artificial Neural Network (ANN). Despite being an established fault diagnosis technology, ANNs have the problem of not being able to handle large amounts of data very well [14,15]. This fact typically implies preprocessing data to arrange it and

extract key features or reduce dimensions to transform it into intelligent or usable data. However, deep learning (DL) methods can be used to overcome these difficulties [16].

Deep learning, one of the fastest-growing fields of machine learning in recent years, is increasingly being used in science because of its powerful capabilities, being progressively used in fault diagnosis. Compared with traditional fault diagnosis strategies, deep learning can use a deep network to extract features from an input sample and run automatic feature extraction and classification through non-linear activation functions in each layer. Thus, deep learning does not require manual mining and does not inherently depend on human intervention or experience [17].

Conventional deep CNNs are designed to work only with 2D data such as images and videos. Therefore, a modified version of 2D CNN called 1D convolutional neural network (1D CNN) has recently been developed for more efficient 1D signal processing without performing preprocessing [13]. Many studies have shown that 1D CNNs are beneficial for certain applications and are therefore preferable to their 2D counterparts for 1D signal processing [18–21].

Consequently, this paper proposed a method that does not require an extremely qualified professional to perform the analysis like traditional methods, which overcomes the difficulty encountered by conventional ANN in working with the raw vibration signal and also overcomes the need for preprocessing of 2DCNN, in addition to being extremely faster. All this without losing accuracy compared to the aforementioned methods. When compared to other studies that use 1D CNN to detect faults in electric motors [17,21–26], the majority of the studies only detect bearing failures; thus, this work goes further and identifies more failures than just bearing faults.

The main objective of this article is to classify faults in electric motors using a 1D CNN through the vibration signal of the two different accelerometers. For this, an independent CNN was used to handle the data from each sensor. Each convolution will be referred to as a convolutional head, thus forming a multi-headed CNN. To validate the method, an experimental study was conducted with different electric motors under seven different operating conditions. The results show that the 1D Multi-head CNN is a fast and effective technique to diagnose faults through vibration signals in the time domain.

The article is organized as follows: **Section 2** introduces the principles of CNN and the differences between 1D CNN and 2D CNN. **Section 3** presents the experimental methodology and flowchart of the proposed method. **Section 4** presents the main results. Finally, **Section 5** concludes the paper.

2. Convolutional neural network (CNN)

CNN is a special ANN with a feed-forward structure. The neural mode of CNNs consists of local connections with the sparse response characteristic of biological neurons [26]. CNN has been successfully applied in many different fields, such as image processing [27,28] and speech recognition [29,30], and signal analysis is a new application of CNN.

2.1. CNN's architecture

A typical CNN includes the input layer, convolutional layer, pooling layer, fully connected layer, and output layer [31].

The CNN input layer is responsible for data processing. This data is usually standardized before being imported into CNNs to improve the operational efficiency and learning performance of the CNNs.

In the convolutional layer, the kernels (filter) transform the output of the previous layer (l) and use a nonlinear activation function to build the output characteristics [31]. The general form of the convolution operation is expressed by Eq. (1) [17,18].

$$y_i^{l+1}(j) = K_i^l \times x^l(j) + b_i^l \quad (1)$$

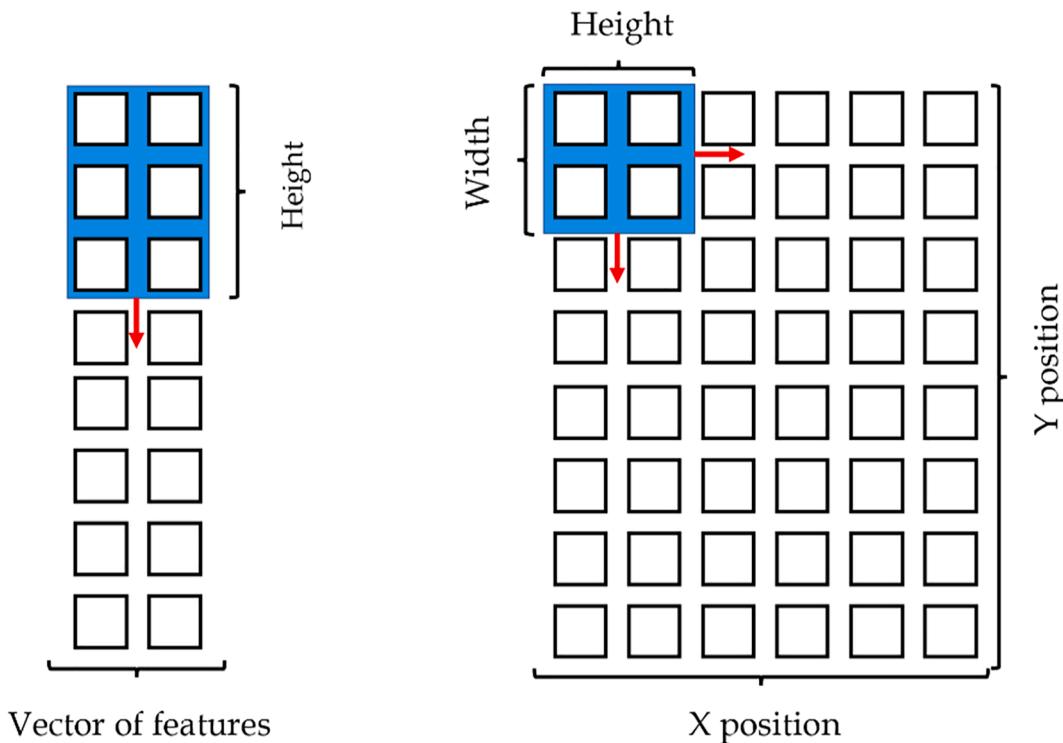


Fig. 2. Comparison between 1D and 2D CNNs filters.

where K_i^l indicates the weights of the i -th ($i = 1, 2, \dots, n$) filter kernel at layer l ; b_i^l indicates the bias of the i -th filter kernel at layer l ; $x^l(j)$ indicates the j -th local region at layer l , and $y_i^{l+1}(j)$ indicates the input of the j -th neuron in frame i of layer " $l + 1$ ". The notation $*$ indicates the dot product of the kernel and local regions.

Then the activation function performs a transformation on the convolution output. The purpose of the activation function is to convert the original multi-dimensional features that are not linearly separable into another space, where they can be separated. In this paper, we have used the Leaky ReLU function of 0.3 α because it has the same characteristics as ReLU (low computational cost and fast training) and can avoid the neuron death problem. The ReLU leakage function can be described by Eq. (2) [17,18]

$$\text{LEAKY RELU} = \phi(x) = \begin{cases} \alpha x & x \leq 0 \\ x & x > 0 \end{cases} \quad \phi'(x) = \begin{cases} \alpha & x \leq 0 \\ 1 & x > 0 \end{cases} \quad (2)$$

Then the pooling layer is usually added. The main function of pooling is to decrease the neural network parameters, thereby reducing computational complexity and the risk of overfitting. In this work, the average pooling method is applied. Often, other pooling layers like maximum pooling or norm pooling can be used. The pooling layer function is determined by Eq. (3) [17,18].

$$P_i^{l+1}(j) = \text{avg}_{(j-1)W+1 \leq i \leq jW} \{q_i^l(t)\} \quad (3)$$

where $q_i^l(t)$ indicates the value of the t -th ($t = 1, 2, \dots, m$) neuron in the i -th ($i = 1, 2, \dots, n$) feature at layer l , ($t \in [(j-1)W + 1, jW]$); W indicates the width of the pooling area, and $P_i^{l+1}(j)$ indicates the value of the neuron at layer " $l + 1$ ".

