

# Image-Processing-Based Intelligent Defect Diagnosis of Rolling Element Bearings Using Spectrogram Images

Syed Muhammad Tayyab, Steven Chatterton \* and Paolo Pennacchi

Department of Mechanical Engineering, Politecnico di Milano, Via G. la Masa 1, 20156 Milan, Italy

\* Correspondence: [steven.chatterton@polimi.it](mailto:steven.chatterton@polimi.it); Tel.: +39-02-2399-8442

**Abstract:** Due to the excellent image recognition characteristics of convolutional neural networks (CNN), they have gained significant attention among researchers for image-processing-based defect diagnosis tasks. The use of deep CNN models for rolling element bearings' (REBs') defect diagnosis may be computationally expensive, and therefore may not be suitable for some applications where hardware and resources limitations exist. However, instead of using CNN models as end-to-end image classifiers, they can also be used to extract the deep features from images and those features can further be used as input to machine learning (ML) models for defect diagnosis tasks. In addition to extracting deep features using CNN models, there are also other methods for feature extraction from vibration characteristic images, such as the extraction of handcrafted features using the histogram of oriented gradients (HOG) and local binary pattern (LBP) descriptors. These features can also be used as input to classical ML models for image classification tasks. In this study, a performance comparison between all these image-processing-based defect diagnosis techniques was carried out in terms of fault detection accuracy and computational expense. Moreover, based upon the detailed comparison, a hybrid-ensemble method involving decision-level fusion is proposed, which is far less computationally expensive compared to CNN models while using them as end-to-end classifiers. The performance of all these models is also compared in the case of minimal training data availability and for diagnosis under slightly different operating conditions to ascertain their generalizability and ability to correctly diagnose despite the minimal availability of training data. The performance of the proposed hybrid-ensemble method remained outstanding for the REBs' defect diagnosis despite the minimal of availability training data as well as the slight variation under operating conditions.

**Citation:** Tayyab, S.M.; Chatterton, S.; Pennacchi, P. Image-Processing-Based Intelligent Defect Diagnosis of Rolling Element Bearings Using Spectrogram Images. *Machines* **2022**, *10*, 908. <https://doi.org/10.3390/machines10100908>

Academic Editors: Xiang Li and Jie Liu

Received: 14 September 2022

Accepted: 5 October 2022

Published: 8 October 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** rolling element bearings; image processing; intelligent defect diagnosis; artificial neural network (ANN); convolutional neural network (CNN); K-nearest neighbor (KNN); support vector machine (SVM); histogram of oriented gradients (HOG); local binary patterns (LBPs)

## 1. Introduction

Rolling element bearings (REBs) are an important part of rotating machines and their damage may result in the unavailability of machines. Therefore, the accurate and timely fault diagnosis of REBs is crucial for the operational availability of machines. Bearing fault diagnosis using vibration data is a well-known method. Vibration analysis is conducted in the time domain, frequency domain and time–frequency domain [1,2]. However, it is an expert-exclusive task. Humans may not be very efficient in the timely management of huge data. Intelligent fault diagnosis techniques based on artificial intelligence (AI) were introduced by the researchers in order to reduce human intervention in this important job and to relieve the experts from this daunting task. During the past two decades, researchers have established many techniques for the defect diagnosis of REBs based on artificial intelligence or classical machine learning models, such as artificial neural networks (ANNs) [3–7], K-nearest neighbor (KNN) [5] and support vector machine (SVM) [6–9].

Moreover, comparisons between the performances of different ML models such as between ANN and KNN [10] and between ANN and SVM [11] have also been carried out for bearing defect diagnosis.

Classical ML-based techniques involve feature extraction and selection, which still require expert knowledge of these domains. In order to further automate the process of fault diagnosis, deep learning (DL)-based techniques such as autoencoder (AE)-based techniques, deep belief network (DBN)-based techniques and convolutional neural network (CNN)-based techniques were introduced. DL-based models do not require manual feature extraction and can automatically extract features from vibration data for end-to-end diagnosis and further reduce human intervention in the process of fault diagnosis.

Convolutional neural networks are a special type of deep neural networks (DNNs) which are famous for their image identification and classification ability [12]. One-dimensional (1D) and two-dimensional (2D) CNN models have been used by the researchers for the defect diagnosis of rolling element bearings. Raw vibration data can be directly used as input to 1D-CNN models [13,14], however, for 2D-CNN models, vibration data must first to be converted into representative images such as spectrograms [15–17], scalograms [17,18], order maps [19] or other types of vibration images [20]. Afterwards, these images which represent the vibration signal in image form are used as input to the 2D-CNN model. These 2D-CNN models are capable of performing end-to-end diagnosis without domain expert involvement by automatically extracting the features from input vibration characteristic images and learning them for correct classification. For this purpose, the CNN model can be developed and trained from scratch or already developed and trained CNN models on other types of datasets can be used with transfer learning. Training a CNN model from scratch may require a large training dataset which is mostly not available in the case of REBs. Therefore, for defect diagnosis, different pretrained image classification networks that have already learned to extract powerful and informative features from training on a subset of the ImageNet database can be used by fine tuning the deeper layers using the available bearings' fault data. The ImageNet database is a large collection of images annotated by humans for providing a comprehensive dataset to develop and test computer vision algorithms [21]. It consists of over fifteen million labeled images belonging to over 22,000 categories. This dataset can be used as a valuable resource for object recognition and image classification applications. From 2010, an annual competition called the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) was started as a benchmark competition in object category classification and detection. A subset of the ImageNet dataset consisting of one thousand categories and more than one million images is used for this competition [22]. The initial layers of the pretrained network are frozen for generic feature extraction and a final few layers are retrained for the learning of specific features [23]. Using a pretrained network with transfer learning is typically much faster and saves in terms of computational cost as compared to training a network from scratch [24,25]; however, it is still computationally expensive and requires a lot of training time if GPU is not available, and therefore, it may not be suitable for online monitoring and diagnosis applications or in the presence of hardware and resource limitations.

Instead of using CNNs as end-to-end classifiers, they can also be used as deep feature extractors, and these features can be used to train conventional ML models such as SVM, KNN and ANN. Xie et al. compared different ML models for the fault diagnosis of bearings using features extracted by a pretrained CNN model on a bearing dataset [26]. Moreover, image features can be extracted using descriptors such as the histogram of oriented gradients (HOG) and local binary patterns (LBPs), whilst those features can be used for fault diagnosis using conventional ML models. Khan and Kim performed the diagnosis of bearing defects using the KNN classifier and LBP features extracted from grayscale vibration images [27]. Kaplan et al. converted the bearings' vibration signal into gray images and performed fault diagnosis using different ML models as classifiers and LBP features as input to ML models [28]. Chen et al. transformed vibration signals into bi-spectrum contour maps and extracted HOG features from the transformed bi-spectrum maps

to use them as input to the random forest (RF) model for the diagnosis of bearings and the impeller defects of self-priming pump [29]. These image-processing-based defect diagnosis techniques are less computationally expensive compared to CNN, even while using it as an end-to-end classifier with transfer learning.

Moreover, the performances of different models may be affected differently; (1) under small sample conditions, i.e., if minimal training data are available; and (2) if the operating conditions under which the model is tested are slightly different to the operating conditions under which the model was trained [19,30]. Therefore, there is a need to evaluate the trade-off between the computational expense and desired accuracy to choose the best image-processing-based technique for the intelligent defect diagnosis of REBs: (1) under normal conditions, when sufficient training samples are available; (2) in the case of minimal data availability; and (3) if the operating conditions change slightly.

In this study, the suitability of eleven different famous pretrained CNN models on a subset of the ImageNet database used in ILSVRC was evaluated for its application in defect diagnosis of rolling element bearings using the transfer learning approach. A comparison of the performances of all these CNN models being used as end-to-end classifiers and with other image-processing-based techniques was performed: (1) CNN + ML models (KNN, ANN, SVM); and (2) handcrafted features (HOG, LBP) + ML models (KNN, ANN, SVM) were carried out in terms of fault detection accuracy and the time taken for training and testing during the defect diagnosis process. Based upon the best selected features type and ML models, a hybrid-ensemble method is proposed for the defect diagnosis of REBs. The performance of all these models is also compared in the case of minimal training data availability and for diagnosis under slightly different operating conditions, as compared to those under which models were trained to ascertain their generalizability and ability to diagnose correctly under the minimal availability of training data.

## 2. Theoretical Background

### 2.1. Spectrogram

A spectrogram using the short-time Fourier transform (STFT) is a visual representation of a signal in the time–frequency domain which is used to investigate the variation in the frequency of the signal over time, as shown in Figure 1. The STFT of a signal  $x(n)$  is calculated by sliding a window of length  $L$  over the signal and calculating the discrete Fourier transform (DFT) of each window. The window hops over the signal at intervals of  $P$  samples. The DFT of each window segment is added into a matrix which contains the phase and magnitude for each point in frequency and time. The number of rows of the DFT matrix is equal to the DFT points and the number of columns of the DFT matrix can be calculated as follows:

$$j = \left\lfloor \frac{N_x - O}{L - O} \right\rfloor \quad (1)$$

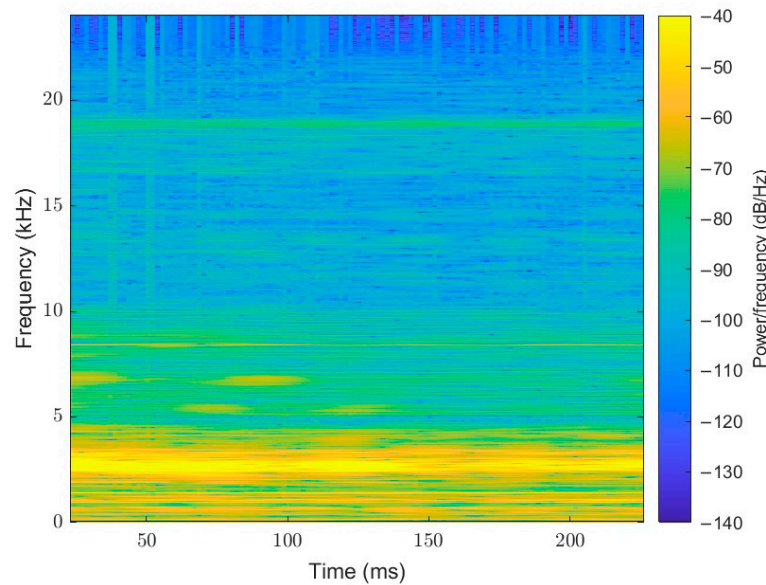
where  $N_x$  is the length of the signal and  $O$  is the overlap length. The STFT matrix can be expressed as follows:

$$X(f) = [X_1(f) \quad X_2(f) \quad X_3(f) \quad \dots \quad X_k(f)] \quad (2)$$

where the  $m_{th}$  element of the matrix can be given as follows:

$$X_m(f) = \sum_{n=-\infty}^{\infty} x(n) g(n - mP) e^{-j2\pi fn} \quad (3)$$

where  $X_m(f)$  is the DFT of the windowed data about  $mP$ ;  $g(n)$  is the window function of length  $L$ ; and  $P$  is the hop size between two successive DFTs. The hop size is the difference between window size  $L$  and overlap size  $O$ .



