Article

Unbalance Detection in Induction Motors through Vibration Signals Using Texture Features

Uriel Calderon-Uribe [1], Rocio A. Lizarraga-Morales [2],* and Igor V. Guryev [1]

[1] Departamento de Estudios Multidisciplinarios, División de Ingenierías, Campus Irapuato-Salamanca, Universidad de Guanajuato, Yuriria 38944, Guanajuato, Mexico; u.calderonuribe@ugto.mx (U.C.-U.); guryev@ugto.mx (I.V.G.)

[2] Departamento de Arte y Empresa, División de Ingenierías, Campus Irapuato-Salamanca, Universidad de Guanajuato, Salamanca 36885, Guanajuato, Mexico

* Correspondence: ra.lizarragamorales@ugto.mx

Abstract: The detection of faults in induction motors has been one of the main challenges to the industry in recent years. An effective fault detection method is fundamental to ensure operational security and productivity. Different models for intelligent fault diagnosis have been recently proposed. However, not all of them are accessible for some manufacturing processes because of the black-box approach, the complexity of hyperparameter tuning, high-dimensionality feature vectors, and the need for sophisticated computational resources. In this paper, a method for the detection of an unbalance fault in induction motors based on a low-dimensional feature vector and a low-complexity classification approach is proposed. The feature vector presented in this manuscript is based on texture features, which are a basic tool for image processing and image understanding. Nevertheless, texture features have not been explored as a powerful instrument for induction motor fault analysis. In this approach, texture features are used to analyze a set of vibration signals belonging to two different classes: an unbalanced motor and a healthy motor. Training-validation and testing stages are developed to build and evaluate the performance of the classifier, respectively. The results show higher accuracy and lower training time in comparison with different state-of-the-art approaches.

Keywords: signal processing; intelligent diagnosis; supervised classification; unbalance detection

1. Introduction

Nowadays, Induction Motors (IM) are one of the most important elements used in industrial processes. IM play an important role in the industry and can be found in areas such as power generation, automobile, aerospace, and many others [1–5]. However, exposure to hard conditions of work affects the functionality of these machines, making them vulnerable to different types of faults. The broad number of possible faults can be classified as bearing faults, rotor faults, and stator faults, which have an incidence factor of 50%, 10%, and 40%, respectively [6]. Within the bearing faults, unbalance represents one of the most common faults in IM [7]. Problems such as the decrease in the lifetime of the bearings, and damage to the machinery, can be generated by not detecting unbalances on time. Therefore, early identification of the unbalance fault is an essential element for preventive and corrective actions that avoid problems in industrial processes.

Due to the importance of unbalance fault detection in IM, different machine-learning methods have been used recently to diagnose motor conditions. These machine learning methods can be divided into two different groups: deep neural network classifiers and statistical features classifiers [8–11]. In deep neural network classifiers, the main purpose is to create a model that allows the decomposition of a signal in representative features to infer the motor condition. Within unbalance classification, convolutional neural networks (CNNs) and recurrent neural networks (RNNs) are the most used models to detect this fault [12,13]. On the one hand, Mey et al. [14] proposed an explainable CNNs to classify

unbalance faults in IM. In the Mey et al. approach, different kernels are used to compute the convolution and extract features from the short-time Fourier transform matrix set. Mey et al. achieved 99.7% of accuracy in the identification of an unbalance fault. On the other hand, RNNs are models based on internal states that allow the remembrance of the faults of the motors. Xiao et al. [15] presented an RNN used to classify the unbalance fault. In the Xiao et al. approach, the acceleration signals and a set of 29 statistical features are used to train and test the RNN. Xiao et al. achieved a 98.9% of accuracy in the detection of the unbalance fault. Although the deep neural network approaches allow the classification of the unbalance fault in IM, they are complex in terms of hyperparameter tuning. In addition, the training time and the computational resources are often expensive, hindering the implementation of a monitoring system in the industry [16].

Statistical classifiers arise as an alternative to complex black-box models. These approaches are based on statistical measurements that allow the description of a given signal. In the detection of faults in IM, these models have been used in the classification of faults such as broken rotor bars, rotor misalignment, bearing faults, unbalance, and many others [17–20]. Within the unbalance fault, signal analysis can be performed in both time or frequency domains. In the time domain, Tahir et al. [21] extracted the statistical features of kurtosis, impulse factor, skewness, RMS, range, and variance in the time domain to classify the unbalance fault with 86.87% accuracy. The detection of faults using the frequency domain signals can also be achieved using statistical features. Gangsar et al. [22] proposed a set of statistical features based on the frequency domain and the time domain to infer the unbalance fault. The features vector is composed of standard deviation, skewness, and kurtosis. Using a support vector machine to classify the unbalance fault, Gangsar et al. [22] achieved 84.51% of accuracy. Although the methods previously mentioned achieved the classification of the unbalanced fault, the use of high-dimensional feature vectors and low representative features makes it difficult to obtain high accuracy in the final model [23]. For these reasons, new features need to be studied to decrease the computational load and keep the accuracy rates high.

To overcome the limitations generated by neural networks and the high-dimensional feature vectors used in statistical classifiers, this study proposes a novel technique for the unbalance fault diagnosis based on texture features and a classification approach. We have named our proposal VTA for Variance-correlation Texture Analysis. Although texture features have been proposed for the classification of different motor faults [24,25], texture has never been explored for the detection of unbalance faults. This approach consists of analyzing vibration signals by using a subset of texture features proposed by Unser [26]. Subsequently, a support vector machine (SVM) [27] is trained and validated to infer the unbalance and healthy classes in the IM. Finally, a test set is used to measure the performance of the classifier. Experiments from this study may lead to an improvement in the accuracy and training time in comparison to other models established in the state-of-the-art.

The rest of the manuscript is structured as follows: Section 2 describes the methodology used to build our VTA approach with an SVM classifier using texture features. In Section 3 the experiments and results are discussed. Finally, in Section 4 the conclusions are presented.

