

Wind Turbine Fault Detection and Classification by Means of Image Texture Analysis

Magda Ruiz^{a,*}, Luis E. Mujica^a, Santiago Alférez^a, Leonardo Acho^a, Christian Tutivén^a, Yolanda Vidal^a, José Rodellar^a, Francesc Pozo^a

^a*Control Dynamics and Applications (CoDALab), Departament de Matemàtiques, Escola d'Enginyeria de Barcelona Est (EEBE), Universitat Politècnica de Catalunya (UPC), Campus Diagonal-Besòs (CDB), Eduard Maristany, 6-12, Sant Adrià de Besòs (Barcelona) 08930, Spain*

Abstract

The future of the wind energy industry passes through the use of larger and more flexible wind turbines in remote locations, which are increasingly offshore to benefit stronger and more uniform wind conditions. The cost of operation and maintenance of offshore wind turbines is approximately 15-35% of the total cost. Of this, 80% goes towards unplanned maintenance issues due to different faults in the wind turbine components. Thus, an auspicious way to contribute to the increasing demands and challenges is by applying low-cost advanced fault detection schemes. This work proposes a new method for detection and classification of wind turbine actuators and sensors faults in variable-speed wind turbines. For this purpose, time domain signals acquired from the operating wind turbine are represented as two-dimensional matrices to obtain grayscale digital images. Then, the image pattern recognition is processed getting texture features under a multichannel representation. In this work, four types of texture characteristics are used: statistical, wavelet, granulometric and Gabor features. Next, the most significant ones are selected using the conditional mutual criterion. Finally, the faults are detected and distinguished between them (classified) using an automatic classification tool. In particular, a 10-fold cross-validation is used to obtain a more generalized model and evaluates the classification performance. Coupled non-linear aero-hydro-servo-elastic simulations of a 5MW offshore type wind turbine are carried out in several fault scenarios. The results show a promising methodology able to detect and classify the most common wind turbine faults.

Keywords: fault detection, fault classification, wind turbine, texture analysis.

2000 MSC: [2010] 00A69

1. Introduction

Offshore wind farms promise to become an important source of energy in the near future. However, their operation and maintenance is more difficult and expensive than equivalent onshore wind farms. The current reliability and failure modes of commercial offshore wind turbines are such that a no-maintenance strategy is not a viable option. Improved preventive and corrective maintenance schemes will become crucial for economic exploitation of offshore wind power. Thus, a promising way to contribute is by applying low-cost advanced fault detection schemes. In this regard, this work proposes a new fault detection and classification method that uses on-line SCADA (supervisory control and data acquisition) data already available at a megawatt-sized industrial wind turbine.

On one hand, several methods for fault detection have been proposed for wind turbines. These methods can be mainly classified according to their procedure. Namely, model based and data based. This work uses the data based approach, as in recent years these techniques seem to have received more attention in academia and industry. For example, Bessa et al. [1] propose a new data-driven fault detection and isolation scheme based on time series and data analysis without using any kind of physical modeling; Gong et al. [2] propose a mechanical fault detection

*Corresponding author.

Email address: magda.ruiz@upc.edu (Magda Ruiz)

URL: <https://codalab.upc.edu/en> (Magda Ruiz)

algorithm by using only nonstationary generator stator current measurements; Teng et al. [3] propose a multi-fault detection method using vibration signals originated from a real multi-fault wind turbine gearbox with catastrophic failure. In all the aforementioned papers, the fundamental part is the signal processing method. Wherein, the goal in signal processing is to highlight the signal behavior. **References [4] and [5] give a detailed survey on wind turbine condition monitoring and fault diagnosis. Also, an excellent review on wind turbine condition monitoring methods is given in [6] subdividing them into subsystem techniques and overall system techniques as well as destructive and non-destructive techniques.**

On the other hand, recent engineering applications have begun to use digital images for fault detection. For instance, Shahriar et al. [7] propose fault diagnosis of induction motors using digital grayscale images. In this sense, we have been working in fault detection of wind turbines based on data [8] and in the classification of abnormal cells images using texture features [9]. Thus, merging these two areas of knowledge; in this work, a methodology for wind turbine fault detection and classification, based on digital image processing, is proposed. **The most common fault detection techniques based on signals focus on the frequencies and magnitude of the signal, which are dealing with a one dimensional domain. In this paper, however, by translating the signal into an image (two dimensions), other local features are explored. Another reason to use image texture analysis is that wind turbines are noisy environments, and, thus, noise is added in the recorded signals. However, when the data is translated into images, the added noise is considered as the illumination of the light to the image [10]. Hence, the effect of noise to the signal is removed. The results achieved by the proposed approach, with high classification accuracies, clearly demonstrate the potential of exploiting the feature design from texture analysis.**

It is well known that a learning model tends to overfit when using a large number of features. To address this problem, feature extraction and feature selection techniques are widely employed for dimensionality reduction in classification problems [11, 12, 13]. On one hand, feature extraction maps the original feature space to a new one with lower dimensions by combining the original feature space. On the other hand, feature selection selects a subset of features from the original set without any transformation. For the classification problem proposed in this work, both, feature extraction and selection are used with the aim to select a subset of highly discriminant features.

In this paper, the benchmark model for large-scale wind turbine fault detection and accommodation proposed in [14] is used. The aim of the benchmark model is to provide a common ground to test and compare different methods considering the most common faults in megawatt-sized wind turbines. Once the time domain signals are stored, they are represented by grayscale images. Next, a set of texture features is extracted from each image. In this process, it is computed the visual patterns that composes an image, which can be regarded as a similarity grouping in an image. Because of the high computational cost associated with the use of all the texture features, the number of features must be reduced. This is achieved by giving priority to the most relevant and less redundant texture features. Finally, a process of training and validation of the classification technique is developed by using a 10-fold cross-validation. Thus, the methodology for fault diagnosis is completed and calibrated. Regarding novelty and contribution, three crucial issues are highlighted:

1. the use of texture features for fault diagnosis in large-scale wind turbines;
2. the integration of different techniques used in other engineering fields to solve a latent problem;
3. the acquisition of new features based on Gabor Filters.

The rest of the paper is organized as follows. The benchmark model for wind turbines is explained in Section 2. Then, the overall design of the methodology, including the stages that comprise it, are reported in Section 3. Likewise, since the features based on Gabor filters are a contribution of this paper, and it is the first time they are applied to fault detection in mechanical systems, a detailed explanation of its realization is included in the same section. Next, results are presented and discussed in Section 4. Finally, conclusions are drawn in Section 5.

2. Benchmark model

2.1. Reference 5MW wind turbine

The benchmark model for fault detection and fault tolerant control of wind turbines proposed in [14] is used in this work. The benchmark is based on the model of a generic three blade horizontal variable speed wind turbine with a full

converter coupling and a rated power of 5MW. The aim is to provide a common ground to test and compare different methods for fault detection and fault tolerant control of wind turbines. This benchmark is based on FAST and thus proposes a higher-fidelity model of the WT, see [15]. FAST is the National Renewable Energy Laboratory's (NREL) primary computed aided engineering tool for simulating the coupled dynamic response of wind turbines. FAST joins aerodynamics models, hydrodynamics models for offshore structures, control and electrical system (servo) dynamics models, and structural (elastic) dynamics models to enable coupled nonlinear aero-hydro-servo-elastic simulation in the time domain. Germanischer Lloyd issued a certificate of evaluation for FAST in 2005. Nowadays, the FAST software has been widely accepted by the scientific community and is used in the development of control systems for wind turbines, e.g. [16, 17, 18]. In this work, numerical simulations with FAST are performed with the NREL 5MW on-shore wind turbine. The wind turbine characteristics are summarized in Table 1, see [19].

Table 1: Gross properties of the wind turbine

Rated power (P_n)	5MW
Number of blades	3
Height of tower	87.6m
Rotor diameter	126m
Cut in, Rated, Cut-Out Wind Speed	3m/s, 11.4m/s, 25m/s
Gearbox ratio	98
Nominal generator speed ($\omega_{g,n}$)	1173.7 rpm

Hereafter, the wind model, the generator-converter actuator model, the pitch actuator model and the baseline control strategy are recalled in order to introduce the notation and the concepts employed in following sections.

2.2. Wind modeling

In the simulations, new wind data sets are generated in order to capture a more realistic turbulent wind simulation and, thus, to test the proposed methodology in a more realistic scenario. The turbulent-wind simulator TurbSim [20] developed by NREL is used and full TurbSim wind fields are employed in the simulations. The generated wind data has the following characteristics: Kaimal turbulence model with intensity set to 10%, logarithmic profile wind type, mean speed is set to 18.2 m/s and simulated at hub height, and the roughness factor is set to 0.01 m. In this work, each simulation is ran with a different wind data set. In particular, 120 different wind data sets, of a duration of 600 seconds each, were used.

2.3. Generator-converter actuator model

The generator torque is controlled by a reference signal $\hat{\tau}_{g,r}$. The dynamics of the generator-converter is approximated by a first-order differential system as [21]

$$\dot{\tau}_g(t) + \alpha_{gc}\tau_g(t) = \alpha_{gc}\hat{\tau}_{g,r}(t), \quad (1)$$

where τ_g is the generator torque and α_{gc} is the generator and converter model parameter (in the simulations $\alpha_{gc} = 50$ [19]). The produced power by the generator is given by

$$P_e(t) = \eta_g \omega_g(t) \tau_g(t), \quad (2)$$

where η_g is the efficiency of the generator (in the simulations $\eta_g = 0.98$) and ω_g is the generator speed measurement.

2.4. Pitch actuator model

The hydraulic pitch system can be modeled by a second order system, see [14], with reference angle $\hat{\beta}_c$ and actual pitch angle β_i as

$$\ddot{\beta}_i(t) + 2\zeta\omega_n\dot{\beta}_i(t) + \omega_n^2\beta_i(t) = \omega_n^2\hat{\beta}_c(t), \quad (3)$$

where ζ is the damping factor and ω_n is the natural frequency. This differential equation is associated to the pitch control system of every blade ($i = 1, 2, 3$). For the healthy case, the parameters $\zeta = 0.6$ and $\omega_n = 11.11$ rad/s are used [14].

2.5. Baseline controllers

The 5MW reference wind turbine given by FAST contains torque and pitch controllers for the full load region, see [19]. In this section we recall these controllers and refer to them as the baseline torque and pitch controllers, as they are used in the literature for comparison purposes, e.g. [16, 17, 22, 23, 24].