**Figure 1.** Example of a spectrogram.

## 2.2. Artificial Neural Network

Inspired by biological systems, artificial neural networks are intelligent systems which are composed of simple elements operating in a parallel way [31]. ANN has the ability to replicate the functioning of the human brain, i.e., generalizing and refereeing through situations by analyzing the processing information [32]. A basic neural network is known as a perceptron which can be used as a classifier. In order to form a network which can solve nonlinear problems, an intermediate layer of neurons is added between the input and output of a single-layered perceptron. Any intermediate layer of the neuron is called a hidden layer. This network is called a multilayer perceptron (MLP) which consists of an input layer, one or more hidden layers and an output layer. The number of input and output variables decide the number of nodes in the input and output layers of the MLP, respectively. Hidden layers may consist of many computational nodes which are called neurons. The number of hidden layers in the MLP and the number of neurons in each hidden layer may affect the computational power and generalization ability of the neural network. A feedforward neural network is the basic type of such networks. In a feedforward neural network, information flows in one direction, i.e., from input nodes to a hidden layer and then to the output layer [10]. Because of the ability of MLP/ANN to measure nonlinear relationships in complex processes, it can be employed in REBs' defect diagnosis.

## 2.3. K-Nearest Neighbors

K-nearest neighbors is a simple but robust non-parametric method for regression and classification. Training data are used as the input and output of the model and are dependent on whether the model is used for regression or classification. Classification is performed on the basis of similarity measure, i.e., the minimum distance between the patterns in the feature space, which can be measured by Mahalanobis or Euclidean distance, etc. [33,34]. The distance metric, number of neighbors value (K) and some decision rules are involved while making the decision. In this study, the KNN model is used as a classifier for the REBs' defect diagnosis where output is a class membership.

## 2.4. Support Vector Machine

The support vector machine is a supervised learning model which can also be used for classification and regression. The SVM classifier separates the training sample using

the hyperplane which is found to maximize the margin. SVM is a binary classifier: therefore, if the classification task involves more than two classes, then multiclass SVM is used, and the multiclass problem is converted into a series of binary classification problems. There are different techniques to handle multiclass classification. In this study, while utilizing SVM as a classifier for REBs' defect diagnosis, we used the one vs. one technique. In this technique, for each binary learner, one class is positive and another class is negative, whereas all other classes are ignored. As such, all combinations of class pairs are used.

### 2.5. Convolutional Neural Network

CNNs are well known for their image recognition and classification capabilities. They comprise of an input layer, hidden layers and an output layer. Hidden layers in convolutional neural networks generally consist of convolutional layers, rectified linear unit (ReLU) layers, subsampling or pooling layers and fully connected layers [35]. By moving the kernels horizontally and vertically, the 2D convolutional layer convolves the input through the dot product of kernels and input and then adds the bias term. The input of the 2D convolutional layer is the output of the preceding layer. Local features of the input region are extracted by kernels. From the results of the convolution operation, output is obtained in terms of features by using an activation function, such as ReLU. It performs a threshold operation in which all the values that are less than zero are set to zero. Equation (4) describes the mathematical model of the convolutional layer.

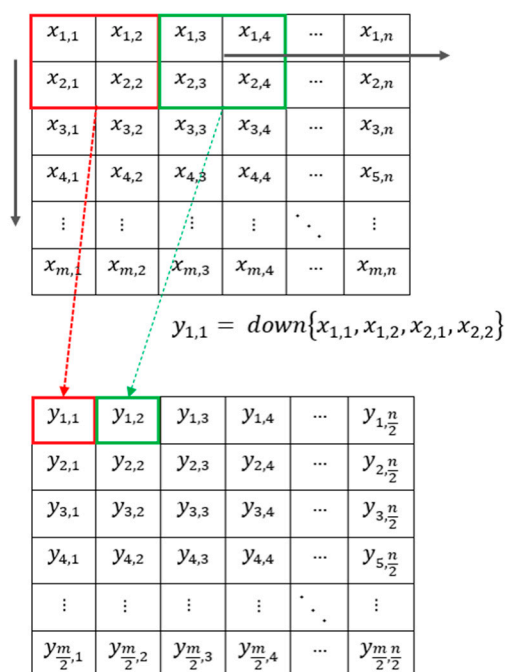
$$Y_j^m = f\left(\sum_{i \in C_j} Y_i^{m-1} * k_{ij}^m + b_j^m\right) \quad (4)$$

where an input map selection is symbolized by  $C_j$ ,  $m$  is the  $m_{th}$  layer in the network,  $Y_i^{m-1}$  is the input of the convolutional channel,  $f$  is a nonlinear activation function such as ReLU,  $b$  is the bias matrix and  $k$  is the kernel matrix.

In order to reduce the size of the input features and network parameters, a subsampling or pooling layer was applied after the convolutional layer. The pooling operation can be average pooling or max pooling based upon the pooling function. Equation (5) describes the mathematical model of the pooling layer.

$$Y_j^m = f\left(\beta_j^m * \text{down}(Y_j^{m-1}) + b_j^m\right) \quad (5)$$

where  $\beta_j^m$  is multiplicative bias and additive bias is represented by  $b_j^m$ . The pooling function is represented by "down (.)" which is the down-sampling function that is used to reduce the dimension of feature maps attained after the convolution operation. It can be  $\max(\cdot)$  in the case of max pooling or can be  $\text{mean}(\cdot)$  in the case of an average or mean pooling operation. It returns a maximum or average value over the selected segment size, as shown in Figure 2.



**Figure 2.** Down-sampling in pooling operation.

Output is fed to a fully connected layer after the multiple stacking of convolutional layers and pooling layers. A fully connected layer is a feedforward neural network (multilayer layer perceptron) that uses softmax as an activation function in the output. Softmax is an activation function which converts the raw output of the network into a vector containing probabilities for each class, i.e., it normalizes the output of the network into probability distribution over predicted classes. Before the application of softmax, some components of the network output vector may be negative or greater than one and their sum may not be equal to one. However, after the application of the softmax activation function, the output vector will contain all the numbers in a range between zero and one, and their sum will be equal to one; therefore, they can be inferred as probabilities related to corresponding classes. The softmax activation function is described in Equation (6)

$$\delta(\vec{x})_i = \frac{e^{x_i}}{\sum_{j=1}^N e^{x_j}} \tag{6}$$

where  $\vec{x}$  is the input to the softmax function  $\delta$  and the input vector  $\vec{x}$  contains  $N$  number of elements for  $N$  classes. Here, the softmax applies the standard exponential function to each element  $x_i$  of input vector  $\vec{x}$  and divides these exponentials by the sum of all exponentials in order to ensure that the sum of all the elements of the output vector  $\delta(\vec{x})$  is equal to one. All the neurons in a fully connected layer are connected to all the neurons in the previous layer because their objective was to gather all the features learned from previous layers to identify patterns.

### Transfer Learning

Training a CNN model from scratch may require a lot of time and a lot of data. An alternate solution may be that of transfer learning, in which the weights of a pretrained network on standard data such as those of the ImageNet dataset are used to train the network on a different dataset. In transfer learning, the parameters of the neural network are determined by transferring the knowledge from a network which was already learned from a huge set of training data. The initial layers of CNN were not updated because they work as a feature extractor. However, the final layers are fine-tuned using the application

training data. In this study, different pretrained CNN models on the ImageNet dataset are used for bearing defect diagnosis using transfer learning.

### 2.6. Image Features Extraction

Image features are distinctive signatures of the image which can be used to identify an image or to differentiate between different images. The different types of image features are given as follows:

#### 2.6.1. Deep Features

The automatic defect diagnosis of REBs using vibration characteristic images without the involvement of experts is also possible by only utilizing the CNN model for feature extraction and those features can be used to train a classical ML model such as SVM, KNN, or ANN for a defect diagnosis task. The activations of a fully connected layer or pooling layer are mostly used as the deep features of an image for an image classification task in combination with different ML models.

#### 2.6.2. Handcrafted Features

Different handcrafted algorithms such as a histogram of oriented gradients and local binary patterns can be used to extract image features which can be further used as input to an ML model for recognition or classification tasks.

#### Histogram of Oriented Gradient

A histogram of oriented gradients is an important method for image feature extraction. It has been widely used for objects and human detection. In this method, the occurrences of gradient orientations in the local regions of an image are counted. Images are described by the sets of local histograms. The local appearance of an object in an image can be described by intensity gradients' distribution or the direction of edges. The image is divided into small cells and a histogram of gradient directions is composed for the pixels within each cell. These histograms are then concatenated. This concatenation produces the HOG features vector [36]. These HOG features can be used as input to the ML model for image classification tasks. In this study, we used the  $(8 \times 8)$  cell size, which is grouped into blocks of size  $(2 \times 2)$ .