2. Methodology

The methodology proposed for the VTA approach is described in this section. The main goal of this study is to develop a system that allows the classification of the unbalance fault in IM using a low-dimensionality feature vector. This low-dimensionality feature vector is composed of correlation and variance texture features. An overview of the proposed methodology is shown in Figure 1. From this figure, it can be observed that the classification system is divided into two stages: training-validation, and testing. In the training-validation stage, two different subsets of vibration signals are used to extract the correlation and variance texture features. In the testing stage, the correlation and variance

texture features are extracted from a different subset of vibration signals to evaluate the final classifier and measure the performance from the VTA model.
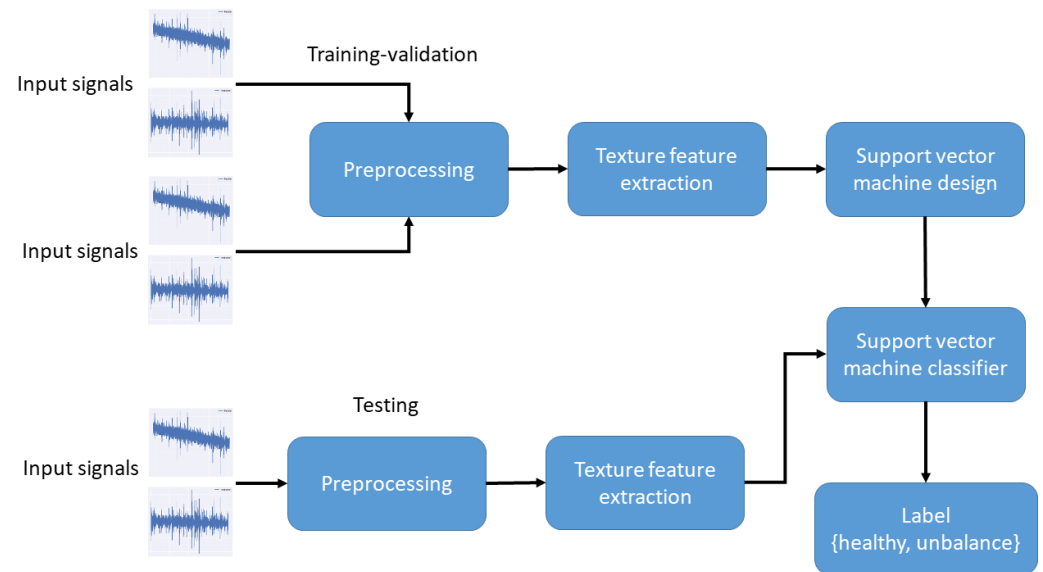

Figure 1. Overall VTA model design to infer the unbalance fault in IM.

2.1. Unbalance Fault

In IM fault analysis, the unbalance fault can be defined as the displacement of the center of mass out of the rotating axis. This displacement is generated when the mechanical load m is not distributed uniformly r. Thus, the unbalance fault can be described according to the Equation (1):

$$U = m \times r, \tag{1}$$

where r represents the eccentricity and m represents the mass. Frequency components are generated in the rotor and stator components because of the unbalance fault. These frequencies are used to distinguish the unbalance fault [28]. The frequencies generated for the unbalance fault in the IM can be defined in Equation (2)

$$f_{unb} = f_s \left(\frac{1 \pm k(1-s)}{p} \right), \quad k = 1, 2, 3, \dots \tag{2}$$

where p represents the pole in the IM, s is the per unit slip, and f_s is the frequency generated by the electrical supply.

2.2. Texture Feature Extraction

In image processing, texture is a property used to measure the variability of gray levels through an image. Such measures can be used as image descriptors. These descriptors are used in image processing to classify different elements contained in an image. There are different statistical features used to describe texture in an image. Although these properties are commonly used in 2-D signal analysis, they can be computed in 1-D.

The texture as a statistical concept has its origin in the gray-level co-occurrence matrix proposed by Haralick et al. [29]. However, since the computational cost is expensive, Unser et al. [26] proposed the sum and difference histograms as an alternative to the co-occurrence matrix. According to Unser, considering a digital image I with size $M \times N$ denoted by $I(i,j), \{i = 1, \dots, M; j = 1, \dots, N\}$ being $G = \{1, 2, \dots, N_g\}$ the set of the N_g quantized grey levels, taking two points in a relative position fixed by u and v, $I(i,j)$, and

$I(i + u, j + v)$, the sum and difference between these two points can be defined by the Equations (3) and (4).

$$S_{i,j,u,v} = I(i,j) + I(i+u, j+v), \tag{3}$$

$$D_{i,j,u,v} = I(i,j) - I(i+u, j+v), \tag{4}$$

where u and v represent the horizontal and vertical displacement, respectively. From the sum and difference, the joint probability functions with parameters u and v are defined in the Equations (5) and (6).

$$h_s(k) = Card = \{(i,j) \in D, S_{i,j} = k\}, \tag{5}$$

$$h_d(l) = Card = \{(i,j) \in D, D_{i,j} = l\}, \tag{6}$$

where D is a subset of indexes specifying a texture region to be analyzed. Then, the normalization of the sum and difference histograms are shown in Equations (7) and (8).

$$P_s(k) = \frac{h_s(k)}{M \times N}; \quad (k = 2, \dots, 2N_g), \tag{7}$$

$$P_d(l) = \frac{h_d(l)}{M \times N}; \quad (l = -N_g + 1, \dots, N_g - 1). \tag{8}$$

From the sum and difference histograms, a subset of two texture features defined by Unser can be extracted. These texture features are defined in Equations (9) and (10).

$$variance = \frac{1}{2} \left\{ \sum_k (k - \mu)^2 \cdot P_s(k) + \sum_l l^2 \cdot P_d(l) \right\}, \tag{9}$$

$$correlation = \frac{1}{2} \left\{ \sum_k (k - \mu)^2 \cdot P_s(k) - \sum_l l^2 \cdot P_d(l) \right\}. \tag{10}$$