The torque and pitch controllers use the generator speed measurement as input. To mitigate high-frequency excitation of the control system, we filtered the generator speed measurement, using a recursive, single-pole low-pass filter with exponential smoothing as proposed by [19].

In the full load region, the torque controller maintains regulated the generated power, thus the generator torque is proposed inversely proportional to the filtered generator speed [15], or,

$$\tau_{g,r}(t) = \frac{P_{\text{ref}}(t)}{\hat{\omega}_g(t)}, \quad (4)$$

where P_{ref} is the reference power (normally the nominal value is used), and $\hat{\omega}_g$ is the filtered generator speed. This controller will be referred as the baseline torque controller. Due to physical limitations of the actuator, the torque controller is saturated to a maximum of 47402.9Nm, and rate limited to a maximum of 15000Nm/s [19].

A collective pitch controller is added to the generator speed tracking error, in order to work collaboratively with the torque controller in the full load region. The collective blade gain scheduling PI-controller (GSPI) is one of the first well-documented controllers and it is used in the literature as a baseline controller to compare the obtained results. This GSPI control is a collective pitch controller that compensates the nonlinearities in the turbine by changing the controller gains according to a scheduling parameter. This controller was originally developed by Jonkman for the standard land-based 5MW turbine, see [19]. The GSPI control has the generator speed as input and the pitch servo set-point, β_c , as output. That is,

$$\beta_c(t) = K_p(\gamma)(\hat{\omega}_g(t) - \omega_{g,r}(t)) + K_i(\gamma) \int_0^t (\hat{\omega}_g(t) - \omega_{g,r}(t)) d\tau, \quad i = 1, 2, 3, \quad (5)$$

where $\omega_{g,r}$ is the reference generator speed (usually the nominal value is used) and the scheduling parameter γ is obtained by averaging the measurements of all pitch angles.

The scheduled gains are calculated following [19]. Finally, to not exceed the mechanical limitations of the pitch actuator, the input signal β_c is saturated to a maximum of 90° and rate limited to 8°/s [19].

2.6. Available SCADA-data

In this work, the proposed fault detection and classification method is SCADA-data based. That is, it uses data already collected in an industrial wind turbine. In particular, Table 2 presents assumed available data on a MW-scale commercial wind turbine that is used in this work by the fault detection and classification method, see [14]. Throughout the paper, these available measurements will also be referred to as variables.

Table 2: Assumed available measurements: sensors that are available on a MW-scale commercial wind turbine.

Number	Sensor Type	Symbol	Units
1	Generated electrical power	$P_{e,m}$	kW
2	Rotor speed	$\omega_{r,m}$	rad/s
3	Generator speed	$\omega_{g,m}$	rad/s
4	Generator torque	$\tau_{c,m}$	Nm
5	First pitch angle	$\beta_{1,m}$	deg
6	Second pitch angle	$\beta_{2,m}$	deg
7	Third pitch angle	$\beta_{3,m}$	deg
8	Fore-aft acceleration at tower bottom	$a_{fa,m}^b$	m/s ²
9	Side-to-side acceleration at tower bottom	$a_{ss,m}^b$	m/s ²
10	Fore-aft acceleration at mid-tower	$a_{fa,m}^m$	m/s ²
11	Side-to-side acceleration at mid-tower	$a_{ss,m}^m$	m/s ²
12	Fore-aft acceleration at tower top	$a_{fa,m}^t$	m/s ²
13	Side-to-side acceleration at tower top	$a_{ss,m}^t$	m/s ²

2.7. Fault scenarios

In this paper, different faults proposed in the fault tolerant control benchmark [25] will be considered, as given in Table 3. These faults selected by the benchmark cover distinct parts of the wind turbine, different fault types and classes, and various levels of severity.

Table 3: Fault scenarios.

Fault	Type	Description
1	Pitch actuator	Change in dynamics: high air content in oil ($\omega_n = 5.73$ rad/s, $\xi = 0.45$)
2	Pitch actuator	Change in dynamics: pump wear ($\omega_n = 7.27$ rad/s, $\xi = 0.75$)
3	Pitch actuator	Change in dynamics: hydraulic leakage ($\omega_n = 3.42$ rad/s, $\xi = 0.9$)
4	Generator speed sensor	Scaling (gain factor equal to 1.2)
5	Pitch angle sensor	Stuck (fixed value equal to 5 deg)
6	Pitch angle sensor	Stuck (fixed value equal to 10 deg)
7	Torque actuator	Offset (offset value equal to 2000 Nm)

In particular, faults in the pitch actuator are studied in the hydraulic system, which result in changed dynamics due to either a high air content in oil (fault 1) or a drop in pressure due to pump wear (fault 2) or hydraulic leakage (fault 3) [26]. High air content in the oil (fault 1) is an incipient reversible process, which means that the air content in the oil may disappear without any necessary repair to the system. The nominal value of the air content in the oil is 7%, whereas the high air content in the oil corresponds to 15%. The effect of such a fault is expressed by the new natural frequency $\omega_n = 5.73$ rad/s and the damping ratio of $\xi = 0.45$. Pump wear (fault 2) is an irreversible slow process over the years that results in low pump pressure. As this wear is irreversible, the only possibility to fix it is to replace the pump, which will happen after pump wear reaches certain level. Meanwhile, the pump will still be operating and the system dynamics is slowly changing, while the turbine structure should be able to withstand the effects of this fault. Pump wear after approximately 20 years of operation might result in pressure reduction to 75% of the rated pressure, which is reflected by the abnormal natural frequency $\omega_n = 7.27$ rad/s and a fault damping ratio of $\xi = 0.75$. Hydraulic leakage (fault 3) is another irreversible incipient fault but is introduced faster than the pump wear. This fault condition reduces the dynamics of the pitch system, and for a reduction of 50% of the nominal pressure the natural frequency is reduced to $\omega_n = 3.42$ rad/s and the corresponding damping ratio is $\xi = 0.9$.

The gain factor fault (fault 4) is introduced when the encoder, which measures the generator speed, reads more marks on the rotating part than actually present. This happens as a result of dirt or other false markings on the rotating part.

Faults in the pitch position measurement (pitch position sensor fault) are also considered. This is one of the most important failure modes found on actual systems [27, 28]. The origin of these faults is either electrical or mechanical, and it can result in a fixed value (faults 5 and 6) on the measurements. This fixed value fault should be easy to detect, and, therefore, it is important that a fault detection, classification, and accommodation scheme is able to deal with it. If not handled correctly, these faults will influence the pitch reference position because the baseline pitch controller is based on these pitch position measurements.

Finally, a converter torque offset fault is considered (fault 7). This fault can originate from either an internal fault in the converter electronics or an offset in the converter torque estimate which itself can be due to design/manufacturing defects. This type of fault can disturb the torque control action with high severity.

3. Design of the fault detection and classification methodology

In this section, a new methodology for wind turbine fault detection and classification is proposed. A total of 140 simulations for the healthy case (normal operation), and 40 simulations for each one of the studied faults (Table 3) are conducted using the benchmark model previously described. In each simulation, 13 variables are measured (Table 2) and each one is converted into a digital image (Section 3.1). Consequently, each simulation is defined by 13 digital images. From each image, initially, 65 texture features are extracted. Therefore, a big amount of features defines each case (Section 3.2), which should lead to a high computational cost. For that reason, the criterion of Maximum

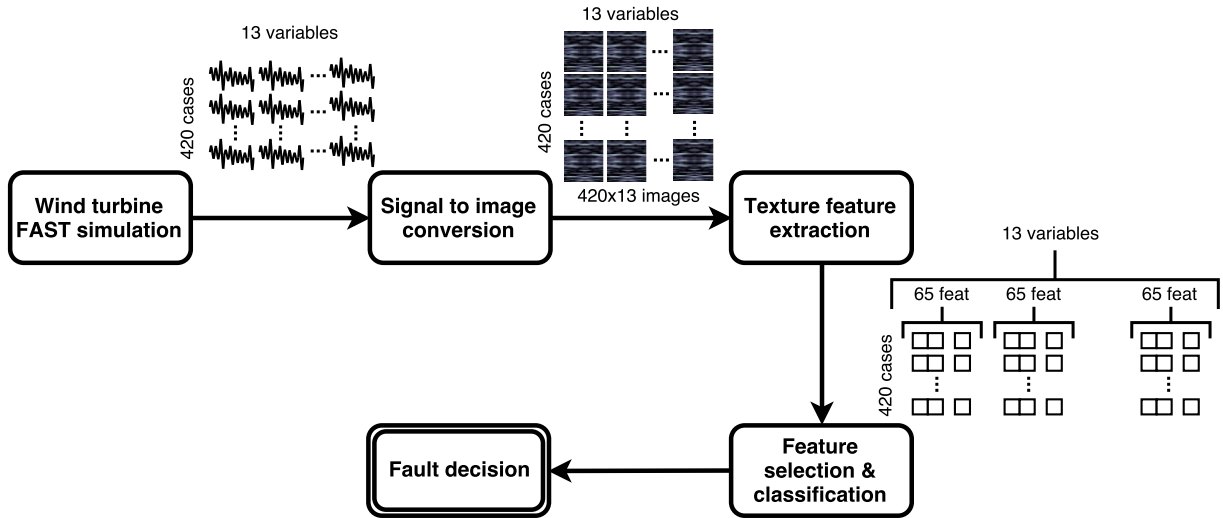


Figure 1: Proposed methodology for wind turbine fault detection and classification.

Relevance Minimum Redundancy (MRMR) is used to select the most significant features (Section 3.3.1). Finally, several classifiers are implemented to select the best one to detect and classify the studied faults (Section 3.3.2).

The process of feature selection and fault classification are conducted by means of a stratified cross validation, to indicate when further training is not resulting in better generalization. This means that, for each iteration of the validation, one portion of the complete set of cases are randomly chosen to define the most significant texture features and to train the classification tool. The remaining cases are used to test the classifier, by using the texture features previously selected, and the configuration of the classification tool previously defined. A general scheme of the methodology is presented in Figure 1. The next subsections explain in detail the overall steps of the proposed methodology.

3.1. Signal to image conversion

The fault diagnosis method first converts time-domain signals, from the 13 measured variables, into 2D gray-level images to exploit texture information from the converted images. The data conversion process is carried out as proposed in reference [7]. In particular, in this work, the image size is 128×128 pixels, and it is constructed as follows: the first 128 data-points determine the first row of the grayscale image; immediately after, the next 128 data-points determine the second row and so on (as shown in Figure 2). Considering that the sampling time is 0.0125s, each image contains approximately 204s of simulation, which ensures to capture all the system dynamics. An example of an acceleration measurement and its representation as an image is shown in Figure 3.