A fully connected layer consists of a simple feed-forward neural network. A fully connected layer expands the output of the final pooling layer into a one-dimensional vector. It also serves as an input to a fully connected layer, connecting inputs and outputs [24]. The formula for a fully connected layer is defined in Eq. (4) [17,18].

$$z^{l+1}(j) = f \left(\sum_{i=1}^m \sum_{t=1}^n W_{ij}^l a_i^l(t) + b_j^l \right) \quad (4)$$

where W_{ij}^l indicates the weighting between the t -th ($t = 1, 2, \dots, m$) neuron in the i -th ($i = 1, 2, \dots, n$) feature at layer l and the j -th ($j = 1, 2, \dots, n$) neuron at layer " $l + 1$ "; $z^{l+1}(j)$ indicates the logits value of the j -th neuron at layer " $l + 1$ "; b_j^l indicates the network offset; $a_i^l(t)$ indicates the output value of t -th neuron in the i -th feature at the previous layer l ; $f(\cdot)$ indicates the activation function Leaky ReLU.

Finally, the last layer is the output layer. The most common function of classification problems is the Softmax function. The softmax classifier is a common multiclass classifier derived from logistic regression [17]. This function takes a vector constructed to evaluate any real value and compresses it into a vector composed of values from 0 to 1. The softmax function is defined by Eq. (5) [17,18].

$$Q(j) = \text{softmax}(z^o(j)) = \frac{e^{z^o(j)}}{\sum_{k=1}^M e^{z^o(k)}} \quad (5)$$

where $z^o(j)$ indicates the logits of the output of the j -th ($j = 1, 2, \dots, n$) neuron at output layer, and M indicates the total number of categories.

2.2. Comparison between 1D and 2D CNNs

1D and 2D CNN share almost the same architecture, with the exception of the filter sliding mechanism [31]. In 1D CNN, the filter slides in vertical order (height) to extract the features, and the height determines the number of sample point for the convolutional operation. In contrast, the 2D CNN filter slides the entire matrix horizontally and vertically (height and width). The height and width of the 2D filter determine the range of convolution operations for each step [31]. Fig. 2 compares 1D and 2DCNN.

As mentioned in Section 1, 1D CNNs have advantages over their 2D counterparts and are preferred for certain applications, especially when processing 1D signals. This is due to the following reasons [18].

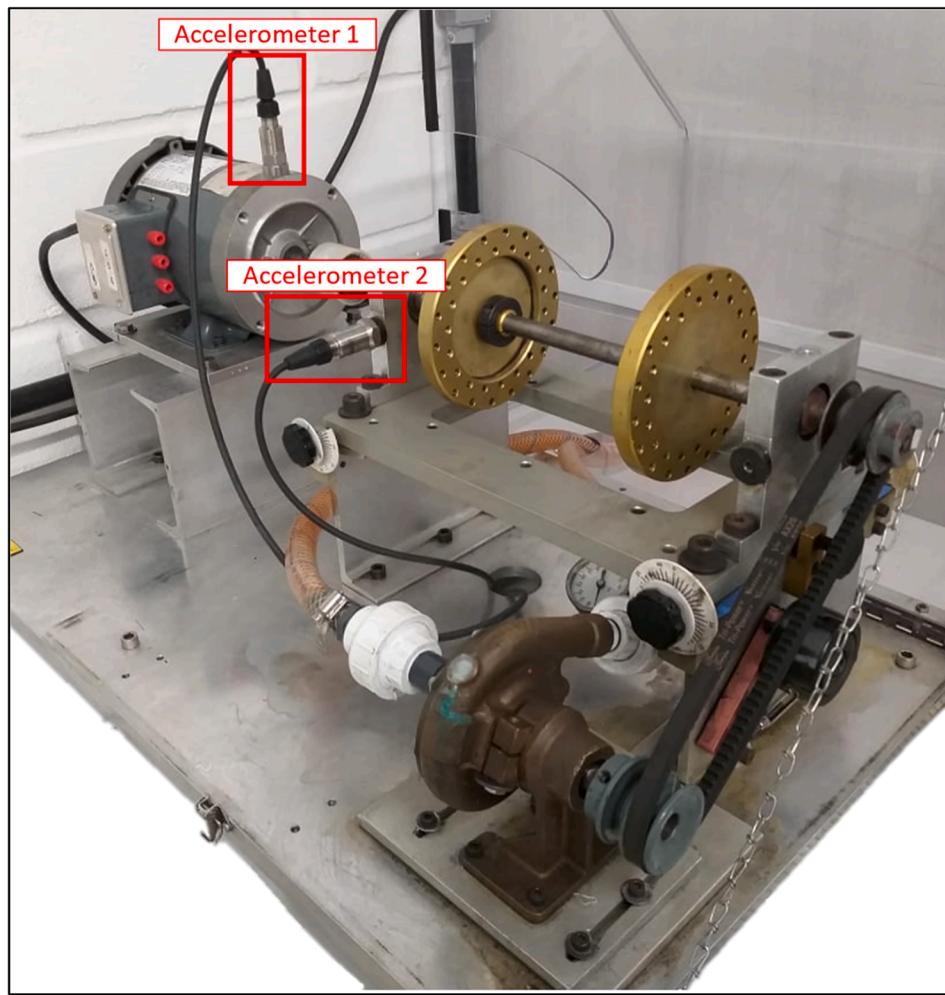


Fig. 3. Experimental setup.

- There is a big difference in the complexity of 1D and 2D convolution calculations. As CNN1D has one dimension less than CNN2D, the computational complexity is much lower under the same conditions (same configuration, same network, same hyperparameters).
- Most CNN 1D applications have a compression configuration consisting of one or two hidden CNN layers and few parameters in the network (usually less than 10,000). Although most 2DCNN applications have 1 million parameters (usually above 10 million) and many hidden layers. Of course, neural networks have a lot to do with training and implementation.
- Due to its low computational requirements, CNN1D is especially suitable for low-cost real-time applications on mobile and portable devices.
- As the 1D CNN works with the signal in time, it is not necessary to process the signal. In the case of a 2D CNN, it would be necessary to perform a preprocessing to transform the signal into something compatible with the 2D CNN.