However, this process can be modified such that texture features (Equations (9) and (10)) can be computed in a signal in the time domain. From the Equations (3) and (4), u and v represent the horizontal and vertical displacement, respectively. In a signal in the time domain, the vertical displacement cannot be achieved due to the signal dimensionality, so only the horizontal displacement can be computed. Nevertheless, this is enough to establish a relationship between two points in different positions. Considering a digital signal $f(t)$ with N samples such that $t \in \{0, 1, 2, 3, \dots, N - 1\}$, and with K different values in $f(t)$, that is $f(t) \in \{0, 1, 2, 3, \dots, K - 1\}$; taking two points in a relative position given by d, $f(t)$, and $f(t + d)$, the sum and difference given by these two points can be defined according to the Equations (11) and (12).

$$S_{t,d} = f(t) + f(t + d), \tag{11}$$

$$D_{t,d} = f(t) - f(t + d). \tag{12}$$

In Equations (3) and (4), the main objective is to compare two points, $I(i,j)$ and $I(i+u, j+v)$ given the parameters u and v. This same behavior is achieved in Equations (11) and (12) when comparing the points $f(t)$ and $f(t + d)$, given a displacement d. Thus, the Equations (11) and (12) can be used to calculate the texture features developed in the Equations (9) and (10). In this study, the variance and correlation are used as the feature vectors to classify the unbalance and healthy classes in IM.

2.3. Support Vector Machine Classifier

The support vector machine (SVM) is a machine learning algorithm that has been widely used in classification tasks. The main goal of the SVM is to create a decision boundary or hyperplane between two or more classes that allows the prediction of labels

from a feature vector [30]. According to Steinwart et al. [30], in a dataset $F^l = \{(x_i^l, y_i^l)\}_{i=1}^n$ with $x_i^l = (x_{1,i}^l, \ldots, x_{v,i}^l)^T \in \mathcal{R}^v$, the method is used to learn a function $z \in Z : x^{(l)} \mapsto y$. This relation can generate a decision boundary (hyperplane) by maximizing the separation between two or more classes. In the function F^l, y_i^l represents the labels from the vector x_i^l. According to Christmann et al. and Gieseke et al. [30,31], it is assumed that $\{(x_i^l, y_i^l)\}_{i=1}^n$ are independents and they are distributed in a range $\mathcal{M} \times \mathcal{N}$. Here, $\mathcal{M}$ represents the set of samples in the dataset and $\mathcal{N}$ represents the set of labels [30]. Thus, the SVM is considered a regularization problem and can be described by the Equation (13):

$$\inf_{z \in \mathcal{H}, z \in Z} \left\{ \frac{1}{n} \sum_{i=1}^n \mathcal{L}(y_i', z(x_i^l)) + \lambda ||z||_{\mathcal{H}}^2 \right\}, \tag{13}$$

where $\lambda \in \mathbb{R}^+$, $\mathcal{L} : \mathbb{R} \mapsto [0, \infty)$ is the lost function, and $||z||_{\mathcal{H}}^2$ is the norm Hilbert space $\mathcal{H} \subseteq \mathbb{R}^{\mathcal{M}} = z : \mathcal{M} \mapsto \mathbb{R}$. However, the unbalance set has a non-lineal behavior, hence an efficient way to deal with this problem is to map the input space $\mathcal{R}^v$ to $\mathcal{H}$ using a kernel [32]. Different kernels are available in the literature (linear kernel, polynomial kernel, and radial basis function kernel), and selecting the appropriate kernel plays an important role in the classification process. In this study, an empirical process was conducted, and the radial basis function kernel [33] was selected since it yielded better results. Thus, the Gaussian kernel is defined according to the Equation (14):

$$k_\sigma, \mathcal{C}^d = exp\left(-\sigma^2 \sum_{j=1}^d (x_j - x_j')^2\right), \tag{14}$$

where $d \in \mathcal{N}$, $x = (x_1, x_2, \ldots, x_d) \in \mathcal{C}^d$, $x' = (x_1', x_2', \ldots, x_d')$, and k_σ can be calculated by Equation (15).

$$k_\sigma(x, x') = exp\left(-\frac{||x - x'||_2^2}{\sigma}\right), \quad x, x' \in \mathcal{R}^d. \tag{15}$$

The σ hyperparameter is tuned by using the cross-validation method to avoid overfitting and was set in $\sigma = 3$ using the validation set.

3. Results

In this section, the experimentation to evaluate and measure the VTA approach is presented. The evaluation is conducted by a thorough quantitative analysis on a standard database. Firstly, the experimental setup used to evaluate the classifier performance is presented. Afterward, the training results, the feature selection, and the horizontal displacement are discussed. Finally, the VTA model is compared with classical models and state-of-the-art approaches.

3.1. Experimental Setup

In order to measure the performance of the VTA approach proposed in this study, different signals of unbalance were considered. For the experimentation, the dataset proposed by Mey et al. from the Fraunhofer Institute for Integrated Circuits, Germany [14] was used. The motor used in the Mey et al. study is an electronically commutated DC motor (WEG GmbH, type UE 511 T), the data acquisition was made by a 4-channel data acquisition system PCB Synotech GmbH, type FRE-DT9837, and the sensor used to extract the vibration signals was PCB Synotech GmbH, type PCB-M607A11/M001AC. To generate the fault, an unbalance holder is attached behind a roller bearing block. The unbalance holder consists of a disc with axially symmetric recesses, in which weights can be inserted to simulate unbalances. A complete description of the fault generation and specifications about the dataset generation can be founded in the article [34]. The dataset consists of signals that represent the unbalance fault and healthy classes. In Figure 2, samples of the signals corresponding to the unbalance and healthy classes are shown. From this figure, it