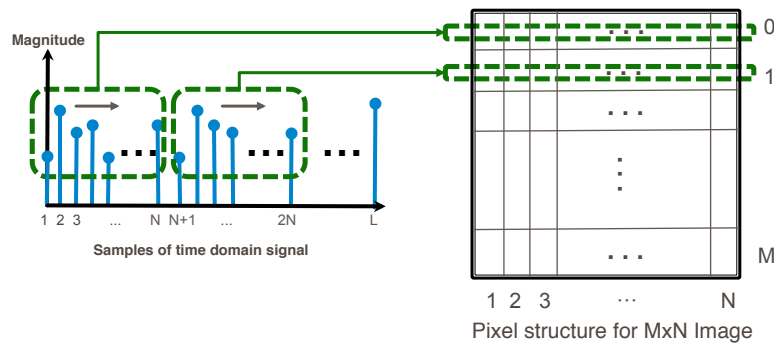


Figure 2: Signal-to-image conversion scheme as in Shahriar et al. [7].

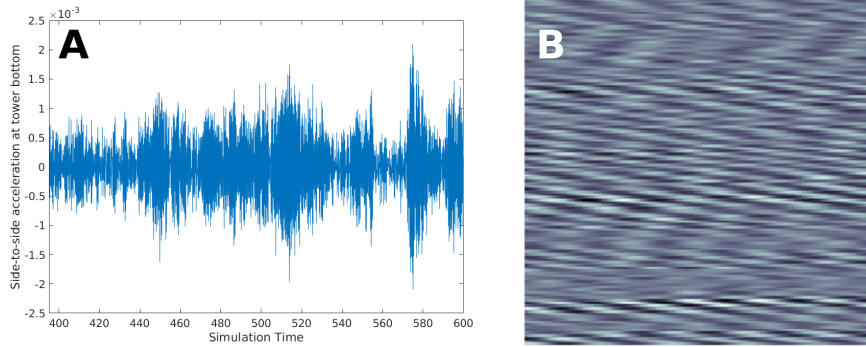


Figure 3: Example of a signal-to-image conversion. A: original signal (side-to-side acceleration at mid-tower); B: image obtained after the signal-to-image conversion.

175 Since 140 simulations are conducted for healthy condition and 40 simulations for each faulty scenario (7 faults), $140 + 40 \times 7 = 420$ cases are simulated with 13 measurements (variables) each one. Finally, 420×13 images are available. Figure 4 shows an example of the images obtained for each scenario (Normal and 7 faults) and for all 13 measured variables. A simple eye recognition of differences between faults is not possible.

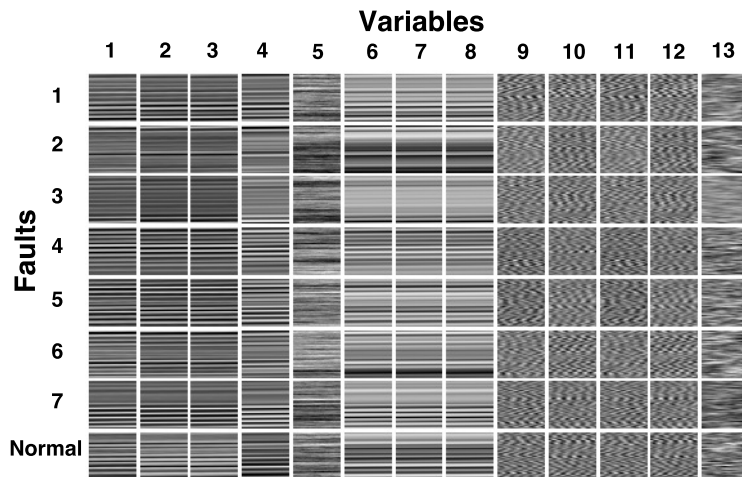


Figure 4: Examples for each of the eight scenarios (normal and 7 faults). For each scenario, 13 variables are represented as an image.

3.2. Texture feature extraction

180 The idea to use texture features in image processing comes from the innate human ability to recognize textural differences using vision and touch. The texture in a digital image is defined by its uniformity, density, thickness, roughness, regularity, intensity, directionality, pixel tone and their spatial relationships, among others. The main goal of the feature extraction is to get quantitative measures in order to identify different texture patterns. Several procedures to analyze the texture of a digital image can be found in the literature [29]. In this work, four types of
 185 texture features are used: statistical, wavelet, granulometric and Gabor. Figure 5 shows a representative scheme of the extracted texture features. The next subsections recall the used texture features.

3.2.1. Statistical features

190 These can be divided into first and second-order statistical features. The first order features are based on the image histogram. The histogram shows the number of pixels $H(f_i)$ having specific pixel intensity value f_i (frequencies) [30]. From the statistical information that the histogram provides, six features are used in this work: mean, standard

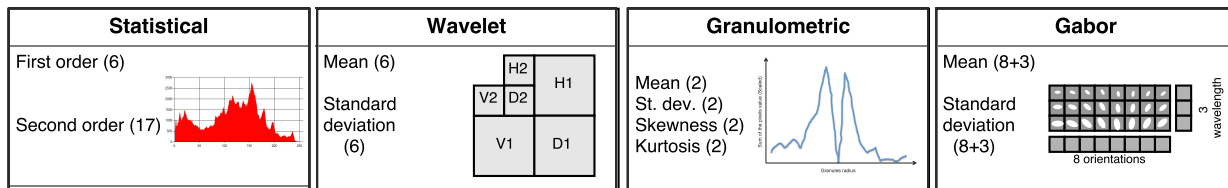


Figure 5: Scheme of the texture features. The parenthesis indicate the number of textures extracted in each case.

deviation, skewness, kurtosis, energy and entropy. The mean is the average of the whole intensity values of the image; the standard deviation measures the dispersion of the intensity values around the mean; the skewness describes the symmetry of the histogram around the mean; the kurtosis measures the flatness of the histogram; the energy represents the uniformity of the histogram; and the entropy describes the variability of the histogram [31].

The second-order statistical features provide information about the texture and they are based on the grey level co-occurrence matrix. This co-occurrence matrix considers the information about the intensity values and the position of the pixels with similar intensities. As many as 15 second-order features proposed in [32] are extracted here: uniformity, contrast, homogeneity, correlation, the sum average, sum variance, sum entropy, entropy, difference variance, difference entropy, information measures of correlation 1 and 2, maximal correlation coefficient and maximum probability. In addition, two more second-order features proposed in [33] are also included in this study: cluster shade and cluster prominence. In summary, 23 statistical features are implemented: 6 first order and 17 second order (see Figure 5).

3.2.2. Wavelet features

The main advantage of the wavelet transform is the decomposition and reconstruction of signals in trends (lower resolution) and fluctuations (local changes). Analyzing the fluctuations, the texture of the image can be characterized. The algorithm implemented for the decomposition is the Quadrature Mirror Filters (QMF) based on the Discrete Wavelet Transformation (DWT). This algorithm computes the wavelet coefficients using a low-pass wavelet filter and a high-pass wavelet filter [34]. Thus, first-level image decomposition and reconstruction is performed where the original image (Figure 6A) is decomposed into 4 sub-images as shown in Figure 6B. As a result, an approximation of the image (A1) and three highlighted versions of the horizontal, vertical and diagonal details (H1, V1 and D1 respectively). This process is repeated in a second-level image decomposition and reconstruction over the above first approximation image (A1) to obtain another level of the approximation (A2) and three more detailed sub-images (H2, V2, and D2) as shown in Figure 6C. On these 6 detail sub-images (H1, H2, V1, V2, D1 and D2), the mean and standard deviation are calculated to obtain 2×6 wavelet features [35].

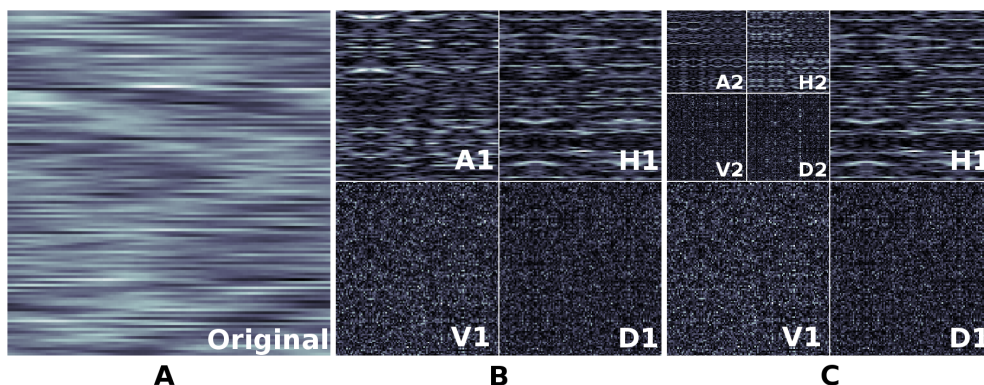


Figure 6: Example of the discrete wavelet transform applied on an image corresponding to the side-to-side acceleration at tower bottom (m/s²). A: original image; B: first level wavelet decomposition; C: second level wavelet decomposition.

215 **3.2.3. Granulometric features**

The main objective of granulometric features is to simplify the image eliminating unwanted objects. To do so, the image is filtered through mathematical morphology operations. In this way, the characteristics of portions of the images (called objects) are identified. The mathematical morphology compares one object from the image, of which no prior information is available, with a known object (generally it is a circle). Thus, the image can be characterized in a topological and geometric continuous space, for instance: shape, size, convexity, connectivity and geodesic distance. The first two characteristics (shape and size) are well known. The term convexity determines the pixels that are or are not included within the object of an image; the connectivity is the measure of the relationship between a pixel and its neighbors; geodesic distance is the minimum distance between two pixels [36].

225 These characteristics are determined from two curves: the granulometric curve and the pseudo-granulometric curve (Figure 7). Both curves represent the size distribution of the bright and dark granules on the image. The pseudo-granulometric curve is calculated by successive applications of the elementary morphological operations: dilation and erosion. In dilation operation, the image becomes lighter and dark details are reduced. On the other side, in erosion operation, the image becomes darker and light details are reduced. The granulometric curve is calculated by successive applications of mathematical morphological operations: opening and closing. In opening operation light details are reduced, while in closing operation dark details are reduced.