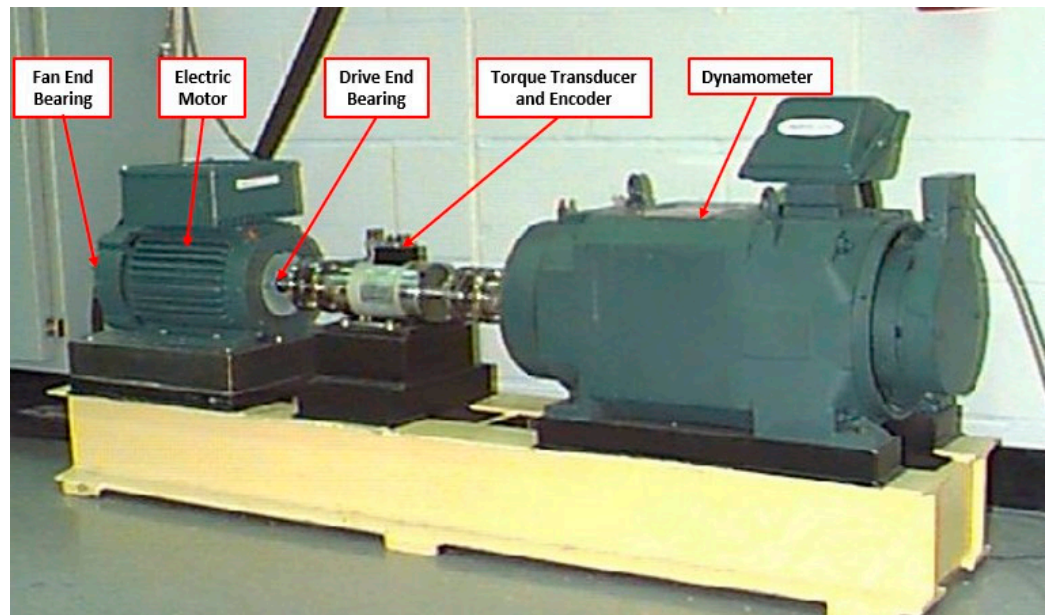
#### Local Binary Patterns

LBP is an efficient descriptor for texture classification. LBP features can be used as input to an ML model to categorize objects. The image is divided into cells. Each pixel in the cell is compared to each of its neighbors following a circle [37]. Normally, the neighbors' value varies from 4 to 24. In this study, the value of neighbors is kept at 8. If the neighbor's pixel's value is greater than the center's pixel value, then "1" is written—otherwise "0" is written. As such, an eight-digit binary code is obtained which is converted into a decimal number. A histogram which is a multi-dimensional feature vector that is computed over the cell. The feature vector for the entire window is obtained by normalizing and concatenating the histograms of all cells.

## 3. Experimental Setup and Vibration Data

In this study, an open source benchmark-bearing dataset provided by the CWRU bearing center was utilized [38]. The experimental setup is shown in Figure 3. It consists of an electric motor which drives a shaft on which a torque transducer and encoder are mounted. The torque is applied using a dynamometer. In order to collect fault data, three types of defects—namely (1) inner race; (2) outer race; and (3) ball defects of different sizes ranging from 0.007 to 0.028 inches—were introduced in fan-end and drive-end bearings (SKF deep groove ball bearings: 6203-2RSJEM and 6205-2RSJEM, respectively) using electrical discharge machining (EDM). The experimentation was carried out at a constant

speed at different loads between 0 and 3 hp. Due to the change in load, there was a slight change in average speed, ranging between 1796 and 1729 rpm and generating four slightly different operating conditions, as given in Table 1.



**Figure 3.** Experimental Setup [38].

**Table 1.** Different operating conditions.

Operating Condition (OC) Number	Load (HP)	Speed (rpm— Approximate)
1	0	1797
2	1	1772
3	2	1749
4	3	1720

Vibration data were collected in the vertical direction using the accelerometer from the drive-end bearing housing (DE), fan-end bearing housing (FE) and base plate supporting the motor (BA). Sampling frequencies of 12 kHz and 48 kHz were used for the data collection. The outer race defects can be further divided into three groups depending on the fault position with respect to the load zone: (1) centered (fault at 6.00 o'clock position); (2) orthogonal (fault at 3.00 o'clock position); and (3) opposite (fault at 12.00 o'clock position). In this study, vibration data collected at 48 kHz sampling frequency from the drive-end bearing were utilized. Three levels of fault severity/size—0.007, 0.014 and 0.021 inches—are considered for all three types of defects: inner race, outer race and ball defects. This creates 10 classes including the normal base line data, as given in Table 2. For the outer race, centered defects are used.

**Table 2.** Selected bearing fault classes.

S. No	Defect Class Number	Defect Type	Defect Size (Inches)
1	0	Normal/healthy	-
2	1	Ball defect	0.007
3	2	Ball defect	0.014
4	3	Ball defect	0.021
5	4	Inner race defect	0.007
6	5	Inner race defect	0.014

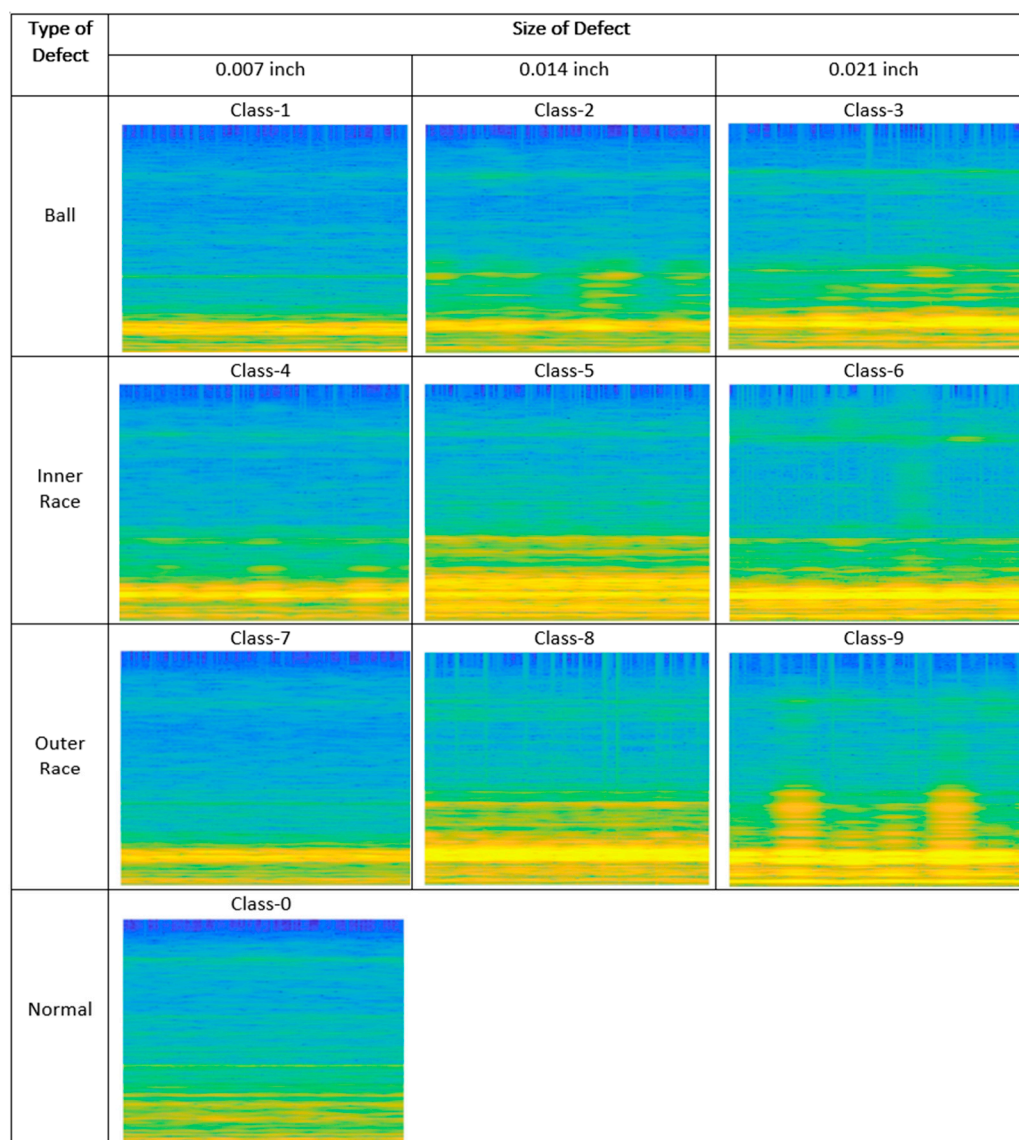
---

7	6	Inner race defect	0.021
8	7	Outer race defect	0.007
9	8	Outer race defect	0.014
10	9	Outer race defect	0.021

---

#### 4. Applied Methodology

In order to carry out the defect diagnosis of REBs using image-processing-based techniques, the vibration data are first converted into representative time–frequency images and then image classification-based techniques are applied. The vibration data for each considered class in this study were divided into segments using a window size of 0.25 s and an overlapping of 80%. Each signal segment was converted into a spectrogram using short-time Fourier transform. The spectrograms for all ten classes considered in this study, at 2 hp, i.e., OC-3, are shown in Figure 4. These spectrogram images were used for the defect diagnosis of REBs. Three image-processing-based defect diagnosis techniques involving different models and combinations were compared. Initially, the performance comparison in terms of defect diagnosis accuracy and the time taken for training and testing (as a metric for computational expense) was carried out using a sufficient number of training images (170 images for each class). Afterwards, the performance in terms of defect diagnosis accuracy was compared under the following conditions: (1) when minimal training data are available (using only 20 and 14 spectrogram images); and (2) when the operating conditions are slightly changed. The hardware platform used in this study was a laptop with an Intel(R) Core (TM) i7-4810MQ CPU @ 2.80GHz and 8 GB of RAM. A description of all three considered image-processing-based defect diagnosis methods and a proposed hybrid-ensemble method is given as follows:



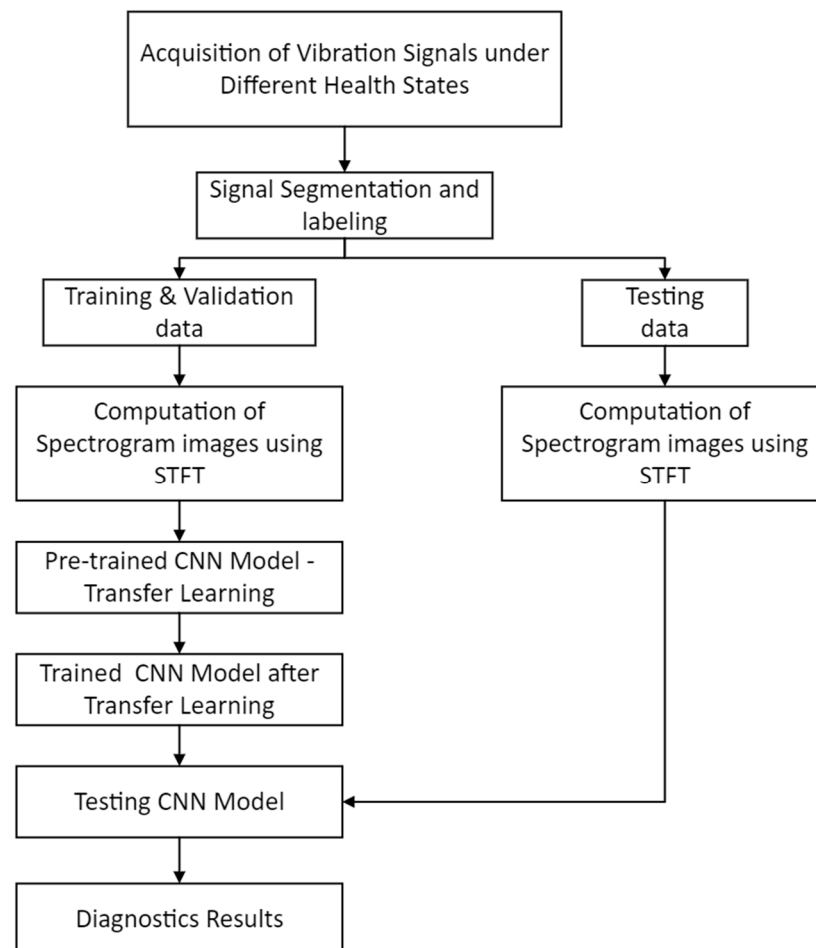
**Figure 4.** Spectrogram images for different defect classes listed in Table 2.