can be observed that although there are differences between the signals, it can be difficult for an expert to detect this fault. Therefore, signal processing is required. For this study, a set composed of 5000 signals was used to extract the texture features. This set was formed by 2500 signals which represented the healthy class and the rest formed the unbalance class. Each signal contained 4096 samples normalized between 0 and 255, according to those established in the image analysis. The final set was formed by applying the sum and difference histograms and the variance and correlation texture features. Thus, the final dataset was composed of 5000 × 2 values, where 5000 represented the samples in the final dataset and 2 represented the dimensionality. In order to generate the training-validation and testing sets, two subsets of the final set were formed. The training-validation subset was formed for 4000 samples (2000 samples representing the unbalance fault and the rest representing the healthy class), which was 80% of the final dataset, where 800 samples (400 samples representing the unbalance class and the rest representing the healthy class) were used to tune the hyperparameter in the SVM. The testing subset was formed by 1000 samples where 500 samples represented the unbalance class and the rest represented the healthy class. The test set was used to evaluate the tuned model.
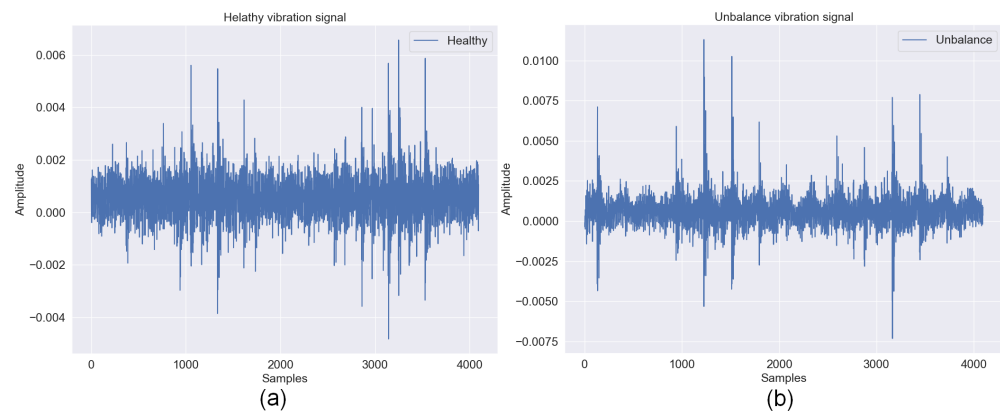

Figure 2. Vibration signals used to extract texture features, (**a**) healthy class, (**b**) Unbalance fault.

In order to follow the same evaluation setup as Mey et al. [14], the accuracy metric was used to evaluate the performance of the model. This metric is defined as how close a given set of measurements (observations or readings) are to their true value and is represented by Equation (16):

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN},$$ (16)

where TP represents the healthy class correctly assigned, TN corresponds to the unbalance instance correctly classified, FP represents the healthy instance classified as the unbalance class, and FN corresponds to the unbalance class classified as the healthy instance.

3.2. Texture Feature Selection

In this section, the exploration, analysis, and selection of texture features are presented. There are previous works that use texture features as tools to describe different faults in IM. In the study presented by Ferrucho et al. [24] and Lizarraga et al. [25] the contrast and homogeneity texture features were used to detect bearing faults in IM. In order to evaluate the viability of these features and include others, combination tests were carried out and their feasibility was qualitatively evaluated. Figure 3 shows an empirical analysis that shows examples of combinations of the previously mentioned texture features. From Figure 3a,c, the features are overlapping, making it difficult to create a decision boundary that allows separation of the unbalance and healthy classes. In Figure 3b the samples show a small separation allowing the creation of a model that classifies these faults. However, the accuracy achieved with these texture features is less in comparison with the features

used in this study. Figure 3d shows a separation generated by the correlation and variance texture features. This separation has a non-linear behavior, hence a classifier that allows the manipulation of the non-linearity is required.
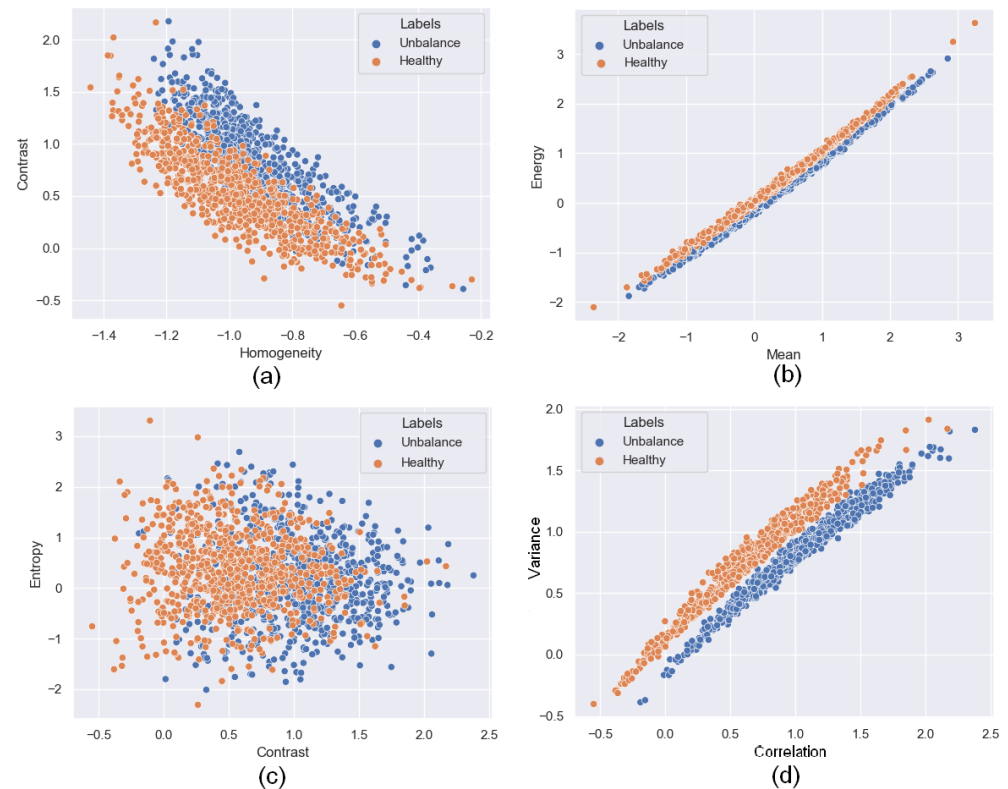

Figure 3. Subset of texture features used for determining the better separability between the unbalance class and the healthy class, (**a**) Homogeneity and contrast texture features, (**b**) Mean and energy texture features, (**c**) Contrast and entropy texture features, (**d**) Contrast and variance texture features.