230 After, the mean, standard deviation, skewness and kurtosis are calculated for each curve [31, 37]. These eight parameters constitute the granulometric features.

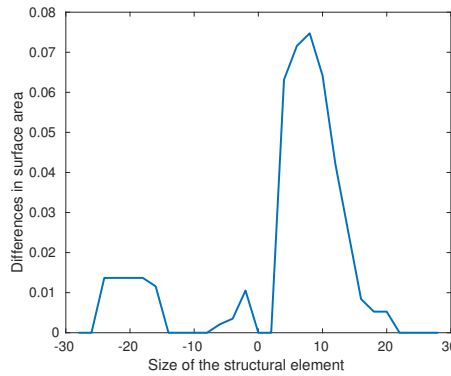


Figure 7: Pseudo-granulometric curve for variable 13 on the first row in Figure 4.

3.2.4. Gabor features

235 A Gabor feature $r(x, y)$ is obtained by the convolution between an image $I(x, y)$ and a 2D Gabor function $g(x, y)$, where $(x, y) \in \Omega$, as follows:

$$r(x, y) = \iint_{\Omega} I(\xi, \eta) g(x - \xi, y - \eta) d\xi d\eta. \quad (6)$$

In this paper, the following family of Gabor functions (filters) are used [38]:

$$g_{\lambda, \Theta}(x, y) = \exp\left(-\frac{x'^2 + \gamma y'^2}{2\sigma^2}\right) \exp\left(i\left(2\pi\frac{x'}{\lambda} + \varphi\right)\right), \quad (7)$$

where, $x' = x \cos \Theta + y \sin \Theta$, $y' = -x \sin \Theta + y \cos \Theta$, $\sigma = 0.56\lambda$, $\gamma = 0.5$ and $\varphi = 0$. In Figure 8 it can be seen the magnitude of four Gabor filters for a combination of two wavelengths ($\lambda = \{8, 15\}$) and two orientations ($\Theta = \{45^\circ, 90^\circ\}$).

240 Then, Gabor features are obtained from different versions of the image processed by Gabor filters. In this work, a bank of 24 filters corresponding to three wavelengths ($\lambda = \{8, 12, 15\}$) and eight orientations ($\Theta = \{22.5^\circ, 45^\circ, 67.5^\circ, 90^\circ, 112.5^\circ, 135^\circ, 157.5^\circ, 180^\circ\}$). With this filter bank, three rotation-invariant Gabor responses (one by each λ) are

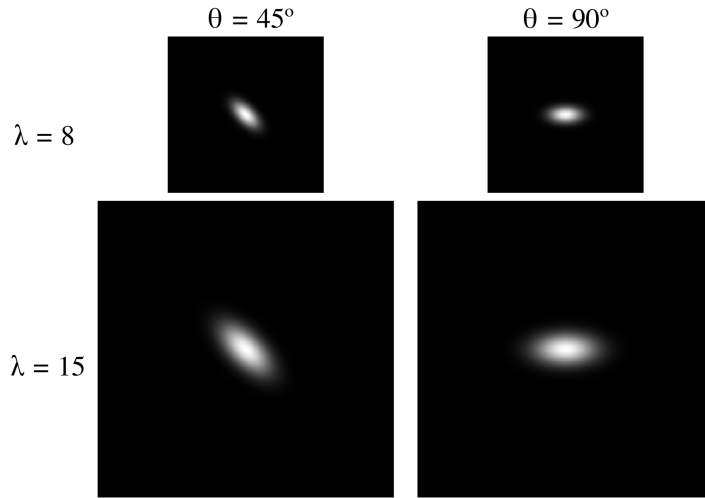


Figure 8: Magnitudes of a Gabor filter bank belonging to $\lambda = \{8, 15\}$ and $\Theta = \{45^\circ, 90^\circ\}$

obtained by summing all the filtered images of the eighth orientations (the rightmost column in Figure 9). In the same way, eight scale-invariant Gabor responses (one by each Θ) are obtained by summing all the filtered images of the three wavelengths (the bottom row of Figure 9) [39]. Therefore, a total of 11 Gabor image responses are used. For each combined response, the mean and the standard deviation are calculated, in this way $(3 + 8) \times 2$ Gabor features are extracted by each measurement.

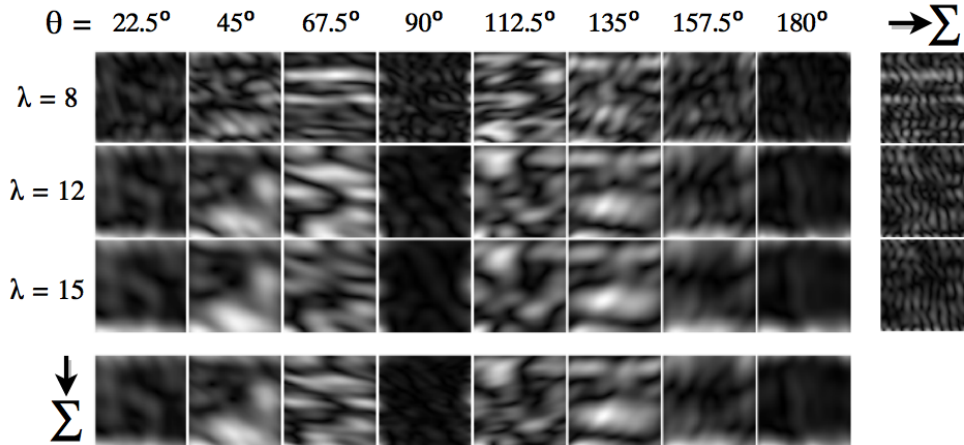


Figure 9: Gabor responses for an image from the side-to-side acceleration at mid-tower.

In summary, 65 texture features are extracted from each image: 23 statistical, 12 wavelet, 8 granulometric and 22 Gabor features. Since 13 variables are measured (13 images), a total of 845 features describe each simulation.

3.3. Feature selection and classification algorithm

As previously mentioned, a more generalized and improved methodology is obtained if a stratified cross-validation is used. Therefore in this work, the feature selection process and the configuration of the best classification tool are conducted by a 10-fold cross-validation. Firstly, ten portions are randomly chosen (keeping the proportion between classes). Next, in each iteration, 9/10 portions are used as training set. Finally, the remaining portion (a tenth) is used to validate the classification achievement using the texture features previously selected, and the configuration of

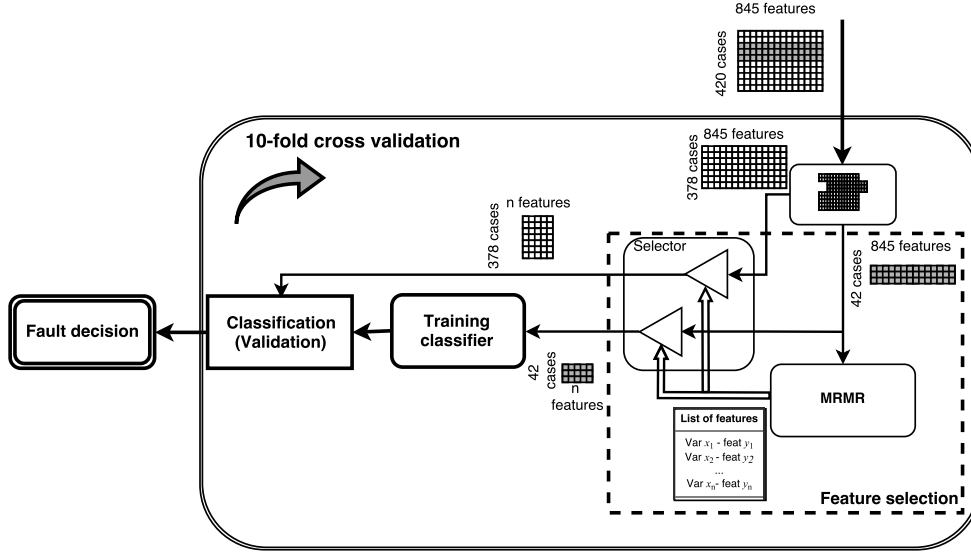


Figure 10: Feature selection and classification: 10-fold cross-validation.

the classification tool previously defined. Thus, results are independent of the dataset. The proposed methodology is summarized in Figure 10, and it is explained in detail in the next sections.

3.3.1. Feature selection

Since 845 features are obtained, the complexity of the problem description makes difficult to build a good classification system. Feature selection defines a topic commonly used in machine learning to select the most significant features. Feature selection allows to improve the classification performance making faster and more profitable the classifiers. Besides, the data processing is better understood and, at the same time, the data dimensionality is reduced.

In this work, a well known technique in the area of recognition of genes and phenotypes is used. The feature selection technique is based on the classification accuracy paradigm using the Maximum Relevance Minimum Redundancy (MRMR) criterion [40], which calculates the conditioned entropy of the features over the presence of a class level. In this context, MRMR is defined by the following criterion:

$$J_{MRMR}(X_k) = I(X_k; Y) - I(X_k; S), \quad (8)$$

where the Relevance $[I(X_k; Y)]$ is the mutual information between the unselected feature (X_k) and the class label (Y) and the Redundancy $[I(X_k; S)]$ is the mutual information between the unselected feature (X_k) and the selected features (S) [41]. Thereby, the features are sorted from the most relevant (and less redundant) to the less relevant (and most redundant) evaluating the inclusion of a specific feature in a set delimited by the conditional likelihood of the class label (Figure 10).

3.3.2. Classification algorithm

Each tested classification algorithm is trained using the 9/10 portions used to select the most relevant features. If required, it is adjusted and finally it is validated using the remaining 1/10 portion with the same features previously selected.

Five classification algorithms are compared, namely, k-Nearest Neighbors model classification (k-NN), Linear Discriminant model classification (LDA), Decision trees model classification (Tree), Ensemble model classification Bag-tree (Bag tree) and Linear Support Vector Machine (LSVM) model classification with linear Kernel.

The first one is the k -nearest neighbors model classification (k -NN). This algorithm does not assume any type of distribution of the data (non-parametric) and makes a decision based on the multiclass training data set. In this

work, using the Euclidian distance function, k -NN finds the $k = 10$ closest neighbors. Next, the belonging probability of a new observation respect to each training set classes is evaluated [42].