#### 4.1. Method-1 (CNN as End-to-End Classifier – Transfer Learning)

Spectrogram images are used as input to pretrained CNN models for end-to-end defect diagnosis. The last few layers of different pretrained CNN models, trained on the ImageNet database, were fine-tuned using the training data of spectrogram images of REBs' vibration data. AlexNet [39] is considered the pioneer CNN model which was implemented on GPU and provided a breakthrough in deep learning by connecting the growing computation power with deep learning. Later on, encouraged by the deep learning revolution, researchers introduced many powerful architectures such as GoogLeNet and SqueezeNet. For this study, eleven different CNN models—mostly used by the researchers of different fields for image classification applications—were considered. The pretrained CNN models considered in this study along with the required input image size are given in Table 3. For detailed information about each considered CNN model, the references mentioned against them can be consulted. The trained CNN models after transfer learning were tested using the testing data of spectrogram images. Method 1 is shown in Figure 5. The performances of eleven different CNN models were compared in terms of fault diagnosis accuracy and the time taken for the training and testing processes. Training was carried out for five epochs for all models.

**Table 3.** CNN models.

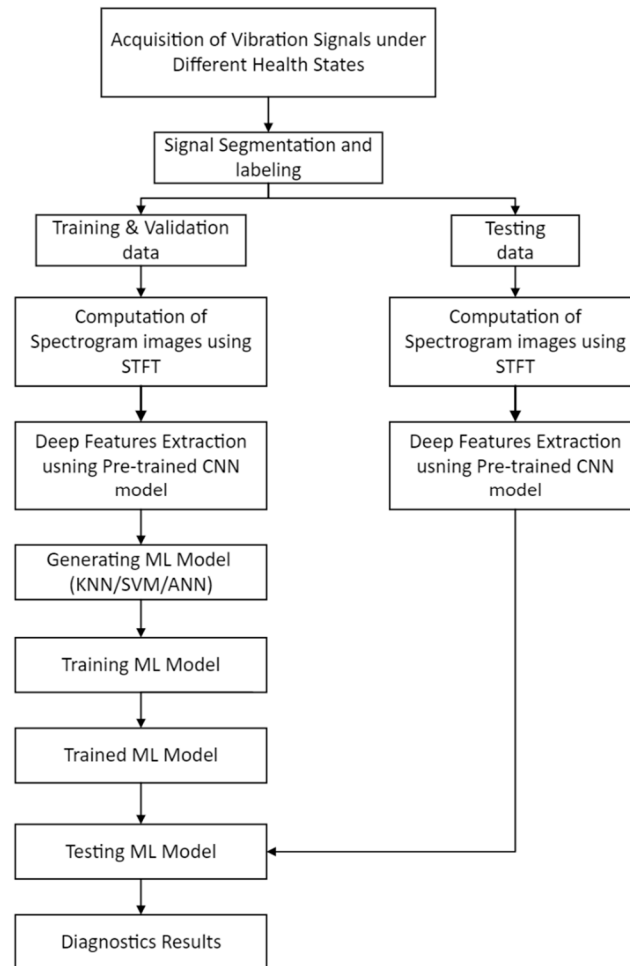
S. No	Network/Model	Input Image Size
1	AlexNet [39]	227-by-227
2	SqueezeNet [40]	227-by-227
3	GoogLeNet [41]	224-by-224
4	ResNet-18 [42]	224-by-224
5	MobileNet-v2 [43]	224-by-224
6	Inception-v3 [44]	299-by-299
7	DenseNet-201 [45]	224-by-224
8	ResNet-50 [42]	224-by-224
9	ResNet-101 [42]	224-by-224
10	VGG -16 [46]	224-by-224
11	VGG -19 [46]	224-by-224

**Figure 5.** Method-I.

#### 4.2. Method-II (Deep Features + Classical ML Models)

Instead of using CNN as an end-to-end classifier, it can be used for the extracting deep features from images. Activations from convolutional or pooling layers can be used as input to a classical ML model for a defect diagnosis task. The CNN model was selected based upon defect diagnosis accuracy and the least time taken for training and testing during the process of using CNN models as end-to-end classifiers. The selected CNN model was used for deep feature extraction from spectrogram images. The extracted features were used to train three ML models: KNN, SVM and ANN. The performance of all

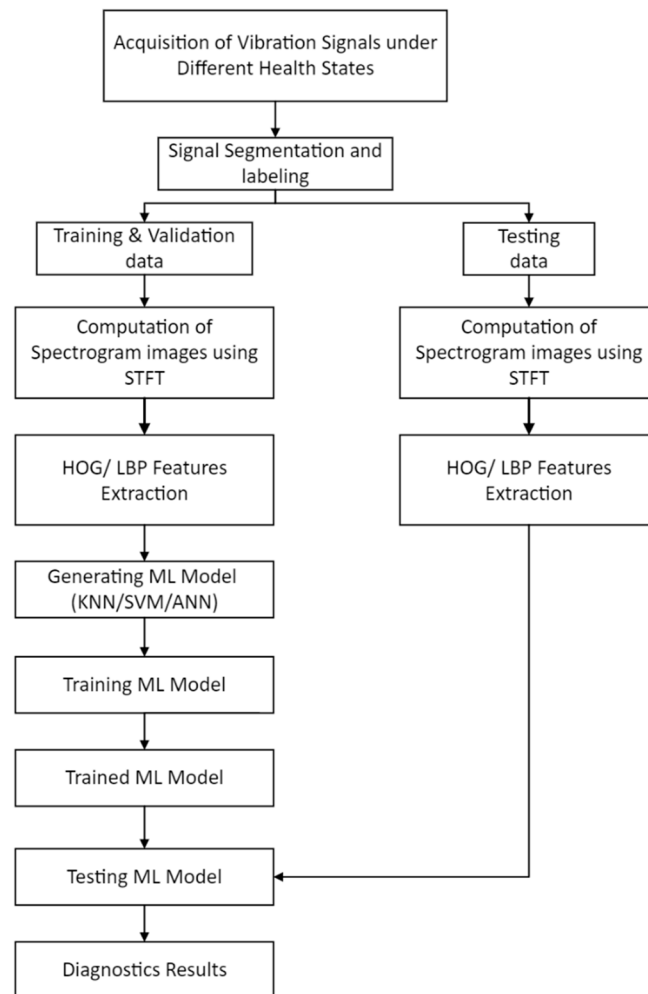
three ML models for the defect diagnosis of REBs, using deep image features as input, was compared with all CNN models used as end-to-end image classifiers. Method-II is shown in Figure 6.



**Figure 6.** Method-II.

#### 4.3. Method-III (Handcrafted Features + ML Model)

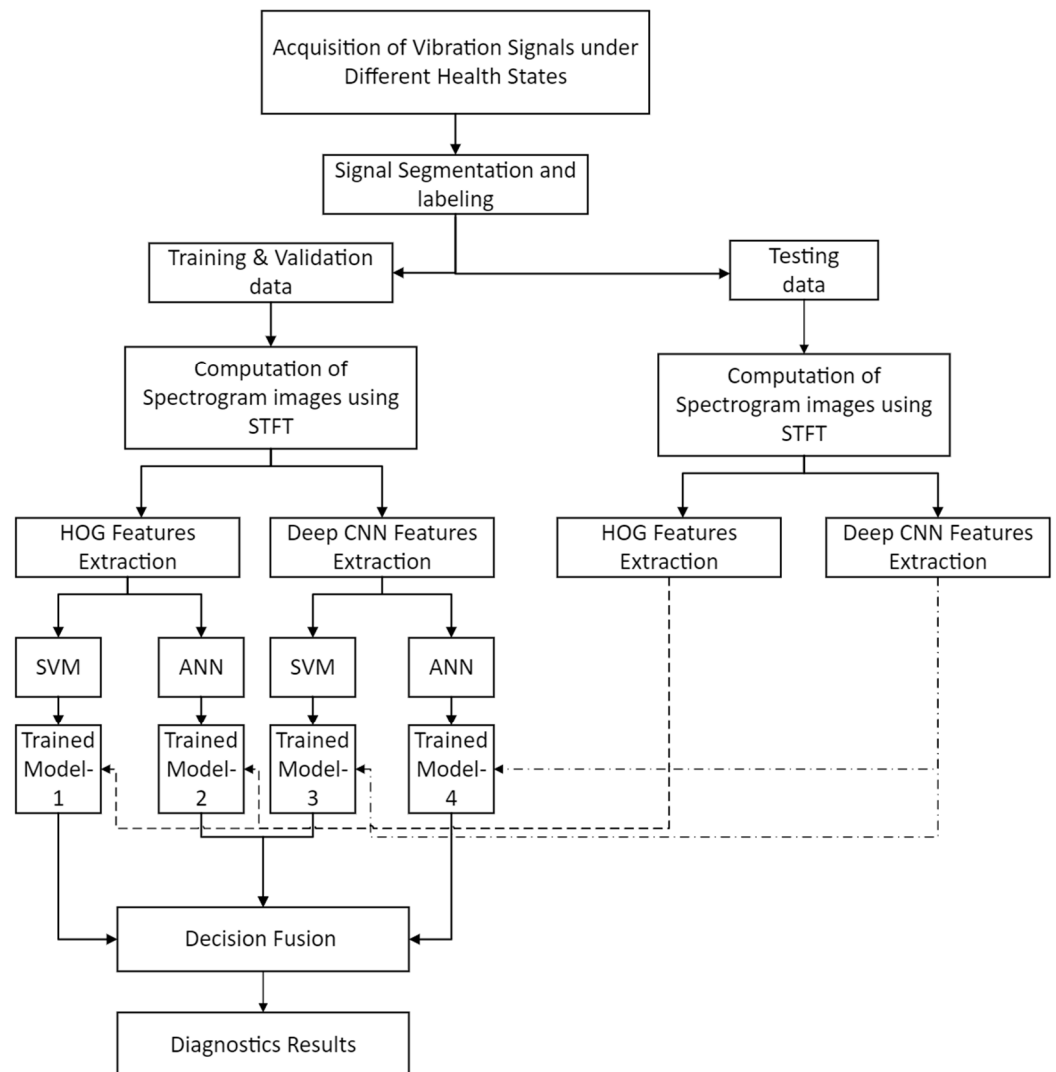
Handcrafted features were extracted from the spectrogram images using HOG and LBP descriptors. These features were used as input to three ML models—KNN, SVM and ANN—for the defect diagnosis of REBs. The performance of all these combinations was compared with the performance of all combinations of Method-I and Method-II. Method-III is illustrated in Figure 7.