2.3. Multi-head CNNs

In a multi-head convolutional neural network, each time series is handled independently; that is, each time will be handled by a convolutional head [25,32]. In many industrial situations, machines are installed with multiple sensors independent of each other, so there may be no correlation [25,32]. In this way, we can analyze multiple sensors with the same CNN and extract the individual characteristics of each sensor. In addition, it can be choosing a sliding window that performs

better in the different signals collected.

2.4. Tricks of CNNs

There are two tips that can be used to improve training and, therefore, the adaptability of CNN. The first is to use batch normalization. That is, a technique that normalizes the input of a layer by keeping the mean close to zero and the standard deviation of the output close to 1. This technique is designed to reduce the shift of internal covariance and accelerate the training of deep neural networks [26]. Batch normalization can be described by Eq. (6) [26].

$$\hat{x}^{(k)} = \frac{x^{(k)} - E[x^{(k)}]}{\sqrt{Var[x^{(k)}]}} \quad (6)$$

where $\hat{x}^{(k)}$ the normalized value of the k th ($k = 1, 2, \dots, n$) hidden unit. $E[x^{(k)}]$ is the expectation of the k th units values also called the mean value and $Var[x^{(k)}]$ is the variance of the k th hidden unit.

The second is using Early Stop, an algorithm to stop training when a monitored metric stops improving. Assuming the common goal of training is to minimize the loss, the algorithm will check at the end of every epoch whether the loss is no longer decreasing after the N epochs chosen. After N epochs, if the loss does not decrease, the algorithm restores the weights of the best epoch.

Table 1
Marathon motor nameplate.

Parameter	60 Hz	50 Hz
Power (HP)	0.50	0.33
Rotation (RPM)	3450	2850
Voltage (V)	208–230/460	190/380
Amperage (A)	2.1–2.2/1.1	2.0/1.0
Service Factor	1.15	1.15

3. Experimental methodology

3.1. Experimental setup

All tests were carried out at PS Soluções in Itajuba (Brazil). The test bench used (Fig. 3) consisted of a 0.5 HP induction motor directly connected to the SpectraQuest machine failure simulator [33]. Accelerometer 1 measures vibration in the y direction and accelerometer 2 measures vibration in the x direction. Table 1 also shows the main characteristics of the motor at 60 Hz and 50 Hz.

The experimental bench is composed of several electric motors with different operating conditions. Normal, bent shaft, broken bar,

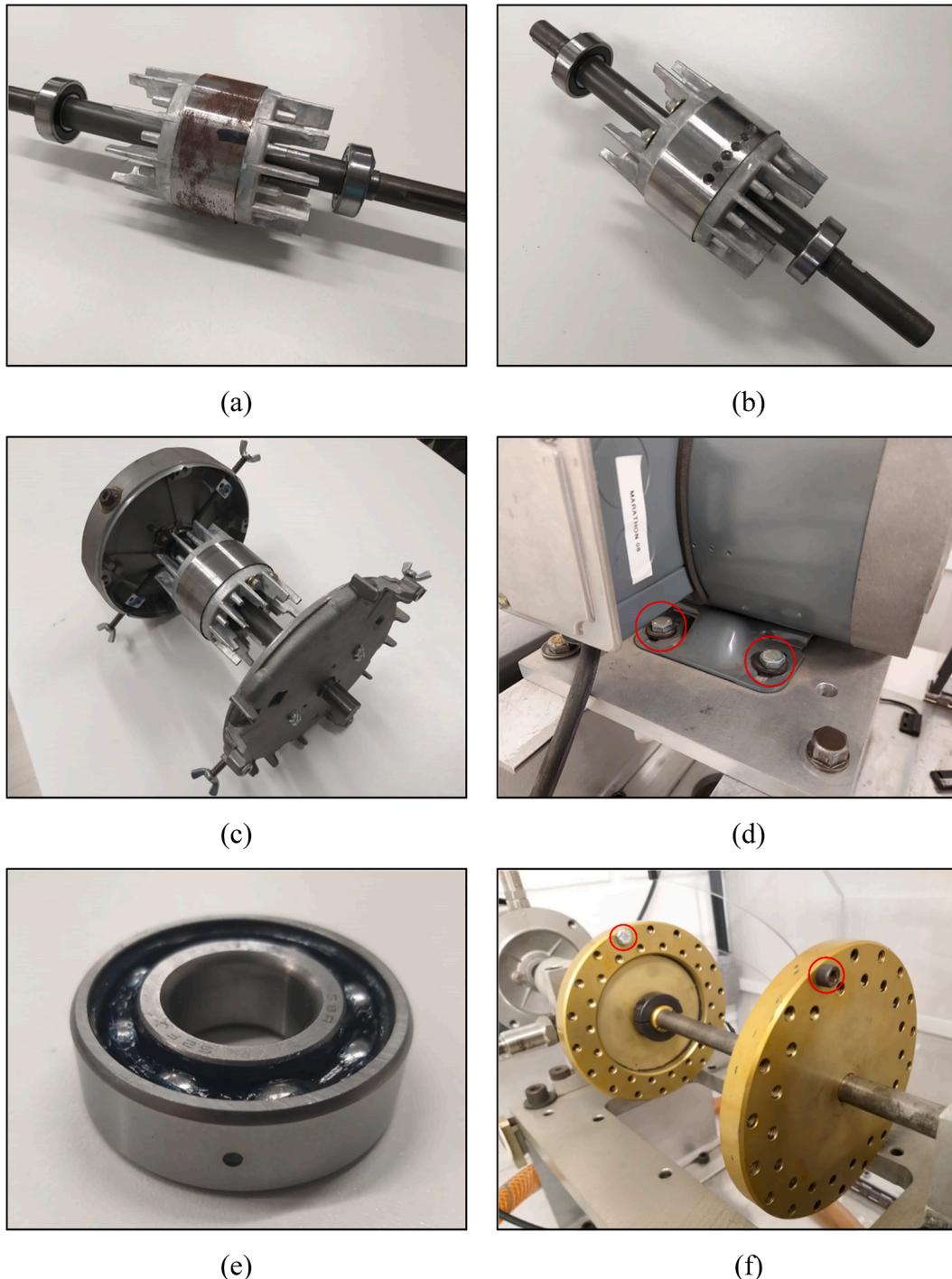
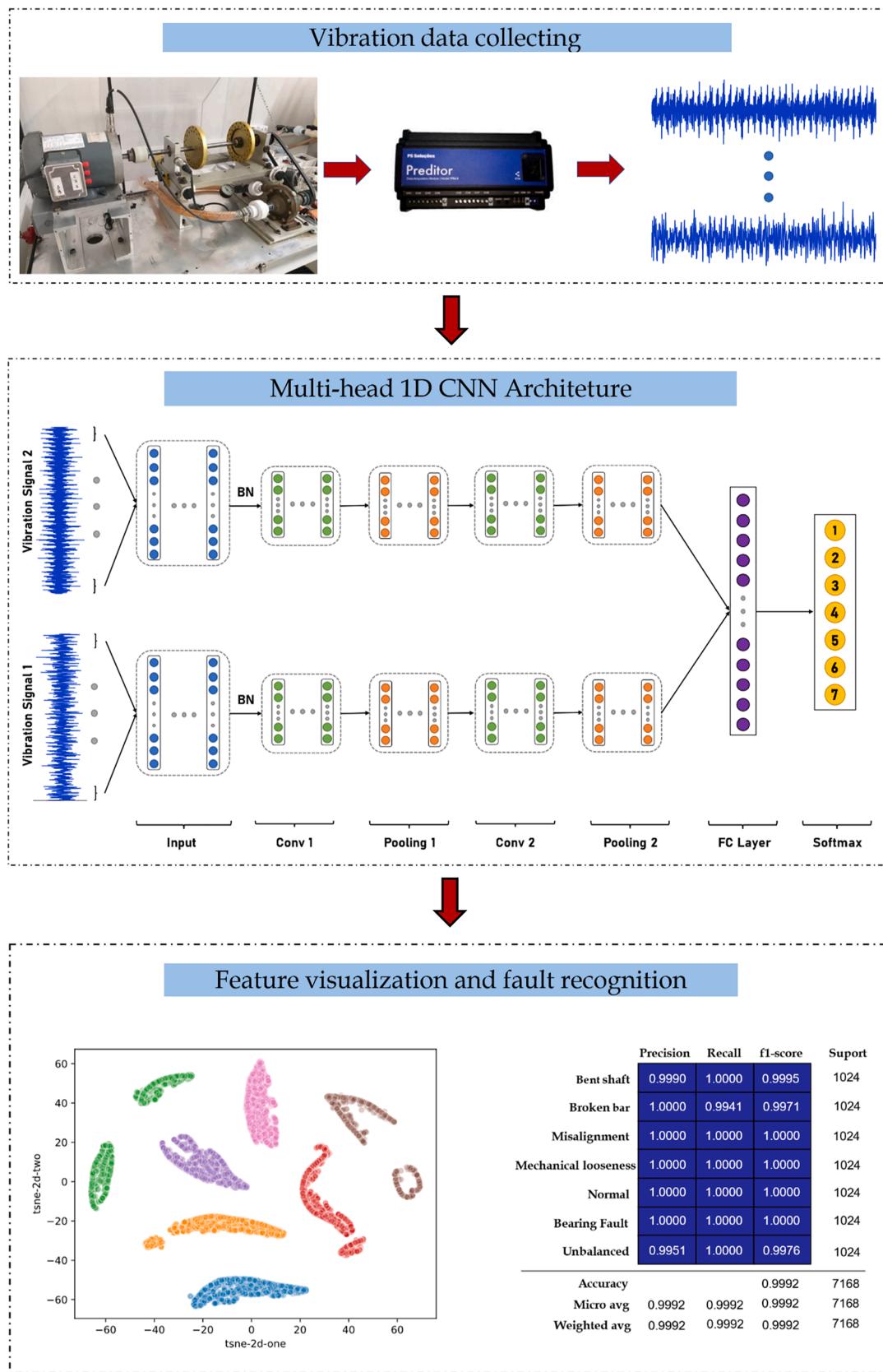