Linear Discriminant model classification (LDA) is a well know technique for data classification and dimensionality reduction. This method maximizes the ratio of between-known classes variance guaranteeing more separability between classes. It also helps to better understand the data distribution. In general terms, the algorithm can be summarized as the calculation of the mean of each class. Where the mean of the entire data set is estimated using the membership probability for each class. Immediately afterwards, the covariance matrix is used to calculate the optimal separability criterion. This same criterion is used to change the vector space using the eigenvalues and eigenvectors. Finally, the test vectors are transformed and classified using the Euclidean distance from each class mean [43].

A decision tree involves a main node root (topmost node), branches with nodes (secondary nodes), and leafs. Each secondary node implies a test on an attribute. Next, each branch indicates the outcome of a test, and finally, each leaf has a class label. The design of the tree can be developed according to the specific needs or also use the general trees that are already established. However, in both cases, overfitting the training data must be avoided. This can be performed before or after finishing the algorithm using Pre-pruning or Post-pruning. In Pre-pruning, the tree is pruned when it is under construction. Contrary, Post-pruning removes one o more sub-trees from a completely build tree [44].

Ensemble model classification Bag-tree (Bag tree) comes from the Decision trees model classification (Trees). Using the philosophy behind machine learning, Bag tree combines the results of many decision trees reducing the excessive adjustment and improves the generalization. In this way, the predictive performance is increased by the weighted combination multiple trees. Then, the final prediction is given by voting or by averaging [45].

The Support Vector Machine (LSVM) model classification using linear Kernel main idea, for linearly separable patterns, is to find the optimum hyperplane which separates and maximizes the distance between classes (functional margin). The optimum hyperplane is the one that maximizes the margin of the training data. The margin is calculated from the maximum distance between the hyperplane and the closest data. This idea is extended to non-linearly patterns using the Kernel functions which takes low dimensional input space and transform it to a higher dimensional space. In this way, the non-separable problem is converted into a separable problem (the hyperplane may have curved shapes) [46].

In a nutshell, the selection of the best classification tool and the selection of the most crucial features is performed in parallel. For each iteration, a list of the descriptors and the overall results of the classification, with a defined configuration, are obtained. The process of deciding the best tool, and the most crucial features, is based on: averaging the overall results (accuracy, false positive rate, true positive rate) and concatenating the lists.

4. Results and discussion

4.1. Generalities

Recall that in this work:

- 8 different scenarios are simulated: one belongs to normal operation and seven belong to different faults (Section 2.7).
- 140 simulations are conducted for healthy condition and 40 by each faulty condition, therefore $140+40\times 7 = 420$ different cases are simulated.
- 13 variables are measured by each case (Section 2.6).
- each variable is represented by an image and, from each image, 65 texture features are extracted (Section 3.2). Thus, $13 \times 65 = 845$ features are initially considered for each case.

The results shown in this section are organized as follows. Five different classifiers are compared and, among them, the best one is selected according to the accuracy criterion and ROC (Receiver Operating Characteristic) curves. By using the best classifier, the optimal number of selected features is determined by analyzing the accuracy of all possibilities. Then, using the best classifier and the optimal number of features, the classification results are shown in terms of the confusion matrix, the ROC curves, accuracy and probability of detection (POD). Finally, with the purpose of reducing the number of variables, an analysis of the most descriptive variables and texture features is conducted.

4.2. Comparison of the different classifiers

As mentioned in Section 3.3.2, five classification tools are tested where the inputs are defined by the best features. These more representative features have been obtained from the analysis of 10 iterations into the cross-validation. In turn, for each of these repetitions, the features are organized in descending order. That is, the first feature is the most relevant and the last one is the least relevant. Then, the first 100 features are used as inputs in the classification tool. In general terms, the selection of the most representative features and the classification of the different scenarios is carried out simultaneously for each iteration. At the end of the iterative process, two measurements per classification tool are obtained:

1. Accuracy
2. Area under the ROC curves (plot of the true positive rate against the false positive rate)

The accuracy indicates the average ratio between the cases correctly classified and the total number of cases in all validation iterations. Thus, an accuracy of 100% represents a perfect classification. The ROC denotes the goodness of the classification and it depends on how well the classification separates the groups. For example, an area of 1 represents a perfect separation between two classes, an area above 0.8 is considered as a good separation, and if it is close to 0.5 represents a worthless separation. In this work, there are more than 2 classes, so, the “one versus all” approach is applied. In other words, one class is taken as positive class and the remaining classes together as the negative class. This procedure is repeated for each class [47].

Table 4 summarizes, for each classification tool, the accuracy (%) and the main parameters used to obtain the best results. It is noteworthy that the Bag Tree algorithm obtains the best accuracy.

Table 4: Accuracy by classification tool

Tool	Accuracy [%]	Parameters
Bag Tree	80.238	Number of ensemble learning cycles = 20; Learners tree
Tree	79.524	Maximum number of decision splits = 20
kNN	45.238	Euclidean distance, Number of neighbors = 10
LDA	46.905	No parameters
LSVM	58.333	Linear kernel, C = 10

Similarly, the relationship between the sensitivity versus the specificity is analyzed by comparing each one of the conditions against the others using the ROC curves. Results for every condition (rows) with each classification tool (columns) are presented in Table 5. The column with the highest values belongs to the Bag Tree algorithm. From this column, it can be emphasized that the lowest area is obtained with fault 6 with 0.909, but it is the highest one compared to the other classification tools. Fault 7 is perfectly classified by all methodologies except *k*-NN.

Table 5: AUROC for each classification tool

Condition	Bag Tree	Tree	kNN	LDA	LSVM
Fault1	0.916	0.810	0.681	0.719	0.737
Fault2	0.930	0.855	0.611	0.718	0.693
Fault3	0.995	0.981	0.602	0.748	0.842
Fault4	1.000	0.986	1.000	0.999	1.000
Fault5	0.970	0.883	0.942	0.940	0.944
Fault6	0.909	0.757	0.869	0.901	0.890
Fault7	1.000	1.000	0.907	1.000	1.000
Normal	0.986	0.938	0.703	0.755	0.783

Comparing simultaneously both tables 4 and 5, it is perceived that the best result is obtained by using Bag Tree algorithm. Thus, a further and deeper analysis is developed focusing only on the results from this classifier.

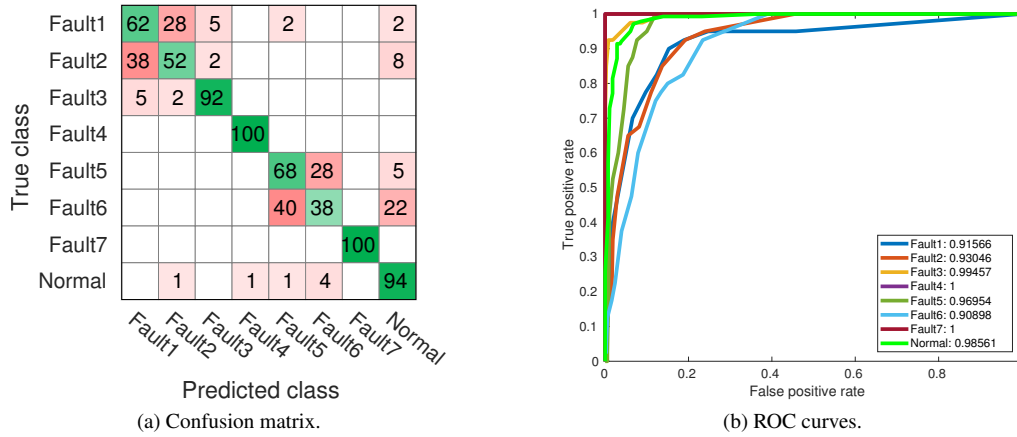


Figure 11: Results of the Bag tree classifier using 100 features

4.3. Optimal number of features

Once the best classification tool is selected (Bag Tree), a deeper analysis is conducted by analyzing the confusion matrix shown in Figure 11a. This matrix indicates the ratio (percentage) of cases (simulations) from each condition (normal and 7 faults) that are correctly classified and the ratio of cases that are wrongly classified specifying each one of the other conditions. For instance, from 40 cases belonging to Fault 1: 62% (24 cases) are correctly classified, 28% (11 cases) are classified as Fault 2, 5% (2) as Fault 3, 2% (2) as Fault 5, and 2% (1) as Normal. Likewise, the ROC curves and the area under the curve (AUROC) for the classification of each condition are depicted in Figure 11b. In general, the relationship in the curve is good: Fault 6 area is the smallest, followed very closely by Fault 1. In the opposite case, the classification is perfect in Faults 4 and 7. Closely followed by Fault 3 and Normal operation.

Although results are promising, the number of selected features (100) was randomly chosen. Thus, next, a brief analysis of the optimal number of features is conducted. The average accuracy of the classification is calculated using the S more relevant features, where $S = 1, 5, 10, 20, \dots, 100$ (see Figure 12). As it can be seen, if more than 10 features are selected, hardly any changes in the accuracy are observed, but the computational cost increases as the number of features does. In conclusion, the optimal number of features is 10 for this specific scenario.

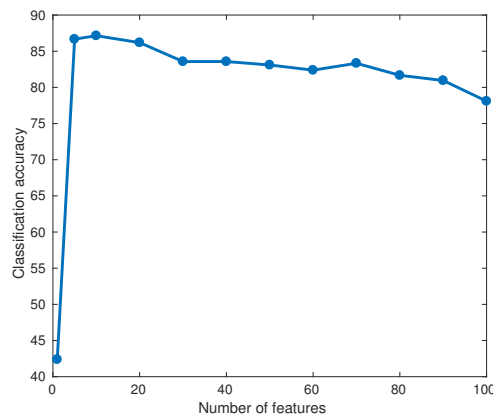


Figure 12: Number of selected features versus classification accuracy for the Bag tree classifier .

By implementing a Bag tree classifier with the 10 most relevant features, the results of the classification are presented by means of: Total accuracy equals to 87.14, Confusion matrix shown in Figure 13a, ROC curves depicted in Figure 13b.