3.3. Horizontal Displacement Selection

The correlation and variance texture features have hyperparameters to tune. From Equations (11) and (12), its evaluation depends on horizontal displacement d. This displacement d controls the number of values to compare. Figure 4 shows an empirical analysis from different horizontal displacements allowing select $d = 1$ as the best parameter. For the values, $d = 2$ and $d = 3$ the separation is minimum, making the creation of a decision boundary using the SVM difficult. For the horizontal displacement $d = 4$, the samples are completely superimposed, preventing the generation of a hyperplane using the SVM.

3.4. Training Results

The decision boundary (or hyperplane) generated from the SVM classifier is shown in Figure 5. In this figure, the red zone represents the healthy class and the blue zone represents the unbalance class. Note, the healthy class and unbalance class are close to each other. However, the SVM is able to create a hyperplane that allows the separation of both classes. In the SVM classifier, a semicircular hyperplane is generated through the radial basis function kernel, which describes the non-linearity between the classes. Linear models can be applied to describe this non-linearity. However, a lower accuracy can be achieved using this kind of model. Figure 6 shows the decision boundary generated by two linear models (logistic regression and simple perceptron). From this figure, it can be observed that although the unbalance fault can be inferred, samples of both classes are misclassified, achieving a lower accuracy in comparison with the proposed VTA model.
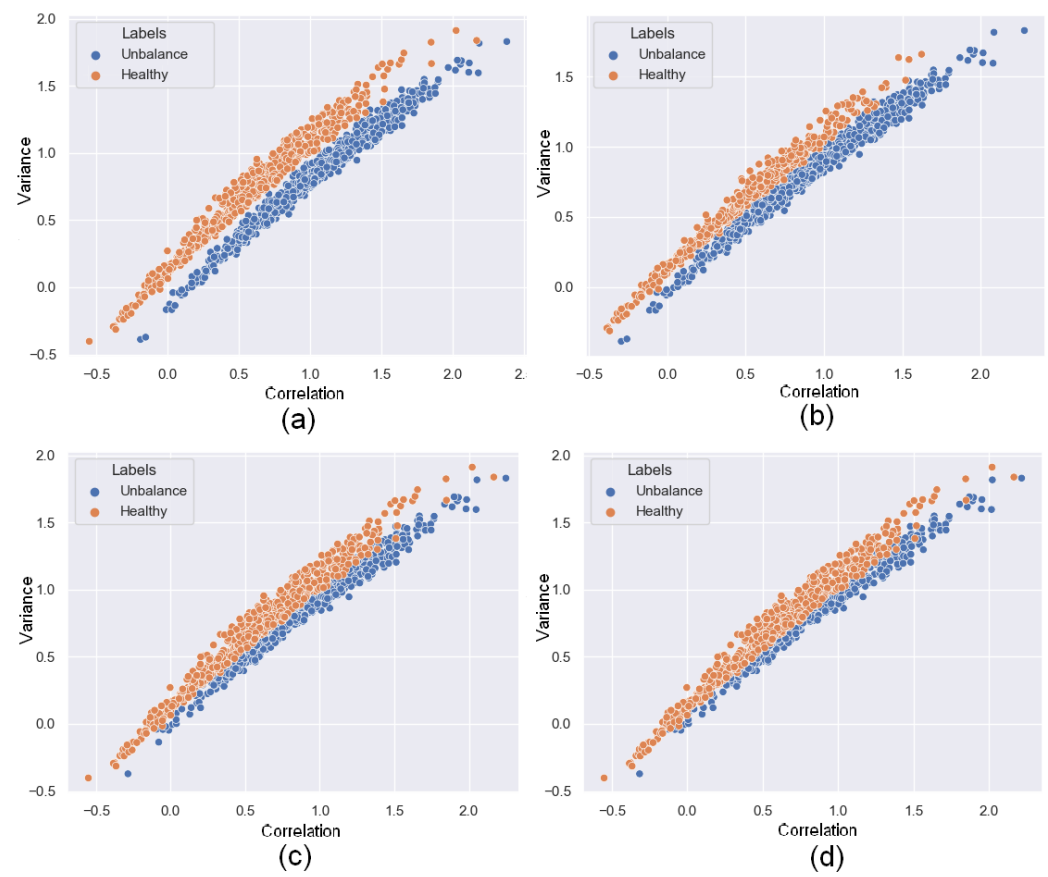
Figure 4. Variation of the horizontal displacement *d*, (**a**) Horizontal displacement $d = 1$, (**b**) Horizontal displacement $d = 2$, (**c**) Horizontal displacement $d = 3$, (**d**) Horizontal displacement $d = 4$.
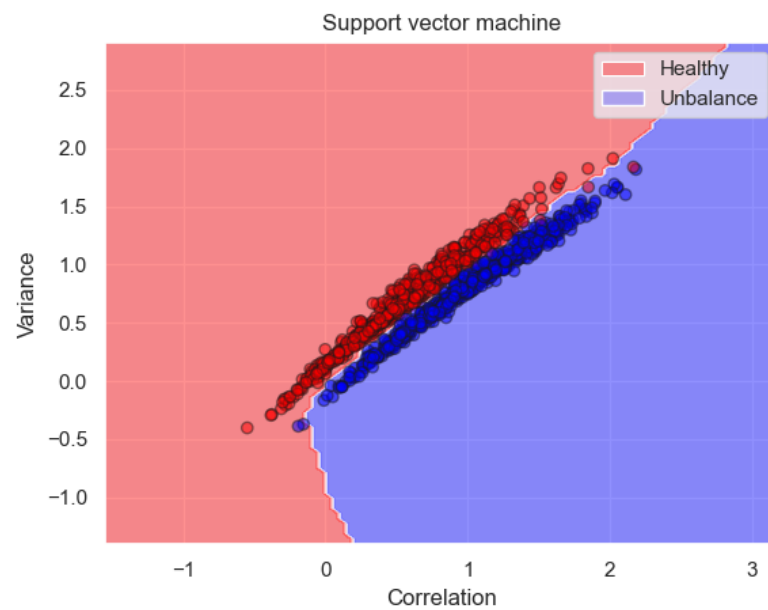