**Figure 7.** Method-III.

#### 4.4. Proposed Hybrid-Ensemble Method

The performance comparison of all models and image-processing-based defect diagnosis techniques considered in this study was carried out in terms of fault detection accuracy and the time taken for the training and testing processes in function of the sufficient availability of training data. Based upon best the combinations of image feature types and ML models, a hybrid-ensemble method involving the majority voting-based decision fusion technique is proposed for the defect diagnosis of REBs. The proposed method is illustrated in Figure 8.



**Figure 8.** Proposed hybrid-ensemble method.

After the computation of spectrogram images, HOG and deep-CNN features (selected based upon the performance comparison given in Section 5) are extracted in parallel to train the selected ML models (SVM and ANN) separately. The output of all models, treated equally, is fused to give the final decision about the state of the bearing's health using majority voting-based decision level fusion. In this method, the final decision about the class label is made based upon the most frequent output from the combined set of individual classifiers' output. As such, the voting benefit from different ML models trained on different types of features is achieved to predict the correct health state of rolling element bearings.

## 5. Results and Discussion

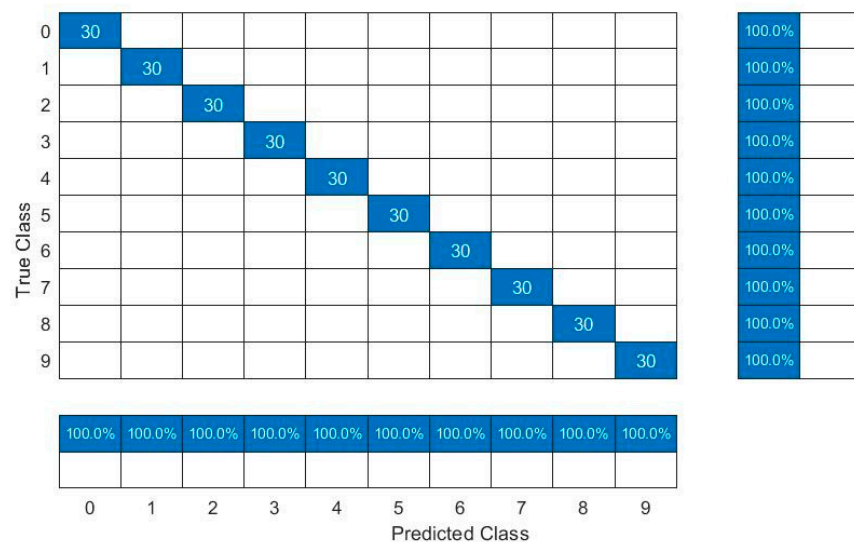
The performance of all image-processing-based intelligent defect diagnosis techniques considered in this study was initially compared in terms of the fault diagnosis accuracy and time taken for the training and testing processes under the conditions when sufficient training samples were available (170 spectrogram images for each defect class). The vibration data of operating condition-3, as given in Table 1, were used for this comparison. The results are given in Table 4.

**Table 4.** Performance comparison results using different sizes of training datasets (OC-3).

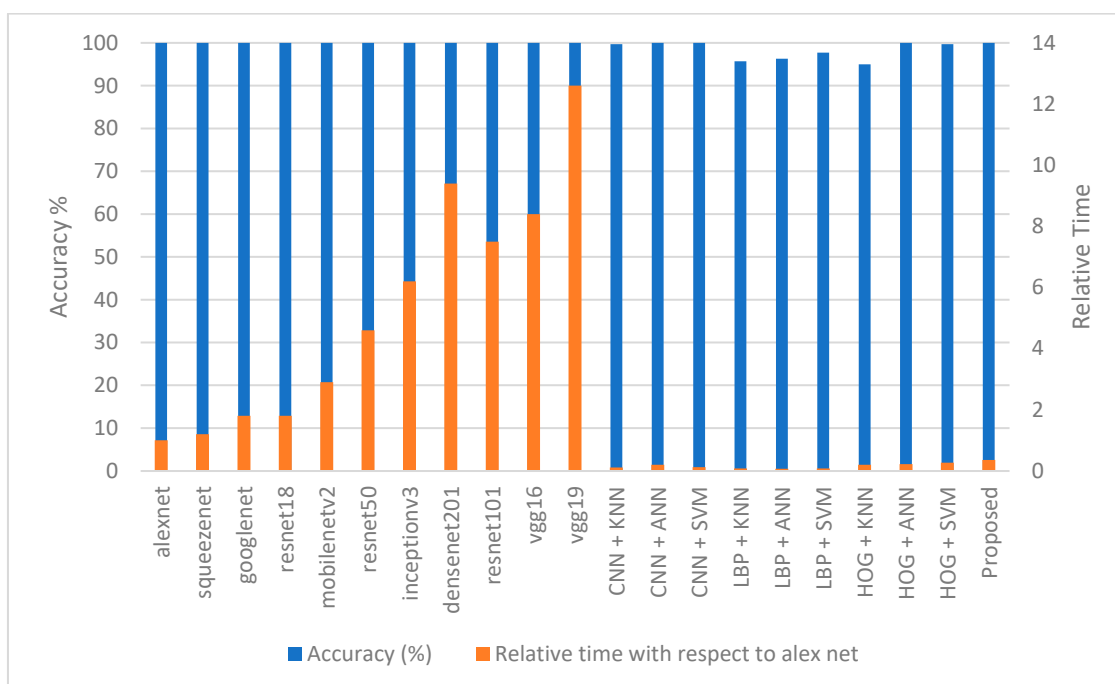
Size of Training Dataset (for Each Class)		170 Images			20 Images	14 Images	
Method		Accuracy (%)	Time (s)	Relative Time with Respect to AlexNet	Accuracy (%)	Accuracy (%)	
1	Transfer learning from pre-trained CNN models on ImageNet dataset	AlexNet	100	1026.1	1	87.5	78.1
		SqueezeNet	100	1206.4	1.2	87	74.5
		GoogLeNet	100	1817.9	1.8	90.3	84
		ResNet-18	100	1849.8	1.8	100	99.9
		MobileNet-v2	100	3053.1	2.9	99.6	99.6
		ResNet-50	100	4772.4	4.6	99.9	99.5
		Inception-v3	100	6347.5	6.2	99.4	99.2
		DenseNet-201	100	9655.8	9.4	100	99.9
		ResNet-101	100	7691.9	7.5	99.9	99.9
		VGG -16	100	8632.2	8.4	40.7	23.2
	VGG -19	100	12875.5	12.6	77.6	50.4	
2	Deep CNN features + ML model	CNN + KNN	99.7	117.5	0.11	99.5	98.1
		CNN + ANN	100	205.4	0.2	99.7	99.7
		CNN + SVM	100	118.03	0.12	100	99.7
3	Handcrafted features + ML model	LBP + KNN	95.7	53.2	0.08	92	91.7
		LBP + ANN	96.3	60	0.07	93.1	89.7
		LBP + SVM	97.7	58.99	0.08	97.5	94
		HOG + KNN	95	195.5	0.2	86.1	83
		HOG + ANN	100	221.5	0.22	95	94.8
		HOG + SVM	99.7	231.6	0.27	98.5	94.1
4	Proposed hybrid-ensemble meth.	HOG and deep CNN features + ANN and SVM	100	367	0.36	99.8	99.5

For transfer learning (Method-I) using pretrained CNN models, all the CNN models considered in this study performed fault diagnosis with 100% accuracy on unseen testing data. However, some models were much more computationally expensive compared to other models and consumed much more time for the training and testing processes keeping the same number of epochs. AlexNet took least time, and therefore, the relative time taken by all other models is shown in terms of the time taken by AlexNet. The VGG19 model took the most time, which was 12.6 times that of AlexNet. For very large datasets such as the ImageNet dataset which contains 1000 classes of images, the performance may improve by improving the model such as by increasing the depth at the expense of the computation cost. However, for the application considered in this study, it is observed that very deep networks are not required if a sufficient number of training images are available. A relatively simple network such as AlexNet can perform sufficiently well for defect classification with lesser computational expense. Therefore, AlexNet was selected for the extraction of deep features from spectrogram images for Method-II. The extracted features were used to train KNN, ANN and SVM models as classifiers. ANN and SVM achieved fault diagnosis with 100% accuracy and KNN achieved fault diagnosis with 99.7% accuracy using deep features extracted using the AlexNet. The time for all three models in Method-II was significantly less compared to that of the CNN models of Method-I due to it having less computational requirements, thus making it more suitable for online monitoring and diagnosis. In Method-III, two descriptors, LBP and HOG, were used for feature extraction from spectrogram images. All three ML models considered in

this study were trained using these features separately. HOG + ANN and HOG + SVM achieved fault diagnosis with 100% and 99.7% accuracy, respectively. However, other all combinations considered in Method-III achieved fault diagnosis with less than 97.8% accuracy. The time taken by all HOG combinations is comparable to the time taken by Method-II combinations; however, the time taken by LBP combinations is much less compared to HOG combinations. However, the performance of LBP features was significantly inferior to the performance of HOG and deep features in terms of defect diagnosis accuracy. Moreover, the overall performance of KNN was much more inferior to the performance of ANN and SVM in terms of defect diagnosis accuracy. Therefore, two ML models—SVM and ANN—and two types of features—deep features and HOG features—were selected for the proposed hybrid-ensemble technique. The proposed method achieved fault diagnosis with 100% accuracy, as shown in Figure 9, and only took 36% of the time taken by AlexNet while being used as end-to-end classifier in Method-I. A comparison of the performances of all the considered image-processing-based techniques is shown in Figure 10.