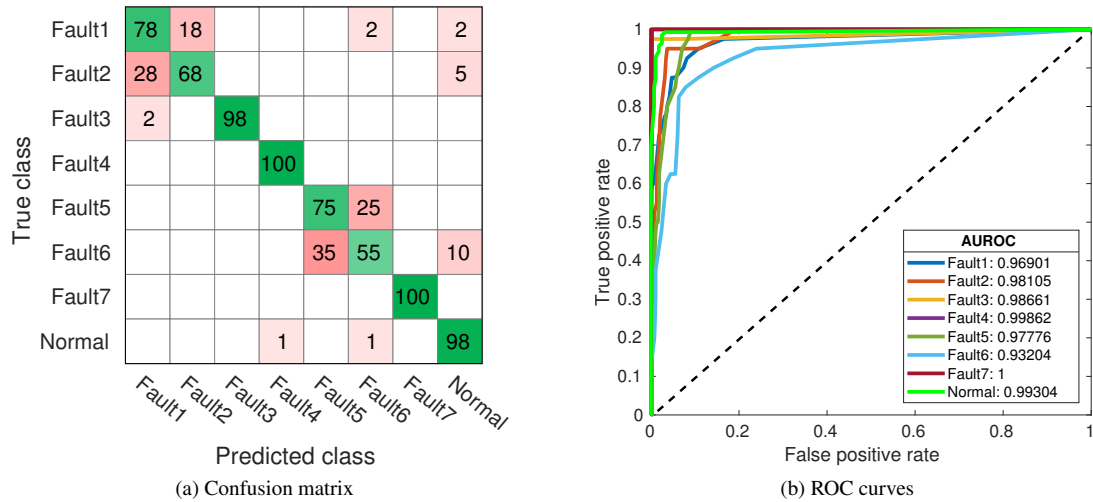


Figure 13: Results of the Bag tree classifier using 10 features

370 In general, a slight improvement is obtained compared to the results with 100 features. In both cases, the classifiers have some problems to distinguish between Faults 1 and 2. Similarly, Faults 5 and 6 are not easily distinguishable and besides, 10% of Fault 6 are classified as normal operation. By analyzing the information provided in Section 2.7 about the faults scenarios, it can be deduced that Faults 1 and 2 could be joined as a single class. Although, Fault 3 has been almost perfectly separated from the others (only 2% was classified as Fault 1), it is included in the same class with Faults 1 and 2 because all of them have the same nature (change in dynamics of the pitch actuator). The same decision is made with Faults 5 and 6 since they are faults in the pitch angle sensor (stuck at 5 and 10 degrees respectively). Thus, Faults 1, 2, and 3 are now labeled Fault123. Faults 5 and 6 are labeled as Fault56. Fault 4, Fault 7 and Normal operation are still labeled in the same way.

380 Using this new grouping of the different scenarios, a new classifier is trained and validated, the resulting total accuracy is 97.62%, the confusion matrix is presented in Figure 14a, the ROC curve in Figure 14b. It is obtained by 40 supervised learners (ensemble ideology) allowing more flexibility among the classes with a low computational cost. Now 97% of the cases with Fault123 are correctly classified. Furthermore, 96% of cases with Fault56 are correctly classified and only 4% as normal operation condition. Similarly, 2% of cases with normal operation are confused as Fault123. On the other hand, the AUROC has observably improved and all areas are above 0.99. As a suggestion, if an extra distinction between Faults 1, 2 and 3 is needed, a new phase of classification should be implemented.

385 Finally, a comparison of both confusion matrices (8 conditions and 5 conditions) shows that the classification has significantly improved when the faults with the same nature have been grouped. Thus, this grouping is kept for the rest of the paper.

390 Finally, to highlight the reliability of the approach, the membership probability is shown in Figure 15, for a representative set of simulations. The height of each color in the bar represents the probability of cases belonging to different subgroups. For instance, in Figure 15 (top left), simulations with Fault 123 are shown. It can be observed that the third simulation has only a chance of 5% of belonging to Fault 123, but a chance of 75% to belong to Fault 56, 5% to Fault 7 and 15% to Normal. This particular simulation was wrongly classified as Fault 56. Similarly, the sixteenth simulation has a chance of 75% of belonging to Normal; therefore this simulation was also misclassified. On the contrary, for the rest of simulations the approach is 100% sure that they belong to Fault 123. The other subfigures in Figure 15 show the other cases: F4 simulations (top right), F56 simulations (mid left), F7 simulations (mid right), Normal simulations (bottom). It can be seen that almost all cases have 100% probability of being correctly classified. Some cases have less probability of being correctly classified but they are still above of the threshold and therefore they are properly classified.

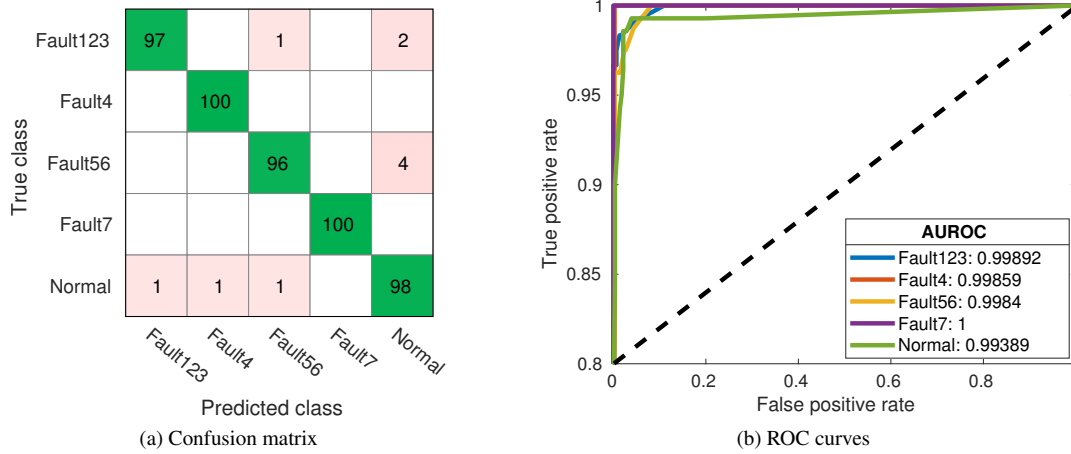


Figure 14: Results of the Bag tree classification using 10 features and 5 different scenarios

4.4. Variable reduction

In this section, the number of variables that can be reduced while still obtaining similar satisfactory results is analyzed. To do this, the best classification tool (Bag Trees) is used, similar faults are grouped, and only the 10 more representative texture features are used, namely:

- Mean of the scale-invariant Gabor texture for an angle of 135°, obtained from the generated power.
- Standard deviation of the first vertical detail from wavelet decomposition, obtained from the third pitch angle.
- Correlation of the generator torque.
- Mean of the generated power.
- Standard deviation of the first diagonal detail from wavelet decomposition, obtained from the third pitch angle.
- Mean of the scale-invariant Gabor texture for an angle of 22°, obtained from the generated power.
- Mean of the scale-invariant Gabor texture for an angle of 45°, obtained from the generated power.
- Mean of the scale-invariant Gabor texture for an angle of 157°, obtained from the generated power.
- Correlation of the rotor speed.
- Mean of the scale-invariant Gabor texture for an angle of 10°, obtained from the generated power.

According to the nature of the texture descriptors, five of them come from Gabor texture features; two from wavelet transformation; one from first-order statistical features; and two from the second-order statistical features. On the other hand, according to the nature of the measured variable: six features belong to generated electrical power; two to pitch angle; one to generator torque; and the last one to rotor speed. In this work, the classification is performed with these four original measured variables in which the statistical mode is the generated electrical power. Hereby, to check these results, the next four cases are studied (following the methodology described in Section 3):

1. Case 1, using only the four most relevant variables: the methodology is validated using only generated electrical power, pitch angle, generator torque and rotor speed.
2. Case 2, without using the four most relevant variables: removing from the original dataset the four previous variables (Case 1), the methodology is validated with the remaining nine electromechanical sensors.
3. Case 3, using only the most relevant variable: only the generated electrical power is used to validate the methodology.

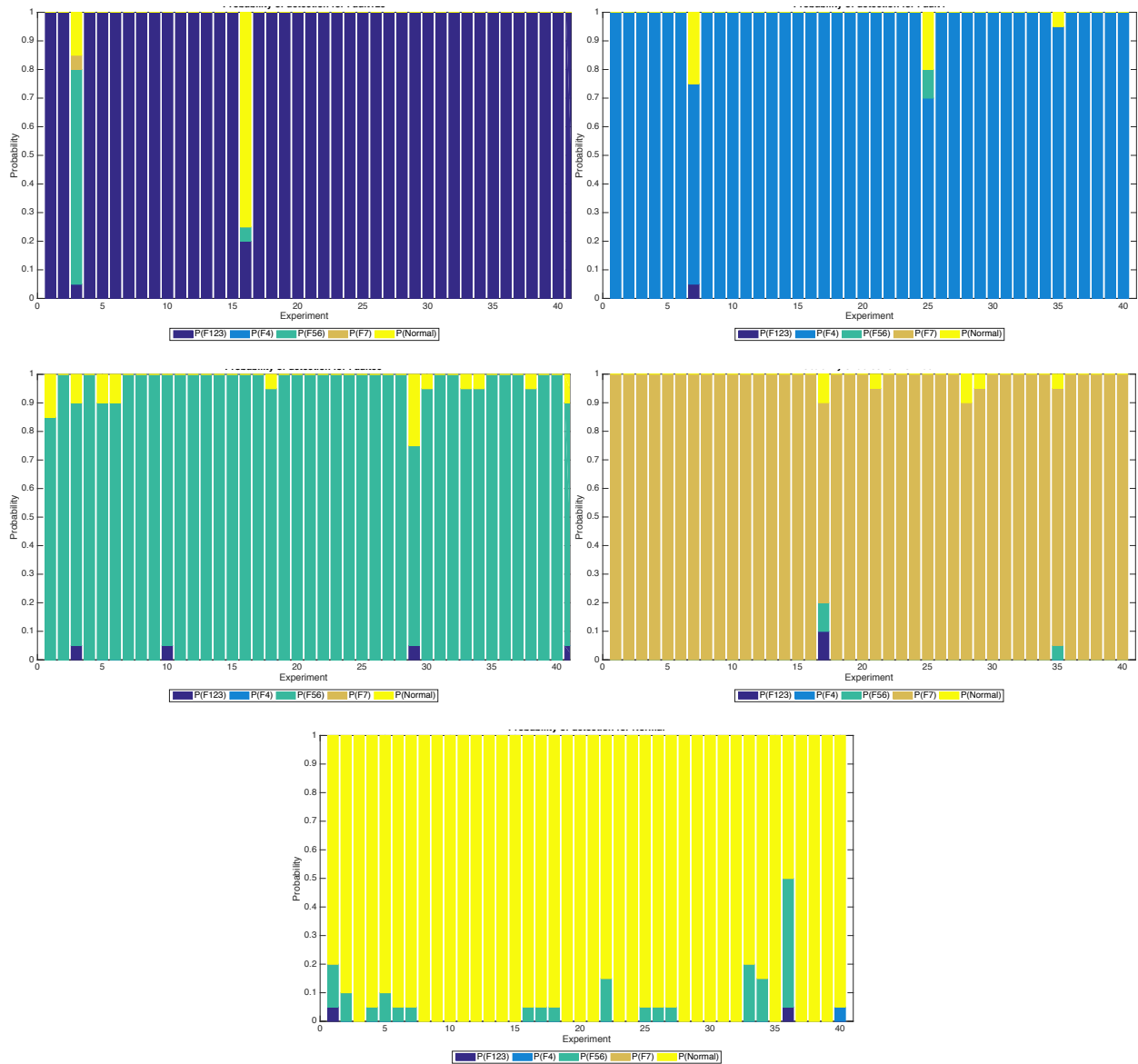


Figure 15: Membership probability of the Bag tree classification by each case using 10 features and 5 different scenarios.