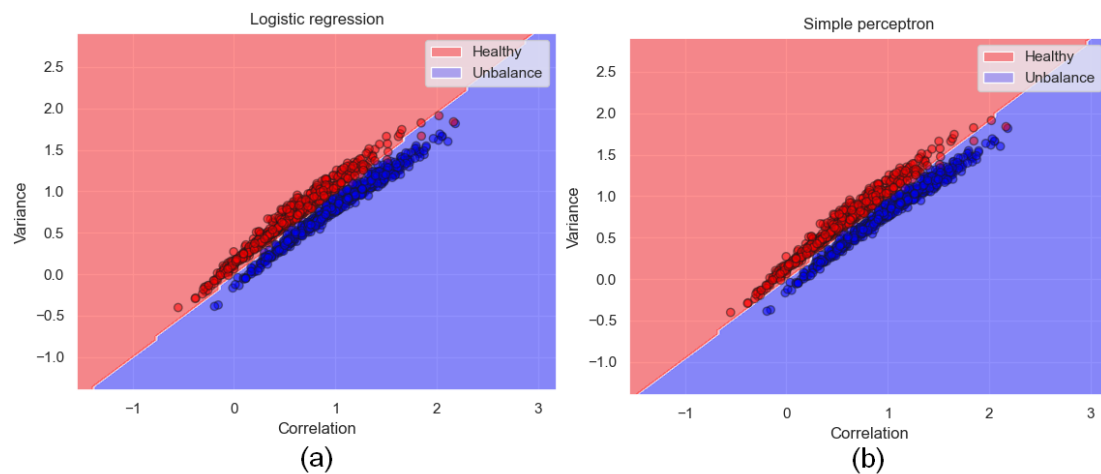**Figure 5.** Decision boundary generated for the SVM in the training-validation stage.

Figure 6. Decision boundary generated for two linear models. The models were trained with the same dataset. (**a**) Logistic regression, (**b**) Simple perceptron.

3.5. Classification Results

The results obtained from the proposed VTA model are shown in Table 1. From the test set compounded of 1000 samples, 997 samples were correctly classified. From these 997 samples, 498 samples correspond to the healthy class, and 499 correspond to the unbalance class. From the remaining samples, two samples corresponding to the healthy class were misclassified to the unbalance class. One sample corresponding to the healthy class was misclassified to the unbalance class. According to these results, the average accuracy obtained with our VTA approach was 99.7%. The remaining 0.3% corresponds to the misclassified samples. Figure 7 shows the learning curve from the VTA model. From this figure, batches of samples of the test set are used to measure and evaluate the performance of the SVM. Note, the performance of the VTA model is increased according to the number of samples used in the test set.
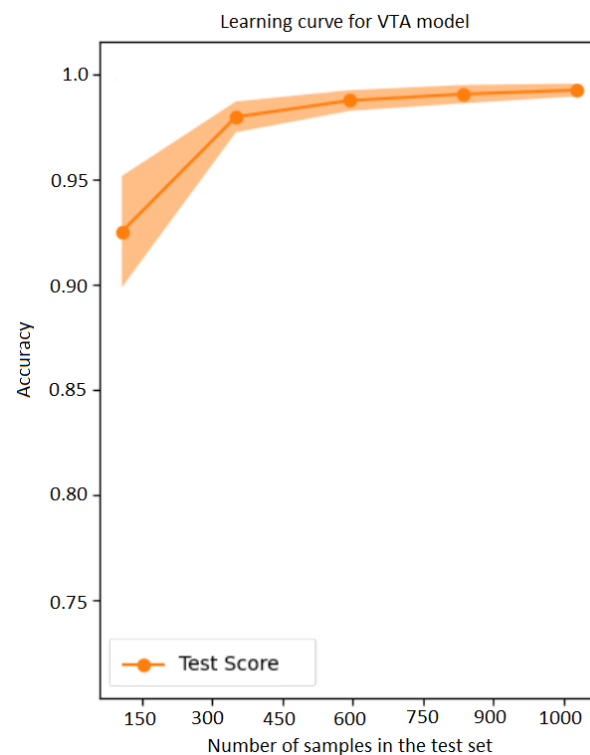

Figure 7. VTA model learning curve.

Table 1. Classification results from the VTA approach.

Class	Healthy	Unbalance
healthy	498	2
unbalance	1	499

Finally, the time consumed to train the VTA model using an Intel core i9 processor to 2.6 GHz was approximately 0.55 min.

3.6. Comparison

This section describes the comparison made to the VTA model. First, the model is compared with three classic machine learning models. This comparison aims to choose a model according to a specific need (in this case the accuracy). Finally, the VTA approach is compared with different state-of-the-art models. In this last comparison, the objective is to analyze the cost according to the complexity and computational resources.

Firstly, the VTA model proposed in this study is compared with three classic machine learning models; logistic regression (LR), K-nearest neighbors (KNN), and a simple perceptron (SP) are used as comparative models. These models were trained using the same training-validation and test sets. Table 2 shows a comparison between the VTA model and classic machine learning classifiers.

Table 2. Classification results using classic machine learning models.

Model	Accuracy %	Tunable Parameters	Dimensionality	Training Time Minutes
LR	99.3	1	1×2	0.36
SP	99.5	1	1×2	0.31
KNN	99.3	1	1×2	0.28
VTA model	99.7	1	1×2	0.55

From this Table 2, the accuracy achieved for the VTA approach is superior to the classic models. Our VTA model achieves an accuracy of 99.7% in comparison with the 99.3% achieved by the LR and KNN, and the 99.5% achieved by the SP, respectively. This is because the classic models are based on linear approximation, preventing the capture of the subjacent structure of the data (non-linearity). On the one hand, the time consumption of each model is considerably low, allowing the industry to use these models as a monitoring tool only if time is the most important resource to save. On the other hand, if both accuracy and time are the most important resources to use, the VTA model has the best qualities for the industry.