Fig. 4. Induced Fault conditions: (a) bent shaft, (b) broken bar, (c) misalignment, (d) mechanical looseness, (e) bearing fault and (f) Unbalanced.

**Fig. 5.** Proposed method flowchart.

misalignment, mechanical looseness, bearing fault, and unbalanced are examples of these conditions.

- **Normal condition** is the electric motor in a healthy state, without any fault.
- **Bent shaft**, as shown in Fig. 4(a), has a shaft that is slightly bent, causing a scratch between the rotor and the stator. Overloads and coupling situations are the most common causes of this problem.
- **Broken bar** refers to the motor whose squirrel cage rotor bar is broken. This fault is more common on bigger electric motors. As shown in Fig. 4(b) this fault is simulated by drilling holes in the rotor.
- **Misalignment** refers to the motor whose shaft is misaligned. This fault occurs due to bad installation or overloads. Fig. 4(c) shows how this defect is induced.
- **Mechanical looseness** refers to the motor whose screws are loosening. This fault commonly occurs due to bad installation, but high vibration may also cause this fault. Fig. 4(d) shows which screws are loosened.
- **Bearing fault** refers to the motor whose bearing is damaged. This is the most common fault in electric motors. Fig. 4(e) shows the bearing fault in the outer race.
- **Unbalanced** refers to the motor whose load is unbalanced. This fault is very common in electric motors, being the reason for other faults. This fault is illustrated in Fig. 4(f).

These conditions cover the most common faults in electric motors. Several studies [10,14,34–36] verify that the chosen failures are in fact the most common faults in electric motors. We carried out 4 experiments runs with all operating conditions, collecting 28 signals in each accelerometer.

3.2. Vibration signal acquisition

The two vibration signals are collected by the IMI uniaxial accelerometer. The accelerometer has a frequency range of ± 3 dB, a measurement of ± 50 g, a sensitivity of 100 mV/g, and resonance at 25 kHz. These characteristics make the sensor suitable for vibration measurement, since the frequency of the received signal is low and the mass of the motor is much larger than the mass of the accelerometer. The first accelerometer is fixed to the motor using an installed threaded base. The second is also fixed by a threaded base installed in the bearing housing. The acquisition system (Preditor®) has a sampling frequency of 11718.5 Hz and an acquisition window of 22.37 s, collecting 262,144 sample points.

3.3. Procedures of the proposed method

In this paper, a multi-head 1D CNN based on the vibration signals algorithm is proposed for the diagnosis of faults in electric motors. This algorithm uses the fused data set of vibration signals in two channels as the input and then intelligently classifies these signals to identify six different faults.

Fig. 5 shows a detailed flowchart of the proposed method. The first step is the acquisition of all vibration signals under different operation conditions. We collected four signals for each operational condition in each accelerometer, with 75% of the signals for training and 25% for testing. After that, the signals are split into sample windows, and each sample window has 256 sample points. After the split, the data can be imported into the 1D CNN.

In each head, the 1D CNN has batch normalization, two convolutional layers, and two pooling layers; one fully connected layer and one softmax layer. The convolutional filter has 16 and the kernel size is 5. The pooling filter is 16 and the pool size is 2. The node number is 256 at the fully connected layers, and lastly, there are 7 outputs at the softmax layer, corresponding to the 7 operation conditions in the experiment.