**Figure 9.** Confusion chart of the proposed method for unseen testing data (largest training dataset).



**Figure 10.** Performance comparison under the conditions of the sufficient availability of training data (largest training dataset).

Afterwards, in order to ascertain the capability of the considered image-processing-based defect diagnosis techniques to correctly diagnose the faults when minimal training data are available, the models were trained using only: (1) 20 images for each class; and (2) 14 images for each class. The effect on the fault detection accuracy of all considered models under the conditions of minimal training data is also given in Table 4. For Method-I, by reducing the number of training images for each class from 170 to 20, fault detection accuracy was significantly reduced for AlexNet, SqueezeNet and GoogLeNet. The same trend was observed by further reducing the size of training dataset to 14 images for each class. Out of these three CNN models, minimum fault detection accuracy was exhibited by SqueezeNet, which was 74.5% when 14 spectrogram images were used as training data. However, ResNet18, MobileNetv2, ResNet50, Inceptionv3, DenseNet201 and ResNet101 performed very well under the conditions of the minimal training data availability. Out of these six CNN models, the minimum fault detection accuracy of 99.2% was exhibited by Inceptionv3. However, the performances of the vgg16 and vgg19 models was dropped drastically by reducing the size of the training dataset. The defect diagnosis accuracies for vgg16 and vgg19 were only 23.2% and 50.4%, respectively, when 14 images for each class were used as training data. Out of the six CNN models whose performances were not significantly affected by reducing the size of the training dataset, ResNet18 was found to be the best model and the least computationally expensive. The defect diagnosis accuracy for ResNet18 was 100% and 99.9% for 20 and 14 training images, respectively. A confusion chart for ResNet18 while using the smallest training dataset is shown in Figure 11. For Method-II, the defect diagnosis accuracy was not significantly affected for ANN and SVM models by reducing the training data. The minimum defect diagnosis accuracy for both models remained 99.7%; however, it dropped to 98.1% for KNN. For Method-III, the defect diagnosis accuracy was significantly reduced for all combinations. However, the performance of the proposed method was not significantly affected by reducing the size of the training dataset. The defect diagnosis accuracy for the proposed method remained 99.8% and 99.5% while using 20 and 14 spectrogram images as the training datasets, respectively. A confusion chart for the proposed hybrid-ensemble method with the smallest training dataset is shown in Figure 12. The impact of reducing the size of the training

dataset on all the considered image-processing-based defect diagnosis techniques is shown in Figure 13.

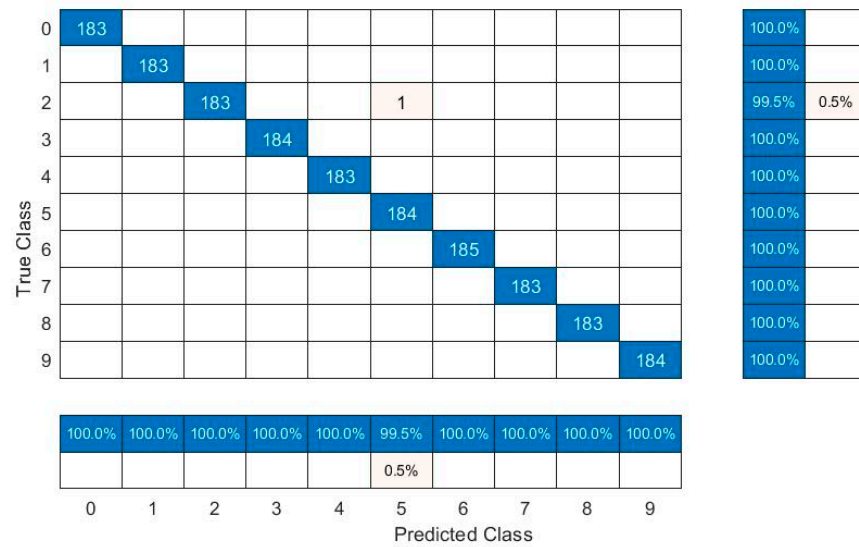


Figure 11. Confusion chart for ResNet18 (smallest training dataset).

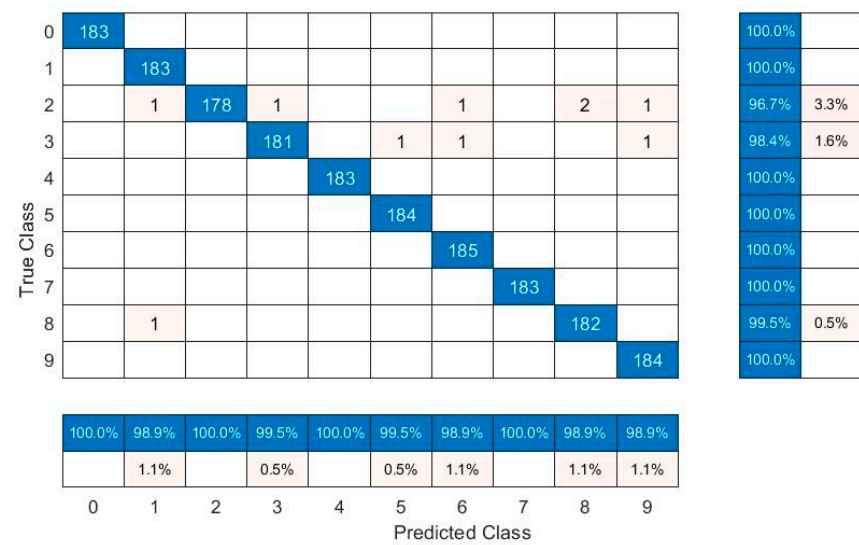


Figure 12. Confusion chart for proposed method (smallest training dataset).

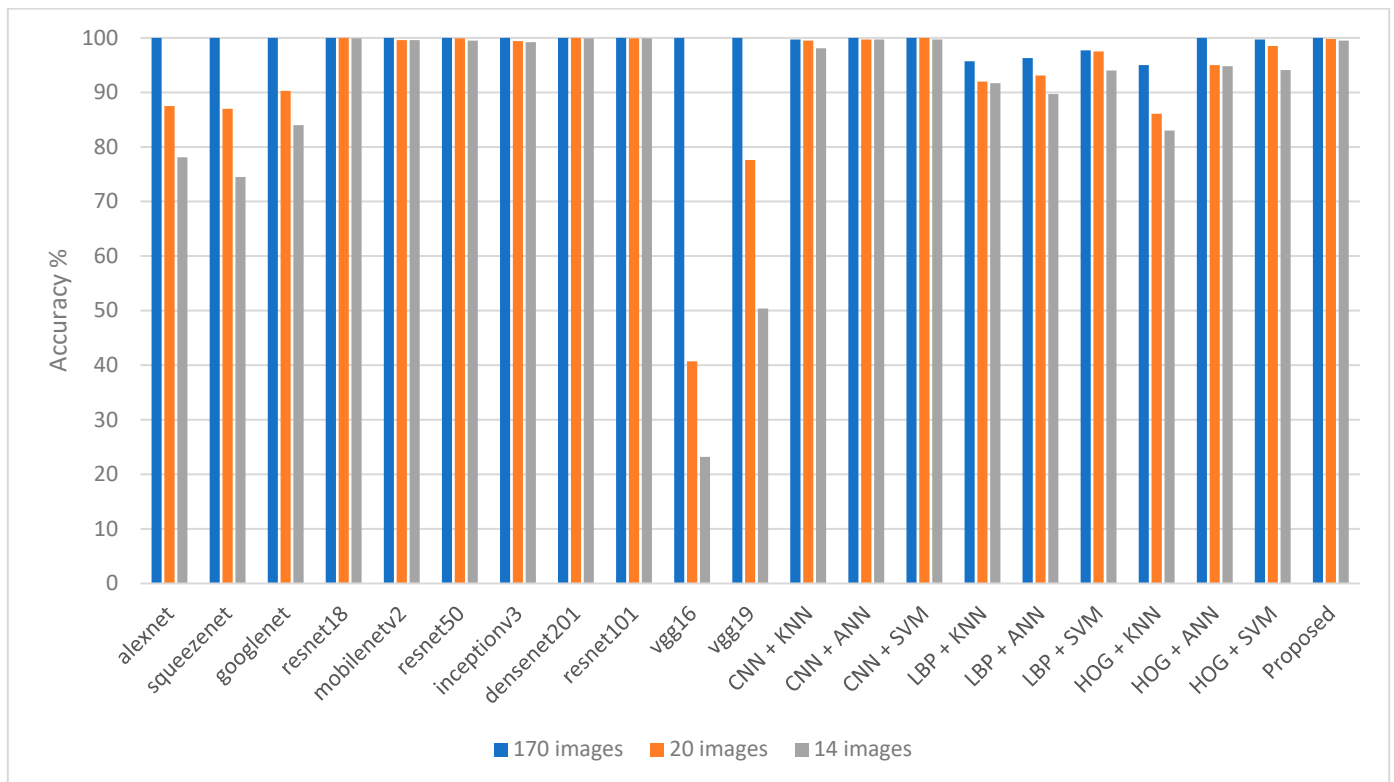


Figure 13. Impact of reducing the size of the training dataset.

To ascertain the defect diagnosis capability of all considered image-processing-based defect diagnosis techniques under slightly different operating conditions, models trained under operating condition-3 were tested under all other considered operating conditions in this study, as shown in Table 1. The results are illustrated in Table 5. For Method-I, the maximum average defect diagnosis accuracy under slightly different operating conditions was 94.3%, as exhibited by ResNet101, and the minimum average accuracy was 87.3%, which was depicted by vgg16. For Method-II, the maximum and minimum average accuracies for defect diagnosis under slightly different operating conditions were 91.1% and 87.9%, as depicted by SVM and KNN, respectively. For Method-III, the average defect diagnosis accuracy was drastically reduced, especially for LBP features’ combinations. The maximum and minimum average defect diagnosis accuracies under slightly different operating conditions were 92.8% and 35.6%, as exhibited by HOG + ANN and LBP + KNN, respectively. The proposed hybrid-ensemble method performed the defect diagnosis task under slightly different operating conditions with 95.7% accuracy, and outperformed all the combinations and models deemed under all three methods compared in this study. A confusion chart for the proposed method tested under OC-4 is shown in Figure 14 and the impact of changing operating conditions on the defect diagnosis accuracy of all considered image-processing-based defect diagnosis techniques is shown in Figure 15.

Table 5. Performance comparison results for different operating conditions.