Finally, a comparison with different approaches is presented in Table 3. These models are the most popular tools in the state-of-the-art in IM fault detection: Convolutional neural networks [14,35], Long-short term memory neural networks [15], and artificial neural networks [36]. From this table, we can see that these approaches represent a powerful tool for the classification process. The closest approach is the one proposed by Mey et al. in terms of accuracy with 99.6% in comparison with our 99.7%. As we can see from this table, the number of features required by other approaches is considerably higher in comparison with our VTA. In addition, the dimensionality of each method is considerably higher, which makes it susceptible to overfitting [23]. Additionally, the tuning process is complex due to the number of hyperparameters (hidden layers, filters in convolutions, activation functions, padding, dropout layers, learning rate, regularization factor, etc.). Finally, a comparison with the model presented by Mey et al. is presented. This model is trained according to the specifications established in his paper [14]. Note, the time consumption for training the model is 15 min. Specific hardware may be required to decrease training time. Furthermore,

testing different ANN configurations becomes a complicated task due to the time it takes to train and test different models.

Table 3. Comparison with different state-of-the-art models in terms of Dimensionality (Dim), Number of hyperparameters to be tuned (No. Hyper), Training Time in minutes (TT), and Accuracy (Acc).

Approach	Method	Dim	No. Hyper	TT (min)	Acc %
Guo et al. [35]	CNN	128×128	256	-	85.5
Xiao et al. [15]	LSTM	16×64	256	-	98.2
Yongbo et al. [36]	ANN	$100 \times 100 \times 3$	96	-	98.6
Mey et al. [14]	CNN	600×300	16	15	99.6
VTA approach	Texture features	1×2	1	0.55	99.7

In comparison with the VTA approach, the hyperparameters are reduced to one σ, in comparison with the models presented by [15,35] where the number of hyperparameters to tune is considerably higher (256 hyperparameters to tune). This improvement allows us to test different model configurations quickly. The dimensionality is reduced to two features, in comparison with the [14,36] approaches where the inference process requires the evaluation of 30,000 and 18,000 values, respectively. This improvement avoids using specific hardware to train the model. In addition, the training time is reduced in comparison with the Mey et al. approach, which allows us to tune and test different models quickly. Although some models presented in the Table 3 do not show the training time, it is possible to consider that the training time is similar or higher than the model proposed by Mey et al. This is because the dimensionality and the hyperparameter tuning are higher in comparison with the Mey et al. approach. Finally, considering that the method only needs the vibration signals and that the signals are obtained during the motor functioning, the VTA model can be used during normal operation. The only delay is the time for taking the sample of the signal, plus the analysis time.

4. Conclusions

Unbalance faults are among the most recurrent faults in IM. For this reason, many research works are focused on the detection of these faults. Recent studies show the use of deep learning models to detect unbalance faults. However, specific hardware and complex feature vectors are used to detect these faults. Therefore, in this paper, a VTA approach to unbalance detection in IM is presented. This approach consists of the extraction of texture features to build and tune an SVM. The texture features are extensively used in image processing. However, it has not been used to unbalance detection in IM. The results of this study show an improvement in training time and accuracy in comparison to classical methods and state-of-the-art approaches. In addition, the dimensionality and the hyperparameter tuning were reduced considerably. The performance obtained, in comparison with other models, makes the VTA approach a viable option for the industry. Considering the broad number of occurring faults in induction motors, future work will include the detection of different types of faults using this approach.

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

References

1. Akimov, D.A.; Matyukhina, E.N.; Ignatov, A.S. Application of recurrent neural network in turbine control with regard to thermal expansion. *Educ. Transform. Issues* **2018**, *3*, 1–18.
2. Aktas, M.; Awaili, K.; Ehsani, M.; Arisoy, A. Direct torque control versus indirect field-oriented control of induction motors for electric vehicle applications. *Eng. Sci. Technol. Int. J.* **2020**, *23*, 1134–1143. [CrossRef]
3. Ismagilov, F.R.; Vavilov, V.E.; Gusakov, D.V. Line-Start Permanent Magnet Synchronous Motor for Aerospace Application. In Proceedings of the 2018 IEEE International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles & International Transportation Electrification Conference (ESARS-ITEC), Nottingham, UK, 7–9 November 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–5.
4. Choudhary, A.; Goyal, D.; Shimi, S.L.; Akula, A. Condition monitoring and fault diagnosis of induction motors: A review. *Arch. Comput. Methods Eng.* **2019**, *26*, 1221–1238. [CrossRef]
5. AlShorman, O.; Irfan, M.; Saad, N.; Zhen, D.; Haider, N.; Glowacz, A.; AlShorman, A. A review of artificial intelligence methods for condition monitoring and fault diagnosis of rolling element bearings for induction motor. *Shock Vib.* **2020**, *2020*, 1–20. [CrossRef]
6. Thomson, W.T.; Fenger, M. Current signature analysis to detect induction motor faults. *IEEE Ind. Appl. Mag.* **2001**, *7*, 26–34. [CrossRef]
7. Cubert, J.M. Use of electronic controllers in order to increase the service life on asynchronous motors. In Proceedings of the European Seminar on Electro-Technologies for Industry, Budapest, Hungary, 20–22 May 1992; pp. 393–404.
8. Kumar, P.; Hati, A.S. Review on machine learning algorithm based fault detection in induction motors. *Arch. Comput. Methods Eng.* **2021**, *28*, 1929–1940. [CrossRef]
9. Lei, Y.; Yang, B.; Jiang, X.; Jia, F.; Li, N.; Nandi, A.K. Applications of machine learning to machine fault diagnosis: A review and roadmap. *Mech. Syst. Signal Process.* **2020**, *138*, 106587. [CrossRef]
10. 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. [CrossRef]
11. Jiao, J.; Zhao, M.; Lin, J.; Liang, K. A comprehensive review on convolutional neural network in machine fault diagnosis. *Neurocomputing* **2020**, *417*, 36–63. [CrossRef]
12. Shao, S.; Yan, R.; Lu, Y.; Wang, P.; Gao, R.X. DCNN-based multi-signal induction motor fault diagnosis. *IEEE Trans. Instrum. Meas.* **2019**, *69*, 2658–2669. [CrossRef]
13. Liu, H.; Zhou, J.; Zheng, Y.; Jiang, W.; Zhang, Y. Fault diagnosis of rolling bearings with recurrent neural network-based autoencoders. *ISA Trans.* **2018**, *77*, 167–178. [CrossRef] [PubMed]
14. Mey, O.; Neufeld, D. Explainable AI Algorithms for Vibration Data-Based Fault Detection: Use Case-Adadpted Methods and Critical Evaluation. *Sensors* **2022**, *22*, 9037. [CrossRef] [PubMed]
15. Xiao, D.; Huang, Y.; Qin, C.; Shi, H.; Li, Y. Fault diagnosis of induction motors using recurrence quantification analysis and LSTM with weighted BN. *Shock Vib.* **2019**, *2019*, 8325218. [CrossRef]
16. Tripicchio, P.; D'Avella, S. Is deep learning ready to satisfy industry needs? *Procedia Manuf.* **2020**, *51*, 1192–1199. [CrossRef]
17. Singh, M.; Shaik, A.G. Faulty bearing detection, classification and location in a three-phase induction motor based on Stockwell transform and support vector machine. *Measurement* **2019**, *131*, 524–533. [CrossRef]
18. Saberi, A.N.; Sandirasegaram, S.; Belahcen, A.; Vaimann, T.; Sobra, J. Multi-sensor fault diagnosis of induction motors using random forests and support vector machine. In Proceedings of the 2020 International Conference on Electrical Machines (ICEM), Gothenburg, Sweden, 23–26 August 2020; IEEE: Piscataway, NJ, USA, 2020; Volume 1, pp. 1404–1410.
19. Glowacz, A.; Glowacz, W.; Glowacz, Z.; Kozik, J.; Gutten, M.H.; Korenciak, D.; Carletti, E. Fault diagnosis of three phase induction motor using current signal, MSAF-Ratio15 and selected classifiers. *Arch. Metall. Mater.* **2017**, *62*, 2413–2419. [CrossRef]
20. Asr, M.Y.; Ettefagh, M.M.; Hassannejad, R.; Razavi, S.N. Diagnosis of combined faults in Rotary Machinery by Non-Naive Bayesian approach. *Mech. Syst. Signal Process.* **2017**, *85*, 56–70. [CrossRef]
21. Tahir, M.M.; Hussain, A.; Badshah, S.; Khan, A.Q.; Iqbal, N. Classification of unbalance and misalignment faults in rotor using multi-axis time domain features. In Proceedings of the 2016 International Conference on Emerging Technologies (ICET), Islamabad, Pakistan, 18–19 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 1–4.
22. Gangsar, P.; Pandey, R.K.; Chouksey, M. Unbalance detection in rotating machinery based on support vector machine using time and frequency domain vibration features. *Noise Vib. Worldw.* **2021**, *52*, 75–85. [CrossRef]
23. Debie, E.; Shafi, K. Implications of the curse of dimensionality for supervised learning classifier systems: Theoretical and empirical analyses. *Pattern Anal. Appl.* **2019**, *22*, 519–536. [CrossRef]