The network will be trained until, after 10 epochs, there is no

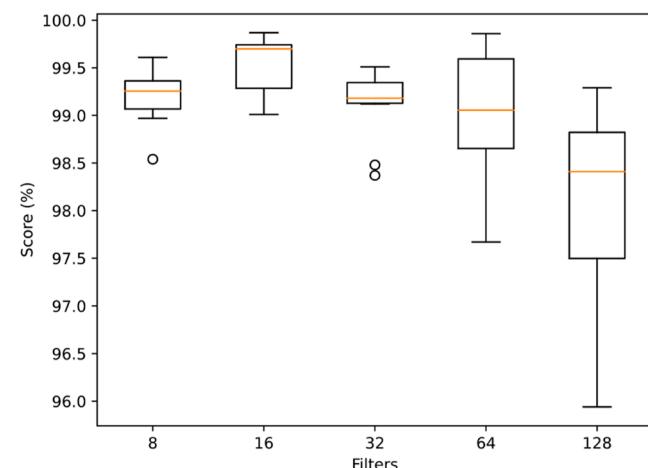


Fig. 6. Accuracy score under different number of filters.

decrease in the test loss. The visualization results and classification report are employed to verify the performance of the proposed method. All features used in the proposed method were developed in the Python language using the Keras API of TensorFlow [37] and Scikit-learn [38].

4. Results and discussion

In this section, we discuss the results obtained by the proposed method. However, before that, an analysis will be performed to select the best 1D CNN hyperparameters. The accuracy of the proposed method is rated by the loss function, that is, the gap between the predicted value and the real value. As mentioned before, we use the Early stop algorithm to aid in the selection of the best epoch of training (lower loss function). This way, the epoch parameter will not need to be optimized and analyzed. Finally, the results of 1D CNN are presented.

4.1. Hyperparameter tuning

Tuning hyperparameters involves choosing the best set of hyperparameters for a learning algorithm. Hyperparameters are valuable parameters used to control the learning process. Finding all the optimal hyperparameters is an arduous process, so, usually some main parameters are chosen to perform the tuning. In this paper, the hyperparameters selected for tuning are number of filters, learning rate, and batch size.

Tuning hyperparameters involves choosing the best set of hyperparameters for a learning algorithm. Hyperparameters are valuable parameters used to control the learning process. Finding all the optimal hyperparameters is an arduous process, so usually some main parameters are chosen to perform the tuning. In this paper, the hyperparameters selected for tuning are the number of filters, learning rate, and batch size.

Thus, we conducted a first attempt, where the learning rate was set at 0.01 and the batch size was 64 (algorithm default values), and the number of filters was changed in each run. In all, we made 10 runs for each filter value. Fig. 6 shows the results in a boxplot form. It was observed the presence of outliers for the number of filters of 8 and 32, in addition to a high variability for 64 and 128, characteristics of unstable networks.

Therefore, based on the evidence, the number of filters was set to 16 because of the high score percentage and network stability. Another analysis that can be done is that the greater the number of filters, the longer it takes for the network to perform an iteration. In the second step, the number of filters was fixed at 16, the batch size was equal to 64, and the learning rate was changed in each run. In the first test, we made 10 runs for each learning rate value. Fig. 7 shows the results in a boxplot

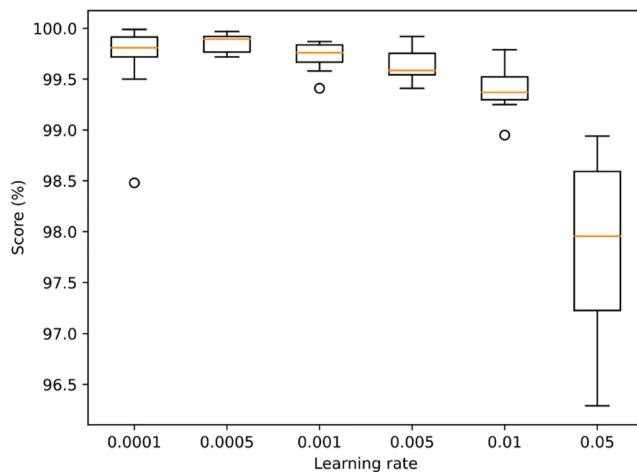


Fig. 7. Accuracy score under different learning rates.

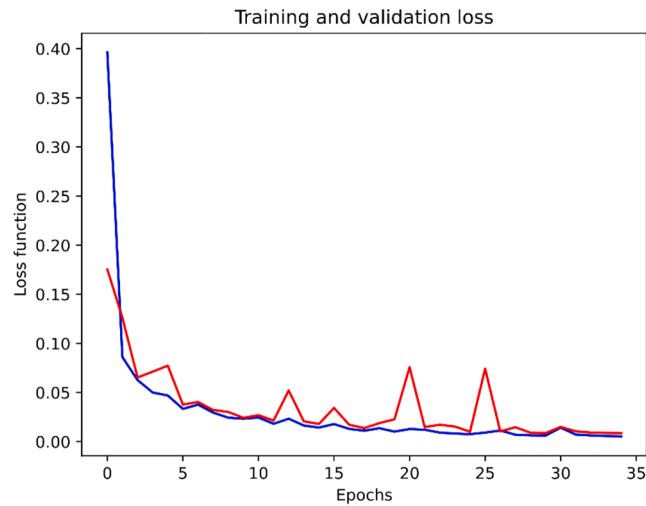


Fig. 10. Training and validation loss (legend: — Test and — Train).

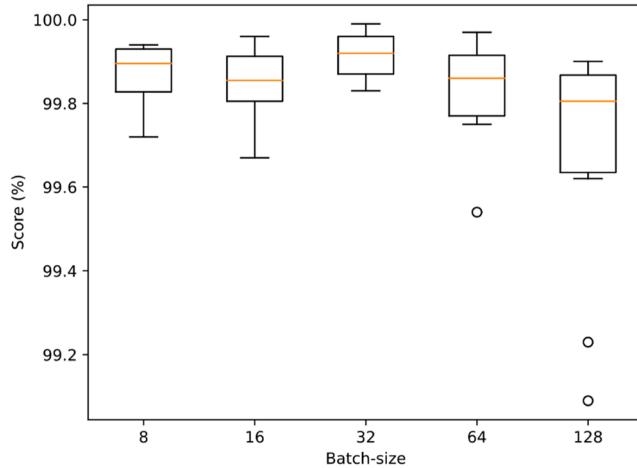


Fig. 8. Accuracy score under different batch sizes.

	Precision	Recall	f1-score	Support
Bent shaft	0.9990	1.0000	0.9995	1024
Broken bar	1.0000	0.9941	0.9971	1024
Misalignment	1.0000	1.0000	1.0000	1024
Mechanical looseness	1.0000	1.0000	1.0000	1024
Normal	1.0000	1.0000	1.0000	1024
Bearing Fault	1.0000	1.0000	1.0000	1024
Unbalanced	0.9951	1.0000	0.9976	1024
Accuracy			0.9992	7168
Micro avg	0.9992	0.9992	0.9992	7168
Weighted avg	0.9992	0.9992	0.9992	7168

Fig. 9. Classification report.

form. It is worth noting that there are some outliers for values of 0.0001, 0.001, and 0.01. In addition, for a value of 0.05, there is great variability.

After analyzing the results, the learning rate was set to 0.0005 to

have the highest score combined with low variability. Another analysis that can be done is that the value of the learning rate does not affect the CNN training time.