Testing Operating Condition		OC-1	OC-2	OC-4	Average Accuracy	
Method		Accuracy (%)	Accuracy (%)	Accuracy (%)	(%)	
1	Transfer learning from pretrained CNN models on ImageNet dataset	AlexNet	96.5	87.5	95	93
	SqueezeNet	92.8	89.7	96.8	93.1	
	GoogLeNet	91.7	89.7	98.6	93.3	
	ResNet-18	98.5	89.7	90.6	92.9	
	MobileNet-v2	96.6	88.9	94.1	93.2	
	ResNet-50	96.9	89	93.5	93.1	

		Inception-v3	96.2	87.7	90	91.3
		DenseNet-201	95.7	89.4	92.1	92.4
		ResNet-101	98.8	89.6	94.6	94.3
		VGG -16	83.9	89.5	88.5	87.3
		VGG -19	87.2	89.4	97.9	91.5
2	Deep CNN features + ML model	CNN + KNN	87.8	86.8	89.1	87.9
		CNN + ANN	88.3	88.1	89.7	88.7
		CNN + SVM	96.4	87.7	89.3	91.1
3	Handcrafted features + ML model	LBP + KNN	32	30.7	44.2	35.6
		LBP + ANN	48.5	40.9	44.4	44.6
		LBP + SVM	45.1	35.9	35.2	38.7
		HOG + KNN	72	66.1	72.9	70.3
		HOG + ANN	92.4	88.3	97.8	92.8
		HOG + SVM	80.4	73.8	87.9	80.7
4	Proposed hybrid-ensemble technique	HOG and deep CNN features + ANN and SVM	99.2	89.6	98.4	95.7

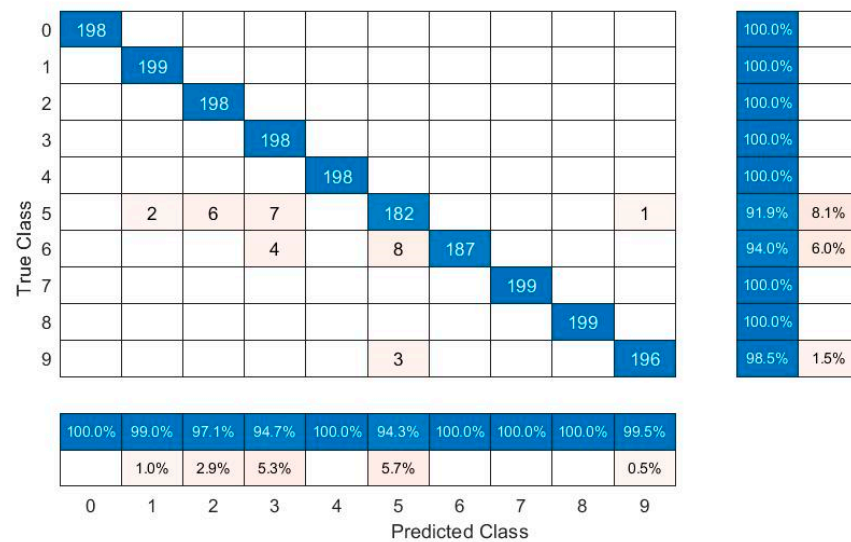
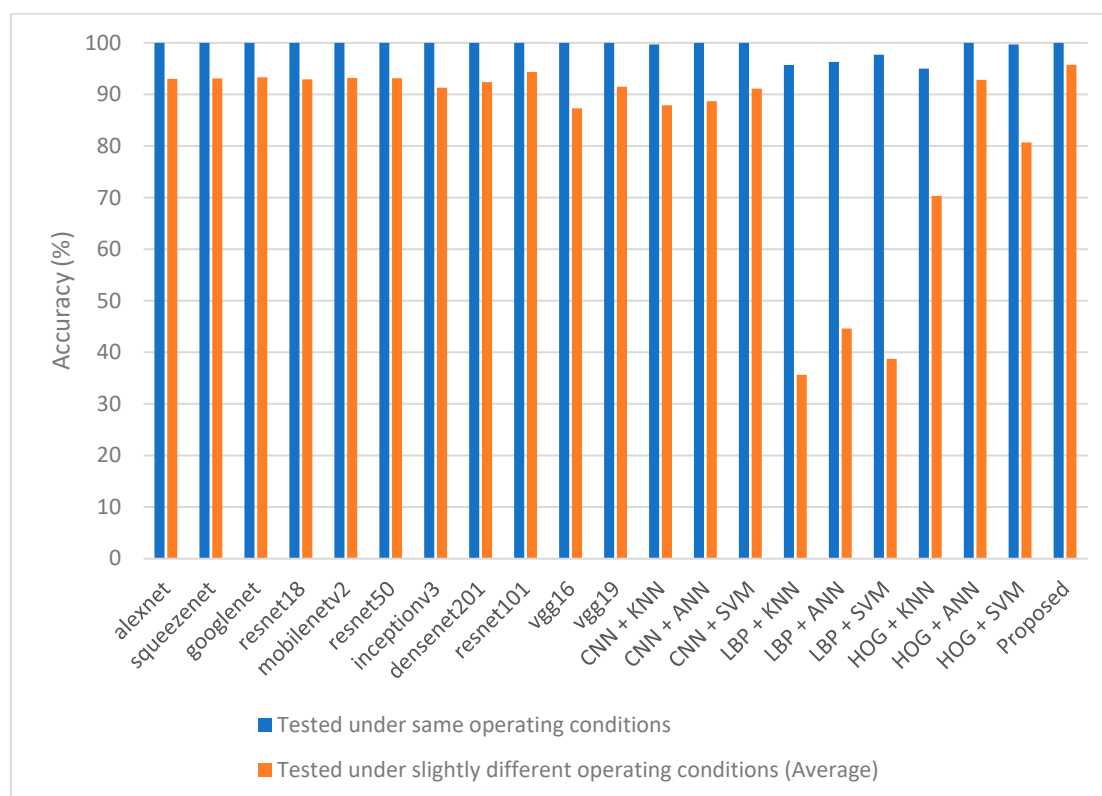


Figure 14. Confusion chart for the proposed method: training performed under OC-3 and tested under OC-4.



**Figure 15.** Impact of testing under slightly different operating conditions.

The application of different image-processing-based methods/techniques for the defect diagnosis of rolling element bearings under different limitations and conditions, as performed in this study, will give confidence to the researchers working in this field in using these image-processing-based defect diagnosis techniques which have not been widely used in the past, but specifically for this application. Moreover, the comparison performed can help researchers select appropriate image-processing-based defect diagnosis technique according to the applicable conditions and limitations. The performance of the proposed hybrid ensemble method remained comparable to that of the best CNN models for defect diagnosis in both cases of the sufficient and minimal availability of training data, and is computationally far less expensive as compared to CNN models. Moreover, for defect diagnosis under slightly different operating conditions, it outperformed all other considered image-processing-based techniques, including all the CNN models considered in this study. The analysis of the effect of different operating conditions on the defect diagnosis accuracy was performed at slightly different speeds ranging between 1720 rpm and 1797 rpm. However, the performance degradation may be more significant if the change in speed is substantial.

## 6. Conclusions

In this study, three image-processing-based intelligent defect diagnosis methods were compared for REBs' defect diagnosis using spectrogram images to ascertain their suitability for the defect diagnosis of REBs under: (1) the sufficient availability of training data; (2) the minimal availability of training data; and (3) slightly different operating conditions. Based upon the best image-features' types and ML models considered in this study, a hybrid-ensemble technique was proposed for the defect diagnosis of REBs.

Under the sufficient availability of training data, while using pretrained CNN models through transfer learning as end-to-end classifiers (Method-I), all the CNN models considered in this study achieved fault diagnosis with 100% accuracy; however, AlexNet was found to be least computationally expensive model and vgg19 was found to be the most

computationally expensive model among all the CNN models considered in this study. All the combinations considered in Method-II, a few combinations in Method-III and the proposed hybrid-ensemble method achieved fault diagnosis with an accuracy comparable to that of Method-I, despite being significantly less computationally expensive. By reducing the size of the training dataset to only 14 images, the defect diagnosis accuracies of six CNN models (ResNet18, MobileNetv2, ResNet50, Inceptionv3, DenseNet201 and ResNet101) was not significantly affected and remained above 99%; however, the defect diagnosis accuracies of the other CNN models considered in this study was drastically reduced. As the least computationally expensive model out of the six models whose performance remained excellent, ResNet18 was found to be the best model among the considered CNN models for defect diagnosis under the conditions of the minimal availability of training data. Among the less computationally expensive methods, the performances of Method-II and the proposed hybrid-ensemble method were not significantly affected and remained above 99%, except for deep features + KNN; however, the defect diagnosis accuracy of Method-III was significantly reduced under the condition of the minimal availability of training data. Overall, LBP features were found to be inferior to HOG and deep features, and ANN and SVM models were found to be superior to KNN due to their better performance in terms of fault detection accuracy.

For defect diagnosis under slightly different operating conditions, the proposed hybrid-ensemble method outperformed all other image-processing-based defect diagnosis methods considered in this study by exhibiting an average accuracy of 96%. The performance of all combinations using LBP features was the worst, which exhibited less than 45% defect diagnosis accuracy under slightly different operating conditions.