4. Case 4, without using the most relevant variable: removing from the original dataset the generated electrical power (Case 3), the remaining 12 available electromechanical sensors are used to validate the methodology.

430 Table 6 summarizes the ROC curve results. It is noteworthy that for case 1 (using only the 4 most relevant variables) the results are exactly the same as using 13 variables, see Figure 14b, but only 260 features were calculated instead of 845. Conversely, results dramatically worsen when these four variables are eliminated (Case 2). The ROC significantly decreases for Fault123, Fault7 and Normal condition. Finally, the importance of the generated electrical power variable is analyzed with the study of cases 3 and 4. Firstly, using only this variable, results are promising for Fault4, Fault56 and Fault7. The classification decreases for Fault123 and normal condition. These results are quite acceptable since only one variable is used. On the other hand, using the remaining 12 variables, only Fault7 has an important decrease and the other four operating conditions can be recognized as well classified. This can be physically
 435

understood from the fact that Fault7 is related to the torque actuator, which is directly related to the electrical power (see Eq. 2), which is the only variable that is not being used in this case.

Table 6: Four case studies using a reduced number of variables - AUROC

	Operation condition	AUROC
Case 1	Fault123	0.999
	Fault4	0.999
	Fault56	0.998
	Fault7	1.000
	Normal	0.994
Case 2	Fault123	0.682
	Fault4	1.000
	Fault56	0.965
	Fault7	0.573
	Normal	0.698
Case 3	Fault123	0.786
	Fault4	0.999
	Fault56	0.966
	Fault7	1.000
	Normal	0.782
Case 4	Fault123	0.993
	Fault4	1.000
	Fault56	0.990
	Fault7	0.735
	Normal	0.903

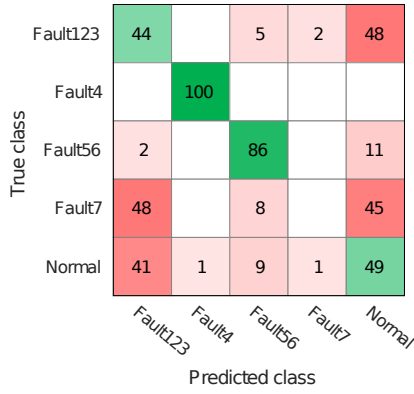
Finally, the confusion matrices and the ROC curves are also presented. For case 1, results are identical to those shown in Figures 14a and 14b. In Case 2, from the confusion matrix (Figure 16a) it is obvious that the accuracy is drastically reduced for Fault 123 and the normal operating condition (the cases are evenly distributed among them). In some systems, confusing different faults is not a severe problem but if a fault is hidden as normal operation, this missing fault could cause large losses. On the other hand, cases with Normal operation are distributed between Fault123 and Normal, leading to some false alarms. Finally, any case from Fault 7 is correctly classified. Observing the ROC curve (Figure 16b), Fault7 is close to the approval limit. Fault123 and Normal operation condition does not differ from Fault7 curve. Fault4 and Fault56 obtain satisfactory results.

Variable GenPower is separately studied in Case 3, and results are shown in Figures 16c and 16d. Fault123 and Normal condition are not distinguishable between them. The accuracy of classifying Fault56 decreases. It can be appreciated, also, that Normal and Fault123 have not satisfactory results regarding to rate of true positives and false positives, while for the other faults the classification is remarkable. However, these results are admissible considering that only one variable is used.

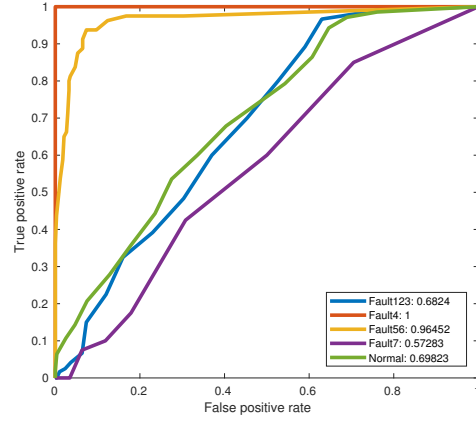
In case 4, there is a small improvement compared to Case 3 in the classification accuracy (Figure 16e) and the ROC (Figure 16f). However 12 variables are being used here and the computational cost is high. For all cases, the values of the AUROC of Figures 14b, 16b, 16d and 16f coincide with the values presented in Table 6.

5. Conclusions

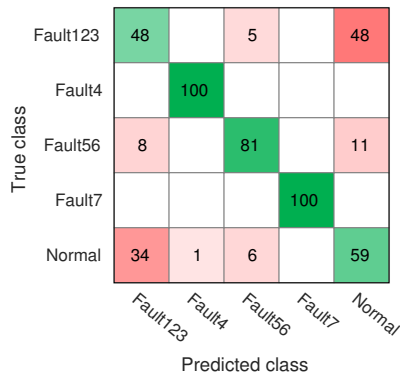
The main challenges of wind turbine fault detection lie in its nonlinearity and unknown disturbances (such as the wind). Expensive condition monitoring systems are often used to monitor the wind turbine, and typically also using additional often expensive sensor signals. On the other hand, most wind farms have installed supervisory control and data acquisition (SCADA) systems for system control and logging data. However, the collected data are not used effectively. This paper proposes a method for early fault identification for the main components of wind turbines



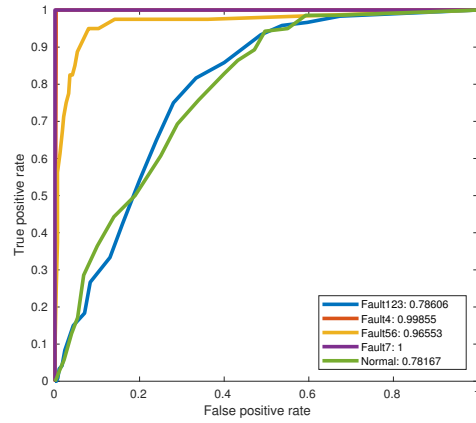
(a) Confusion matrix for Case 2



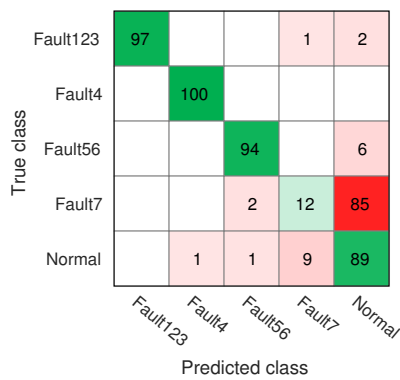
(b) ROC curves for Case 2



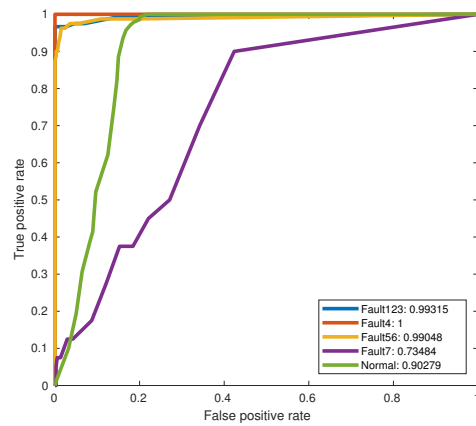
(c) Confusion matrix for Case 3



(d) ROC curves for Case 3



(e) Confusion matrix for Case 4



(f) ROC curves for Case 4

Figure 16: Results for the different cases in Table 6

460 based on the existing data collected by commercial SCADA systems, and therefore no new installation of specific sensors or diagnostic equipment is required.

The paper applies the idea to represent SCADA time domain signals to grayscale images and then study different texture feature selection and classification algorithms by means of stratified cross validation.

465 The comparison of the different classifiers is based on the accuracy and ROC curves. It is found that, despite the complex configuration of the Bagged trees, it has a lesser computational cost and gives better accuracy than the others classification tools. Thus, a further and deeper analysis is developed focusing only on the results from this classifier.

A brief analysis of the optimal number of features shows that the optimal number of features is 10 for this specific scenario. By implementing a Bag tree classifier with the 10 most relevant features, the classifier has some problems to distinguish between Faults 1 and 2, and between Faults 5 and 6, and also are not easily distinguishable and besides, 470 10% of Fault 6 are classified as normal operation. To overcome this issue, a new grouping is proposed, where similar faults (fault related to the same component) are grouped together.

Using this new grouping (5 different scenarios), numerical simulations with a well-known benchmark 5MW wind turbine show that the proposed methodology is capable of classifying between the studied faults (most common in MW-scale industrial wind turbines) with a remarkable overall accuracy of 98% when using the 13 measured variables, 475 the Bag Tree classifier, and 10 features.

Finally, the number of variables that can be reduced while still obtaining similar satisfactory results is analyzed. It is noteworthy that exactly the same results are obtained when using only the four most relevant variables, namely, electrical power, pitch angle, generator torque and rotor speed. From a physical point of view, it makes sense that these are the most relevant variables as they are the most closely related to all the studied actuators and sensors (pitch 480 actuator, generator speed sensor, pitch angle sensor and torque actuator). It is meaningful that the electrical power is the most relevant variable. By using only this one variable, almost all studied faults are detectable and well classified by the exception of the ones related to the pitch actuator.

Future work will introduce the testing in different conditions containing normal level, warning level and alarm level; thereby giving the operators sufficient time to make more informed decisions regarding the maintenance of their 485 machines.