Concluding the tuning process, the number of filters was set to 16, the learning rate to 0.005, and the batch size was changed in each run. We did the same as the other tests, that is, 10 runs for each batch size value. Fig. 8 shows the results in a boxplot form. It was observed that the presence of outliers for the batch sizes of 64 and 128, besides a high variability for all batch sizes, except for the batch size of 32, was observed. Therefore, the batch size was set to 32 because of the highest score and low variability. Unlike the number of filters, the smaller the batch size, the longer it takes to perform an iteration. Based on the results, the optimal hyperparameters chosen were a number of filters of 16, a learning rate of 0.0005, and a batch size of 32.

4.2. Feature visualization and fault recognition

The test classification report is shown in Fig. 9 for detailed classification results. The classification report shows four measures: Accuracy (Eq. (7)), Precision (Eq. (8)), Recall (Eq. (9)) and f1 score (Eq. (10)).

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (8)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (9)$$

$$f1score = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (10)$$

where TP indicates true positive, TN indicates true negative, FP indicates false positive and FN indicates false negative.

A f1 score is often more realistic than precision. This is especially true if the distribution of the classes is not uniform. Accuracy works better if the cost of a false negative is equal to the cost of a false positive. If there is a significant difference between the false positives and the false negative cost, both Precision and Recall should be verified.

The recognition accuracy of the operation conditions is 99.92%. The misalignment, mechanical looseness, normal and bearing fault conditions reached 100% accuracy, but the bent shaft, broken bar, and unbalanced had a minor error. However, it does not compromise the entire network classification, as this error is very small. This shows that a multi-head 1D CNN is a powerful method to identify electric motor

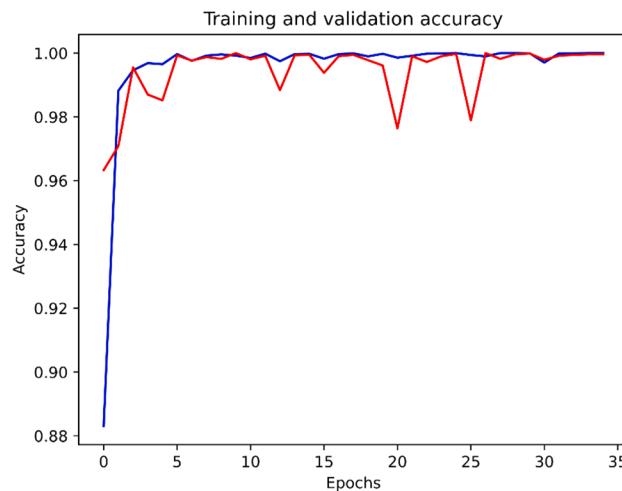


Fig. 11. Training and validation accuracy (legend: — Test and — Train).

faults with a fast and reliable diagnosis. Another advantage of 1D CNN compared to other types of networks is its low computational cost, making it very useful for real-time applications.

Network performance can be measured by both loss and accuracy. Fig. 10 shows the evolution of the loss during each epoch of the training and test. The loss is measured using the categorical crossentropy loss function. Categorical crossentropy is a loss function that is used in multi-class classification tasks. These are tasks where a sample can only belong to one out of many possible categories, and the model must decide which one and can be calculated as shown in Eq. (11) [39].

$$\text{Loss} = - \sum_{i=1}^N y_i \times \log \hat{y}_i \quad (11)$$

where \hat{y}_i is the i -th scalar value in the model output, y_i is the corresponding target value, and N is the output size in the model output.

Fig. 11 shows the evolution of accuracy during each epoch of the training and testing. The proposed method rapidly converges and builds an accurate model. Also, from both loss and accuracy, the performance of the early stop can be seen.

To better understand the process of feature extraction, the t-distributed stochastic neighbor embedding (t-SNE) [40] was used to visualize the learning characteristics of the CNN. t-SNE is a technique to

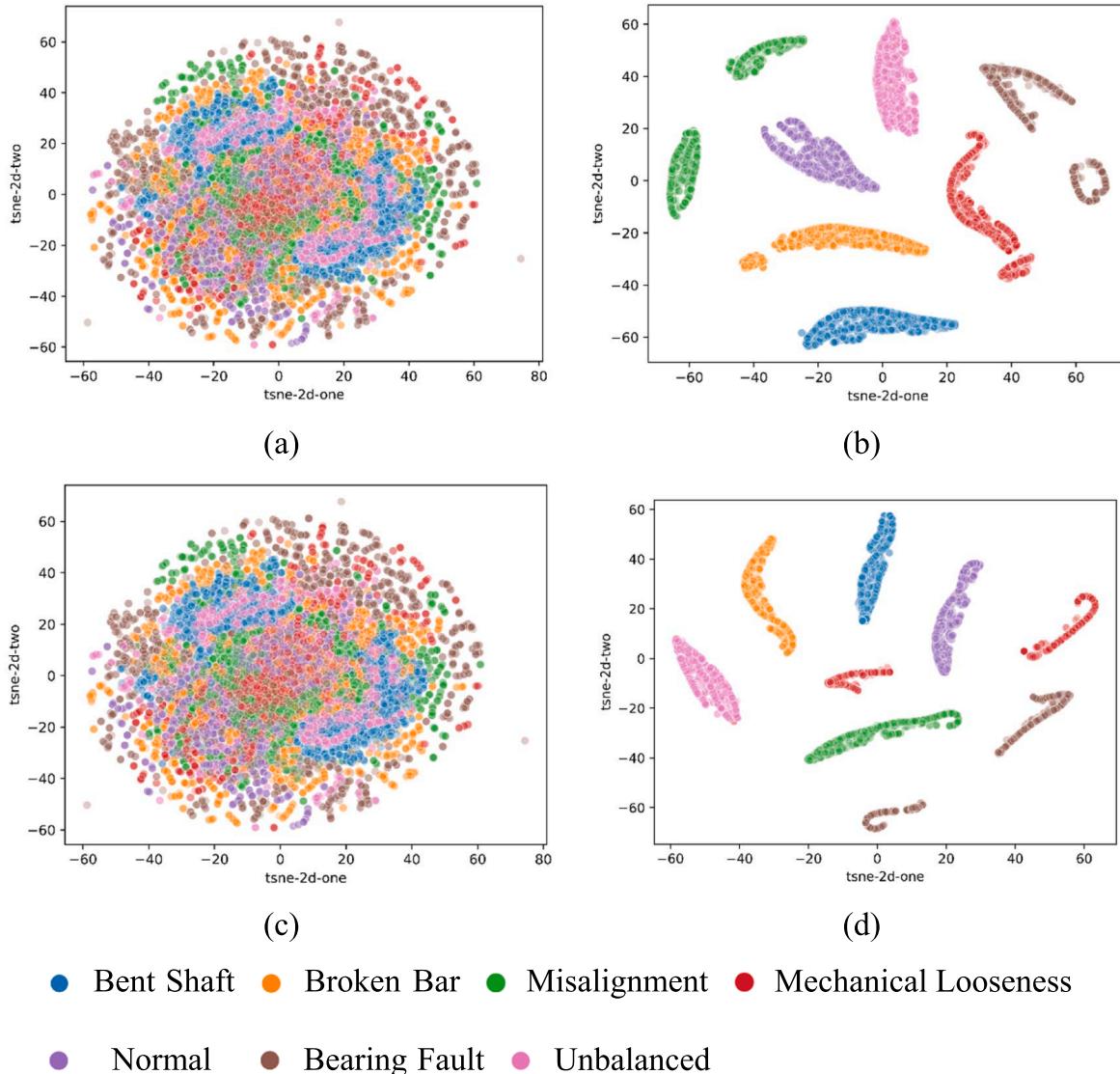


Fig. 12. (a) Training input layer, (b) Training output layer, (c) Test input layer, (d) Test output layer.