24. Ferrucho-Alvarez, E.R.; Martinez-Herrera, A.L.; Cabal-Yepez, E.; Rodriguez-Donate, C.; Lopez-Ramirez, M.; Mata-Chavez, R.I. Broken rotor bar detection in induction motors through contrast estimation. *Sensors* **2021**, *21*, 7446. [CrossRef] [PubMed]

25. Lizarraga-Morales, R.A.; Rodriguez-Donate, C.; Cabal-Yepez, E.; Lopez-Ramirez, M.; Ledesma-Carrillo, L.M.; Ferrucho-Alvarez, E.R. Novel FPGA-based methodology for early broken rotor bar detection and classification through homogeneity estimation. *IEEE Trans. Instrum. Meas.* **2017**, *66*, 1760–1769. [CrossRef]

26. Unser, M. Texture classification and segmentation using wavelet frames. *IEEE Trans. Image Process.* **1995**, *4*, 1549–1560. [CrossRef] [PubMed]

27. Zhang, Y. Support vector machine classification algorithm and its application. In *Proceedings of the Information Computing and Applications: Third International Conference, ICICA 2012, Chengde, China, 14–16 September 2012*; Proceedings, Part II 3; Springer: Berlin/Heidelberg, Germany, 2012; pp. 179–186.

28. Benbouzid, M.E.H.; Vieira, M.; Theys, C. Induction motors' faults detection and localization using stator current advanced signal processing techniques. *IEEE Trans. Power Electron.* **1999**, *14*, 14–22. [CrossRef]

29. Haralick, R.M.; Shanmugam, K.; Dinstein, I.H. Textural features for image classification. *IEEE Trans. Syst. Man Cybern.* **1973**, *6*, 610–621. [CrossRef]

30. Steinwart, I.; Christmann, A. *Support Vector Machines*; Springer: Berlin/Heidelberg, Germany, 2008.

31. Gieseke, F.; Airola, A.; Pahikkala, T.; Kramer, O. Fast and simple gradient-based optimization for semi-supervised support vector machines. *Neurocomputing* **2014**, *123*, 23–32. [CrossRef]

32. Abe, S. Training of support vector machines with Mahalanobis kernels. In Proceedings of the Artificial Neural Networks: Formal Models and Their Applications—ICANN 2005: 15th International Conference, Warsaw, Poland, 11–15 September 2005; Proceedings, Part II 15; Springer: Berlin/Heidelberg, Germany, 2005; pp. 571–576.

33. Ben-Hur, A.; Weston, J. A user's guide to support vector machines. In *Data Mining Techniques for the Life Sciences*; Humana Press: Totowa, NJ, USA, 2010; pp. 223–239.

34. Mey, O.; Neudeck, W.; Schneider, A.; Enge-Rosenblatt, O. Machine learning-based unbalance detection of a rotating shaft using vibration data. In Proceedings of the 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vienna, Austria, 8–11 September 2020; IEEE: Piscataway, NJ, USA, 2020; Volume 1, pp. 1610–1617.

35. Guo, S.; Yang, T.; Gao, W.; Zhang, C. A novel fault diagnosis method for rotating machinery based on a convolutional neural network. *Sensors* **2018**, *18*, 1429. [CrossRef]

36. Yongbo, L.I.; Xiaoqiang, D.U.; Fangyi, W.A.N.; Xianzhi, W.A.N.G.; Huangchao, Y.U. Rotating machinery fault diagnosis based on convolutional neural network and infrared thermal imaging. *Chin. J. Aeronaut.* **2020**, *33*, 427–438.