Acknowledgements

This work has been partially funded by the Spanish Ministry of Economy and Competitiveness through the re- search projects DPI2014-58427-C2-1-R, DPI2015-64170-R(MINECO/FEDER), DPI2015-64493-R(MINECO/FEDER), and by the Catalonia Government through the research project 2014 SGR 859.

490 References

- [1] Iury Valente de Bessa, Reinaldo Martinez Palhares, Marcos Flávio Silveira Vasconcelos D'Angelo, and João Edgar Chaves Filho. Data-driven fault detection and isolation scheme for a wind turbine benchmark. *Renewable Energy*, 87:634–645, 2016.
- [2] Xiang Gong and Wei Qiao. Current-based mechanical fault detection for direct-drive wind turbines via synchronous sampling and impulse detection. *IEEE Transactions on Industrial Electronics*, 62(3):1693–1702, 2015.
- 495 [3] Wei Teng, Xian Ding, Xiaolong Zhang, Yibing Liu, and Zhiyong Ma. Multi-fault detection and failure analysis of wind turbine gearbox using complex wavelet transform. *Renewable Energy*, 93:591–598, 2016.
- [4] Wei Qiao, and Dingguo Lu. A survey on wind turbine condition monitoring and fault diagnosis—Part I: Components and subsystems. *IEEE Transactions on Industrial Electronics*, 62(10):6536–6545, 2015.
- [5] Wei Qiao, and Dingguo Lu. A survey on wind turbine condition monitoring and fault diagnosis—Part II: Signals and signal processing methods. *IEEE Transactions on Industrial Electronics*, 62(10):6546–6557, 2015.
- 500 [6] Pierre Tchakoua, René Wamkeue, Mohand Ouhrouche, Fouad Slaoui-Hasnaoui, Tommy Andy Tameghe, and Gabriel Ekemb. Wind turbine condition monitoring: State-of-the-art review, new trends, and future challenges. *Energies*, 7(4):2595–2630, 2014.
- [7] Md Rifat Shahriar, Tanveer Ahsan, and UiPil Chong. Fault diagnosis of induction motors utilizing local binary pattern-based texture analysis. *EURASIP Journal on Image and Video Processing*, 2013(1):1–11, 2013.
- 505 [8] Francesc Pozo and Yolanda Vidal. Wind turbine fault detection through principal component analysis and statistical hypothesis testing. *Energies*, 9(1):3, 2015.
- [9] Santiago Alférez, Anna Merino, Laura Bigorra, Luis Mujica, Magda Ruiz, and Jose Rodellar. Automatic recognition of atypical lymphoid cells from peripheral blood by digital image analysis. *American journal of clinical pathology*, 143(2):168–176, 2015.
- 510 [10] Van Tuan Do, and Ui-Pil Chong. Signal model-based fault detection and diagnosis for induction motors using features of vibration signal in two-dimension domain. *Strojniški vestnik-Journal of Mechanical Engineering*, 57(9):655–666, 2011.

- [11] Thomas W. Rauber, Francisco de Assis Boldt, and Flávio Miguel Varejão. Heterogeneous feature models and feature selection applied to bearing fault diagnosis. *IEEE Transactions on Industrial Electronics*, 62(1):637–646, 2015.
- [12] Renping Shao, Wentao Hu, Yayun Wang, and Xiankun Qi. The fault feature extraction and classification of gear using principal component analysis and kernel principal component analysis based on the wavelet packet transform. *Measurement*, 54:118–132, 2014.
- 515 [13] Tianlu Chen, Gui Yun Tian, Ali Sophian and Pei Wen Que. Feature extraction and selection for defect classification of pulsed eddy current NDT. *Ndt & E International*, 41(6):467–476, 2008.
- [14] Peter F Odgaard and Kathryn E Johnson. Wind turbine fault detection and fault tolerant control-an enhanced benchmark challenge. In *American Control Conference (ACC), 2013*, pages 4447–4452. IEEE, 2013.
- [15] Jason M Jonkman and Marshall L Buhl Jr. Fast users guide. *National Renewable Energy Laboratory, Golden, CO, Technical Report No. NREL/EL-500-38230*, 2005.
- 520 [16] David S Ochs, Ruth Douglas Miller, and Warren N White. Simulation of electromechanical interactions of permanent-magnet direct-drive wind turbines using the fast aeroelastic simulator. *Sustainable Energy, IEEE Transactions on*, 5(1):2–9, 2014.
- [17] Yolanda Vidal, Leonardo Acho, Ningsu Luo, Mauricio Zapateiro, and Francesc Pozo. Power Control Design for Variable-Speed Wind Turbines. *Energies*, 5(8):3033–3050, AUG 2012.
- [18] Brice Beltran, Mohamed El Hachemi Benbouzid, and Tarek Ahmed-Ali. Second-order sliding mode control of a doubly fed induction generator driven wind turbine. *Energy Conversion, IEEE Transactions on*, 27(2):261–269, 2012.
- [19] J. Jonkman, S. Butterfield, W. Musial, and G. Scott. Definition of a 5 MW reference wind turbine for offshore system development. Technical report, National Renewable Laboratory (NREL), Golden, Colorado, 2009.
- [20] Neil Kelley and Bonnie Jonkman. NWTC computer-aided engineering tools (Turbsim), Last modified 30-May-2013.
- 530 [21] Peter F Odgaard, Jakob Stoustrup, and Michel Kinnaert. Fault-tolerant control of wind turbines: A benchmark model. *IEEE Transactions on Control Systems Technology*, 21(4):1168–1182, 2013.
- [22] B. Beltran, T. Ahmed-Ali, and M. El Hachemi Benbouzid. Sliding mode power control of variable-speed wind energy conversion systems. *Energy Conversion, IEEE Transactions on*, 23(2):551–558, june 2008.
- [23] R. Chaaban, D. Ginsberg, and C.P. Fritzen. Structural load analysis of floating wind turbines under blade pitch system faults. In N. Luo, Y. Vidal, and L. Acho, editors, *Wind Turbine Control and Monitoring*, Advances in Industrial Control, pages 301 – 334. Springer International Publishing, 2014.
- 535 [24] Asier Diaz de Corcuera, Aron Pujana-Arrese, JoseM. Ezquerria, Aitor Milo, and Joseba Landaluze. Design of robust controllers for load reduction in wind turbines. In N. Luo, Y. Vidal, and L. Acho, editors, *Wind Turbine Control and Monitoring*, Advances in Industrial Control, pages 97–133. Springer International Publishing, 2014.
- 540 [25] Peter F Odgaard, Jakob Stoustrup, and Michel Kinnaert. Fault-tolerant control of wind turbines: A benchmark model. *Control Systems Technology, IEEE Transactions on*, 21(4):1168–1182, 2013.
- [26] Rannam Chaaban, Daniel Ginsberg, and Claus-Peter Fritzen. Structural load analysis of floating wind turbines under blade pitch system faults. In Leonardo Acho Ningsu Luo, Yolanda Vidal, editor, *Wind Turbine Control and Monitoring*, pages 301–334. Springer, 2014.
- 545 [27] Jesper Liniger, Henrik C Pedersen, and Mohsen Soltani. Reliable fluid power pitch systems: A review of state of the art for design and reliability evaluation of fluid power systems. In *ASME/BATH 2015 Symposium on Fluid Power and Motion Control*, pages V001T01A026–V001T01A026. American Society of Mechanical Engineers, 2015.
- [28] Lejun Chen, Fengming Shi, and Ruska Patton. Active ftc for hydraulic pitch system for an off-shore wind turbine. In *Control and Fault-Tolerant Systems (SysTol), 2013 Conference on*, pages 510–515. IEEE, 2013.
- [29] Andrzej Materka, Michal Strzelecki, et al. Texture analysis methods—a review. *Technical university of lodz, institute of electronics, COST B11 report, Brussels*, pages 9–11, 1998.
- 550 [30] Rafael C Gonzalez and Richard E Woods. *Digital Image Processing (3rd Edition)*. Prentice Hall, 3 edition, 2007.
- [31] Santiago Alférez. *Methodology for automatic classification of atypical lymphoid cells from peripheral blood cell images*. PhD thesis, Universitat Politècnica de Catalunya, 2015.
- [32] Robert M Haralick, Karthikeyan Shanmugam, and Its’ Hak Dinstein. Textural features for image classification. *Systems, Man and Cybernetics, IEEE Transactions on*, 6:610–621, 1973.
- 555 [33] Fritz Albrechtsen et al. Statistical texture measures computed from gray level cooccurrence matrices. *Image processing laboratory, department of informatics, university of oslo*, pages 1–14, 2008.
- [34] S.G. Mallat. Multifrequency channel decompositions of images and wavelet models. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(12):2091–2110, 1989.
- 560 [35] S Arivazhagan and L Ganesan. Texture classification using wavelet transform. *Pattern recognition letters*, 24(9):1513–1521, 2003.
- [36] Jean Serra. Introduction to mathematical morphology. *Comput. Vision Graph. Image Process.*, 35(3):283–305, September 1986.
- [37] Jesus Angulo. A mathematical morphology approach to cell shape analysis. In *Progress in Industrial Mathematics at ECMI 2006*, pages 543–547. Springer, 2008.
- [38] Peter Kruizinga and Nikolay Petkov. Nonlinear operator for oriented texture. *Image Processing, IEEE Transactions on*, 8(10):1395–1407, 1999.
- 565 [39] Ju Han and Kai-Kuang Ma. Rotation-invariant and scale-invariant gabor features for texture image retrieval. *Image and vision computing*, 25(9):1474–1481, 2007.
- [40] Gavin Brown, Adam Pocock, Ming-Jie Zhao, and Mikel Luján. Conditional likelihood maximisation: a unifying framework for information theoretic feature selection. *The Journal of Machine Learning Research*, 13(1):27–66, 2012.
- 570 [41] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–38, August 2005.
- [42] L. E. Peterson. K-nearest neighbor. *Scholarpedia*, 4(2):1883, 2009. revision #136646.
- [43] Geoffrey J McLachlan. *Discriminant Analysis and Statistical Pattern Recognition*. Wiley, Newark, NJ, 2005.
- [44] Sreerama K. Murthy. Automatic construction of decision trees from data: A multi-disciplinary survey. *Data Mining and Knowledge Discovery*, 2:345–389, 1997.
- 575

- [45] Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.
- [46] Ingo Steinwart and Andreas Christmann. *Support Vector Machines*. Springer Publishing Company, Incorporated, 1st edition, 2008.
- [47] Tom Fawcett. An introduction to ROC analysis. *Pattern Recogn. Lett.*, 27(8):861–874, June 2006.