**Author Contributions:** Conceptualization, P.P. and S.M.T.; methodology, S.M.T.; software, S.M.T. and S.C.; validation, S.M.T. and S.C.; formal analysis, S.M.T.; investigation, S.M.T.; resources, P.P. and S.C.; data curation, S.M.T.; writing—original draft preparation, S.M.T.; writing—review and editing, P.P. and S.C.; visualization, S.M.T.; supervision, P.P.; project administration, S.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data supporting the findings of this paper are available at <https://engineering.case.edu/bearingdatacenter/download-data-file> (accessed on 25 April 2022).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Kiral, Z.; Karagülle, H. Simulation and analysis of vibration signals generated by rolling element bearing with defects. *Tribol. Int.* **2003**, *36*, 667–678. [https://doi.org/10.1016/S0301-679X\(03\)00010-0](https://doi.org/10.1016/S0301-679X(03)00010-0).
2. Orhan, S.; Aktürk, N.; Çelik, V. Vibration monitoring for defect diagnosis of rolling element bearings as a predictive maintenance tool: Comprehensive case studies. *Ndt E Int.* **2006**, *39*, 293–298. <https://doi.org/10.1016/j.ndteint.2005.08.008>.
3. Li, B.; Chow, M.Y.; Tipsuwan, Y.; Hung, J. Neural-network-based motor rolling bearing fault diagnosis. *IEEE Trans. Ind. Electron.* **2000**, *47*, 1060–1069. <https://doi.org/10.1109/41.873214>.
4. Samanta, B.; Al-Balushi, K.R. Artificial neural network based fault diagnostics of rolling element bearings using time-domain features. *Mech. Syst. Signal Processing* **2003**, *17*, 317–328. <https://doi.org/10.1006/mssp.2001.1462>.
5. Tayyab, S.M.; Asghar, E.; Pennacchi, P.; Chatterton, S. Intelligent fault diagnosis of rotating machine elements using machine learning through optimal features extraction and selection. *Procedia Manuf.* **2020**, *72*, 266–273. <https://doi.org/10.1016/j.promfg.2020.10.038>.
6. Seryasat, O.R.; Haddadnia, J.; Arabnia, Y.; Zeinali, M.; Aboalizadeh, Z.; Taherkhani, A.; Tabrizy, S.; Maleki, F. Intelligent fault detection of ball-bearings using artificial neural networks and support-vector machine. *Life Sci.* **2012**, *9*, 4186–4189.
7. Mao, W.; He, J.; Li, Y.; Yan, Y.G. Bearing fault diagnosis with auto-encoder extreme learning machine: A comparative study. *Proc. Inst. Mech. Eng. C J. Mech. Eng. Sci.* **2017**, *231*, 1560–1578. <https://doi.org/10.1177/0954406216675896>.
8. Su, H.; Chong, K.T. Induction machine condition monitoring using neural network modeling. *IEEE Trans. Ind. Electron.* **2007**, *54*, 241–249. <https://doi.org/10.1109/TIE.2006.888786>.

9. Wu, C.X.; Chen, T.F.; Jiang, R. Bearing fault diagnosis via kernel matrix construction based support vector machine. *J. Vibroengineering* **2017**, *19*, 3445–3461. <https://doi.org/10.21595/jve.2017.18482>.
10. Gunerkar, R.S.; Jalan, A.K.; Belgamwar, S.U. Fault diagnosis of rolling element bearing based on artificial neural network. *J. Mech. Sci. Technol.* **2019**, *33*, 505–511. <https://doi.org/10.1007/s12206-019-0103-x>.
11. Patel, J.P.; Upadhyay, S.H. Comparison between artificial neural network and support vector method for a fault diagnostics in rolling element bearings. *Procedia Eng.* **2016**, *144*, 390–397. <https://doi.org/10.1016/j.proeng.2016.05.148>.
12. Hussain, M.; Bird, J.J.; Faria, D.R. A Study on CNN Transfer Learning for Image Classification. In *Advances in Computational Intelligence Systems*; Springer: Cham, Switzerland, 2019; pp. 191–202. [https://doi.org/10.1007/978-3-319-97982-3\\_16](https://doi.org/10.1007/978-3-319-97982-3_16).
13. Eren, L. Bearing fault detection by one-dimensional convolutional neural networks. *Math. Probl. Eng.* **2017**, *2017*, 8617315. <https://doi.org/10.1155/2017/8617315>.
14. Eren, L.; Ince, T.; Kiranyaz, S. A generic intelligent bearing fault diagnosis system using compact adaptive 1D CNN classifier. *J. Signal Processing Syst.* **2019**, *91*, 179–189. <https://doi.org/10.1007/s11265-018-1378-3>.
15. Pham, M.-T.; Kim, J.-M.; Kim, C.-H. 2D CNN-Based Multi-Output Diagnosis for Compound Bearing Faults under Variable Rotational Speeds. *Machines* **2021**, *9*, 199. <https://doi.org/10.3390/machines9090199>.
16. Pham, M.-T.; Kim, J.-M.; Kim, C.-H. Accurate bearing fault diagnosis under variable shaft speed using convolutional neural networks and vibration spectrogram. *Appl. Sci.* **2020**, *10*, 6385. <https://doi.org/10.3390/app10186385>.
17. Verstraete, D.; Ferrada, A.; Droguett, E.L.; Meruane, V.; Modarres, M. Deep learning enabled fault diagnosis using time-frequency image analysis of rolling element bearings. *Shock. Vib.* **2017**, *2017*, 5067651. <https://doi.org/10.1155/2017/5067651>.
18. Tang, H.D.; Tran, X.T.; Van, M.; Kang, H.J. A Deep Neural Network-Based Feature Fusion for Bearing Fault Diagnosis. *Sensors* **2021**, *21*, 244. <https://doi.org/10.3390/s21010244>.
19. Tayyab, S.M.; Chatterton, S.; Pennacchi, P. Intelligent Defect Diagnosis of Rolling Element Bearings under Variable Operating Conditions Using Convolutional Neural Network and Order Maps. *Sensors* **2022**, *22*, 2026. <https://doi.org/10.3390/s22052026>.
20. Hoang, D.-T.; Kang, H.-J. Rolling element bearing fault diagnosis using convolutional neural network and vibration image. *Cogn. Syst. Res.* **2019**, *53*, 42–50. <https://doi.org/10.1016/j.cogsys.2018.03.002>.
21. Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; Li, F.-F. ImageNet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255. <https://doi.org/10.1109/CVPR.2009.5206848>.
22. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. <https://doi.org/10.1007/s11263-015-0816-y>.
23. Bai, Y.; Yang, J.; Wang, J.; Zhao, Y.; Li, Q. Image representation of vibration signals and its application in intelligent compound fault diagnosis in railway vehicle wheelset-axlebox assemblies. *Mech. Syst. Signal Processing* **2021**, *152*, 107421. <https://doi.org/10.1016/j.ymsp.2020.107421>.
24. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? *Adv. Neural Inf. Processing Syst.* **2014**, *27*, 1–9.
25. Udmale, S.S.; Singh, S.K.; Singh, R.; Sangaiah, A.K. Multi-fault bearing classification using sensors and ConvNet-based transfer learning approach. *IEEE Sens. J.* **2019**, *20*, 1433–1444. <https://doi.org/10.1109/JSEN.2019.2947026>.
26. Xie, W.; Li, Z.; Xu, Y.; Gardoni, P.; Li, W. Evaluation of Different Bearing Fault Classifiers in Utilizing CNN Feature Extraction Ability. *Sensors* **2022**, *22*, 3314. <https://doi.org/10.3390/s22093314>.
27. Khan, S.A.; Kim, J.M. Automated bearing fault diagnosis using 2D analysis of vibration acceleration signals under variable speed conditions. *Shock. Vib.* **2016**, *2016*, 8729572. <https://doi.org/10.1155/2016/8729572>.
28. Kaplan, K.; Kaya, Y.; Kuncan, M.; Minaz, M.R.; Ertunç, H.M. An improved feature extraction method using texture analysis with LBP for bearing fault diagnosis. *Appl. Soft Comput.* **2020**, *87*, 106019. <https://doi.org/10.1016/j.asoc.2019.106019>.
29. Chen, J.; Zhou, D.; Wang, Y.; Fu, H.; Wang, M. Image feature extraction based on HOG and its application to fault diagnosis for rotating machinery. *J. Intell. Fuzzy Syst.* **2018**, *34*, 3403–3412. <https://doi.org/10.3233/JIFS-169521>.
30. Tayyab, S.M.; Chatterton, S.; Pennacchi, P. Fault detection and severity level identification of spiral bevel gears under different operating conditions using artificial intelligence techniques. *Machines* **2021**, *9*, 173. <https://doi.org/10.3390/machines9080173>.
31. Er-Raoudi, M.; Diany, M.; Aissaoui, H.; Mabrouki, M. Gear fault detection using artificial neural networks with discrete wavelet transform and principal component analysis. *J. Mech. Eng. Sci.* **2016**, *10*, 2016–2029. <https://doi.org/10.15282/jmes.10.2.2016.6.0190>.
32. Yang, D.M.; Stronach, A.F.; MacConnell, P.; Penman, J. Third-order spectral techniques for the diagnosis of motor bearing condition using artificial neural networks. *Mech. Syst. Signal Process.* **2002**, *16*, 391–411. <https://doi.org/10.1006/mssp.2001.1469>.
33. Liu, R.; Yang, B.; Zio, E.; Chen, X. Artificial intelligence for fault diagnosis of rotating machinery: A review. *Mech. Syst. Signal Process.* **2018**, *108*, 33–47. <https://doi.org/10.1016/j.ymsp.2018.02.016>.
34. Baraldi, P.; Cannarile, F.; Di Maio, F.; Zio, E. Hierarchical k-nearest neighbours classification and binary differential evolution for fault diagnostics of automotive bearings operating under variable conditions. *Eng. Appl. Artif. Intell.* **2016**, *56*, 1–13. <https://doi.org/10.1016/j.engappai.2016.08.011>.
35. Lei, Y.G.; Yang, B.; Jiang, X.W.; Feng, J.; Li, N.P.; Asoke, K.N. Applications of machine learning to machine fault diagnosis: A review and roadmap. *Mech. Syst. Signal Processing* **2019**, *138*, 106587. <https://doi.org/10.1016/j.ymsp.2019.106587>.

36. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 886–893. <https://doi.org/10.1109/CVPR.2005.177>.
37. Kumar, M.D.; Babaie, M.; Zhu, S.; Kalra, S.; Tizhoosh, H.R. A comparative study of CNN, BoVW and LBP for classification of histopathological images. In Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence (SSCI), Honolulu, HI, USA, 27 November 2017–1 December 2017; IEEE: Piscataway, NJ, USA, 2017. <https://doi.org/10.1109/SSCI.2017.8285162>.
38. Case Western Reserve University Bearing Data Center. Available online: <https://engineering.case.edu/bearingdatacenter/download-data-file> (accessed on 25 April 2022).
39. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Adv. Neural Inf. Processing Syst.* **2012**, *60*, 84–90.
40. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360. <https://doi.org/10.48550/arXiv.1602.07360>.
41. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9. <https://doi.org/10.1109/CVPR.2015.7298594>.
42. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26–30 June 2016; pp. 770–778. <https://doi.org/10.48550/arXiv.1512.03385>.
43. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2018, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520. <https://doi.org/10.48550/arXiv.1801.04381>.
44. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2016, Las Vegas, NV, USA, 26–30 June 2016; pp. 2818–2826. <https://doi.org/10.48550/arXiv.1512.00567>.
45. Huang, G.; Liu, Z.; van der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708. <https://doi.org/10.48550/arXiv.1608.06993>.
46. Karen, S.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556. <https://doi.org/10.48550/arXiv.1409.1556>.