view high-dimensional data. It transforms similarities between data points into general probabilities and attempts to minimize Kullback-Leibler divergences between low-dimensional integrals and general probabilities of high-dimensional data. t-SNE provides ideas or insights into how data is organized in large-dimensional space. Therefore, the output of the algorithm gives visibility to high-dimensional data by projecting it into a 2-dimensional space. From Fig. 12, we can see that at the entrance to the network, the different types of faults were all grouped together, making it impossible to identify. At the end of the network, faults of the same type were grouped together, and the different fault types were separated. This shows that the proposed method has powerful feature extraction ability.

5. Conclusions

In this paper, a method based on vibration signals and multi-head 1D-CNN for fault diagnosis on electric motors is designed. The vibration signals of electric motors are measured in two directions and then fed simultaneously to the multi-head 1D-CNN simultaneously to train it. The method is verified experimentally with a setup of 7 different conditions. We achieve a network with 99.92% accuracy by tuning the hyperparameter. Also, the t-SNE method is used to visualize the CNN learning process. Another positive point is that 1D networks are extremely fast to train and test (compared to conventional 2D networks), making them very useful for real-time applications. These results show that the proposed method can successfully extract features and diagnose faults in an electric motor. In this way, it is possible to follow in real-time the condition of the motors and do condition-based monitoring (CBM). Future work will include different types of sensors to the multi-head 1D CNN to improve fault recognition and add other types of faults.

CRediT authorship contribution statement

Ronny Francis Ribeiro: Conceptualization, Software, Formal analysis, Validation, Investigation, Data curation, Writing – original draft, Visualization. **Isac Antônio dos Santos Areias:** Methodology, Investigation, Writing – review & editing. **Mateus Mendes Campos:** Software, Writing – review & editing. **Carlos Eduardo Teixeira:** Software, Writing – review & editing. **Luiz Eduardo Borges da Silva:** Resources, Project administration, Funding acquisition. **Guilherme Ferreira Gomes:** Conceptualization, Writing – review & editing, Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

the authors would like to acknowledge the support from PS Soluções, that provided the experimental laboratory to carry out this manuscript.

Funding

The authors would like to acknowledge the financial support from the Brazilian agency CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) and FAPEMIG (Fundação de Amparo à Pesquisa do Estado de Minas Gerais - APQ-00385-18).

References

- [1] Roozbeh Razavi-Far, et al., Information fusion and semi-supervised deep learning scheme for diagnosing gear faults in induction machine systems, IEEE Trans. Ind. Electron. 66 (8) (2018) 6331–6342, <https://doi.org/10.1109/TIE.2018.2873546>.
- [2] R.B. Randal, *Vibration-based Condition Monitoring: Industrial, Aerospace and Automotive Applications*, John Wiley & Sons, 2011.
- [3] Hui Wang, et al., A new intelligent bearing fault diagnosis method using SDP representation and SE-CNN, IEEE Trans. Instrum. Measur. 69 (5) (2019) 2377–2389, <https://doi.org/10.1109/TIM.2019.2956332>.
- [4] R.F.R. Junior, I.A.D.S. Areias, G.F. Gomes, Fault detection and diagnosis using vibration signal analysis in frequency domain for electric motors considering different real fault types, Sensor Rev. 41 (3) (2021) 311–319, <https://doi.org/10.1108/SR-02-2021-0052>.
- [5] V.T. Didake, D.H. Burande, Oil debris analysis for condition monitoring of an IC engine, Int. Res. J. Multidiscip. Stud. 2 (3) (2016).
- [6] Paula J. Dempsey, Gary Kreider, Thomas Fichter, Investigation of tapered roller bearing damage detection using oil debris analysis, in: 2006 IEEE Aerospace Conference, IEEE, 2006, pp. 11. <https://doi.org/10.1109/AERO.2006.1656082>.
- [7] Davide Astolfi, Francesco Castellani, Ludovico Terzi, Fault prevention and diagnosis through SCADA temperature data analysis of an onshore wind farm, Diagnostyka 15 (2014).
- [8] Y. Xie, Y. Wang, 3D temperature field analysis of the induction motors with broken bar fault, Appl. Therm. Eng. 66 (1–2) (2014) 25–34, <https://doi.org/10.1016/j.applthermaleng.2014.02.008>.
- [9] T. Toutountzakis, C.K. Tan, D. Mba, Application of acoustic emission to seeded gear fault detection, NDT and E Int. 38 (1) (2005) 27–36.
- [10] Erik Leandro Bonaldi et al., Predictive maintenance by electrical signature analysis to induction motors, in: Induction Motors-Modelling and Control. IntechOpen, 2012.
- [11] Ian Culbert, John Letal, Signature analysis for online motor diagnostics: early detection of rotating machine problems prior to failure, IEEE Ind. Appl. Magazine 23 (4) (2017) 76–81, <https://doi.org/10.1016/j.ndteint.2004.06.008>.
- [12] C. Wu, P. Jiang, C. Ding, F. Feng, T. Chen, Intelligent fault diagnosis of rotating machinery based on one-dimensional convolutional neural network, Comput. Ind. 108 (2019) 53–61, <https://doi.org/10.1016/j.compind.2018.12.001>.
- [13] J. Chen, Z. Li, J. Pan, G. Chen, Y. Zi, J. Yuan, B. Chen, Z. He, Wavelet transform based on inner product in fault diagnosis of rotating machinery: a review, Mech. Syst. Sig. Process. 70–71 (2016) 1–35, <https://doi.org/10.1016/j.ymssp.2015.08.023>.
- [14] R.F.R. Junior, F.A. de Almeida, G.F. Gomes, Fault classification in three-phase motors based on vibration signal analysis and artificial neural networks, Neural Comput. Appl. 32 (18) (2020) 15171–15189, <https://doi.org/10.1007/s00521-020-04868-w>.
- [15] I. Triguero, D. García-Gil, J. Maillo, J. Luengo, S. García, F. Herrera, Transforming big data into smart data: An insight on the use of the k-nearest neighbors' algorithm to obtain quality data, Wiley Interdisciplinary Rev. Data Min. Knowl. Disc. 9 (2) (2019), <https://doi.org/10.1002/widm.2019.9.issue-210.1002/widm.1289>.
- [16] Long Wen, et al., A new convolutional neural network-based data-driven fault diagnosis method, IEEE Trans. Ind. Electron. 65 (7) (2017) 5990–5998, <https://doi.org/10.1109/TIE.2017.2774777>.
- [17] X. Wang, D. Mao, X. Li, Bearing fault diagnosis based on vibro-acoustic data fusion and 1D-CNN network, Measurement 173 (2021) 108518, <https://doi.org/10.1016/j.measurement.2020.108518>.
- [18] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, D.J. Inman, 1D convolutional neural networks and applications: a survey, Mech. Syst. Signal Process. 151 (2021) 107398, <https://doi.org/10.1016/j.ymssp.2020.107398>.
- [19] O. Avci, O. Abdeljaber, S. Kiranyaz, M. Hussein, D.J. Inman, Wireless and real-time structural damage detection: a novel decentralized method for wireless sensor networks, J. Sound Vib. 424 (2018) 158–172, <https://doi.org/10.1016/j.jsv.2018.03.008>.
- [20] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, D.J. Inman, Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks, J. Sound Vib. 388 (2017) 154–170, <https://doi.org/10.1016/j.jsv.2016.10.043>.
- [21] Turker Ince, et al., Real-time motor fault detection by 1-D convolutional neural networks, IEEE Trans. Ind. Electron. 63 (11) (2016) 7067–7075, <https://doi.org/10.1109/TIE.2016.2582729>.
- [22] S. Kiranyaz, A. Gastli, L. Ben-Brahim, N. Al-Emadi, M. Gabbouj, Real-time fault detection and identification for MMC using 1-D convolutional neural networks, IEEE Trans. Ind. Electron. 66 (11) (2019) 8760–8771, <https://doi.org/10.1109/TIE.2018.2833045>.
- [23] D. Kolar, D. Lisjak, M. Pajak, D. Pavković, Fault diagnosis of rotary machines using deep convolutional neural network with wide three axis vibration signal input, Sensors 20 (14) (2020) 4017, <https://doi.org/10.3390/s20144017>.
- [24] L. Eren, T. Ince, S. Kiranyaz, A generic intelligent bearing fault diagnosis system using compact adaptive 1D CNN classifier, J. Signal Process. Syst. 91 (2) (2019) 179–189, <https://doi.org/10.1007/s11265-018-1378-3>.
- [25] O. Abdeljaber, S. Sassi, O. Avci, S. Kiranyaz, A.A. Ibrahim, M. Gabbouj, Fault detection and severity identification of ball bearings by online condition monitoring, IEEE Trans. Ind. Electron. 66 (10) (2019) 8136–8147, <https://doi.org/10.1109/TIE.2018.2886789>.
- [26] W. Huang, J. Cheng, Y.u. Yang, G. Guo, An improved deep convolutional neural network with multi-scale information for bearing fault diagnosis, Neurocomputing 359 (2019) 77–92, <https://doi.org/10.1016/j.neucom.2019.05.052>.
- [27] D.-T. Hoang, H.-J. Kang, Rolling element bearing fault diagnosis using convolutional neural network and vibration image, Cognit. Syst. Res. 53 (2019) 42–50, <https://doi.org/10.1016/j.cogsys.2018.03.002>.

- [28] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, *Adv. Neural Inform. Process. Syst.* 60 (6) (2017) 84–90, <https://doi.org/10.1145/3065386>.
- [29] O. Abdel-Hamid, A.-R. Mohamed, H. Jiang, L.i. Deng, G. Penn, D. Yu, Convolutional neural networks for speech recognition, *IEEE/ACM Trans. Audio Speech Lang. Process.* 22 (10) (2014) 1533–1545, <https://doi.org/10.1109/TASLP.2014.2339736>.
- [30] Zhengwei Huang et al., Speech emotion recognition using CNN, in: Proceedings of the 22nd ACM International Conference on Multimedia, 2014, pp. 801–804. <https://doi.org/10.1145/2647868.2654984>.
- [31] Q. Chao, J. Tao, X. Wei, Y. Wang, L. Meng, C. Liu, Cavitation intensity recognition for high-speed axial piston pumps using 1-D convolutional neural networks with multi-channel inputs of vibration signals, *Alexandria Eng. J.* 59 (6) (2020) 4463–4473, <https://doi.org/10.1016/j.aej.2020.07.052>.
- [32] M. Canizo, I. Triguero, A. Conde, E. Onieva, CANIZO, Mikel et al. Multi-head CNN-RNN for multi-time series anomaly detection, *Neurocomputing* 363 (2019) 246–260, <https://doi.org/10.1016/j.neucom.2019.07.034>.
- [33] Spectra Quest, Inc., Machinery Fault Simulator, 2021. <http://spectraquest.com/machinery-fault-simulator/details/mfs>.
- [34] A. Choudhary, D. Goyal, S.L. Shimi, A. Akula, Condition monitoring and fault diagnosis of induction motors: a review, *Arch. Comput. Methods Eng.* 26 (4) (2019) 1221–1238, <https://doi.org/10.1007/s11831-018-9286-z>.
- [35] A. Abraman, situação da Manutenção no Brasil, in: *Anais do 26º Congresso Brasileiro de Manutenção*, Curitiba, Abraman, 2011.
- [36] J. Zarei, M.A. Tajeddini, H.R. Karimi, Vibration analysis for bearing fault detection and classification using an intelligent filter, *Mechatronics* 24 (2) (2014) 151–157, <https://doi.org/10.1016/j.mechatronics.2014.01.003>.
- [37] François Chollet et al., Keras: Deep learning library for theano and tensorflow, 2021. <https://keras.io>.
- [38] Fabian Pedregosa, et al., Scikit-learn: machine learning in Python, *J. Mach. Learn. Res.* 12 (2011) 2825–2830.
- [39] Pushparaja Murugan, Implementation of deep convolutional neural network in multi-class categorical image classification. arXiv preprint arXiv:1801.01397, 2018.
- [40] L. van der Maaten, G. Hinton, ‘Visualizing data using t-SNE’, *J. Mach. Learn. Res.* 9 (2008) 2579–2605.
- [41] A. Rohani Bastami, A. Aasi, H.A. Arghand, Estimation of remaining useful life of rolling element bearings using wavelet packet decomposition and artificial neural network, *Iran. J. Sci. Technol. Trans. Electr. Eng.* 43 (S1) (2019) 233–245.
- [42] Aref Aasi, et al., Experimental investigation on time-domain features in the diagnosis of rolling element bearings by acoustic emission, *J. Vibrat Control* (2021), p. 10775463211016130.
- [43] R. Tabatabaei, A. Aasi, S.M. Jafari, E. Ciulli, Experimental investigation of the diagnosis of angular contact ball bearings using acoustic emission method and empirical mode decomposition, *Adv. Tribol.* 2020 (2020) 1–14.