Structural Damage Diagnosis of Wind Turbine Blades Based on Machine Learning Techniques

José Figueroa, José M. Saavedra, José F. Delpiano, and Rodrigo Astroza(✉)

Faculty of Engineering and Applied Sciences, Universidad de los Andes, Santiago, Chile
rastroza@miuandes.cl

Abstract. This paper presents a method for damage identification of wind turbine blades based on vibration data and machine learning (ML) techniques and their validation using experimental data collected at different states of artificially-induced damage. The acceleration responses collected from accelerometers placed along the blades are preprocessed according to the type of network used for damage diagnosis. The ML approach is a supervised strategy in which a multilayered perceptron (MLP) takes a vector of damage-sensitive features, calculated from the acceleration time series. The accuracy of the approach is evaluated, and the effects of the operational and environmental variables (EOV) are discussed.

Keywords: Damage Diagnosis · Structural Health Monitoring · Wind Turbine · Artificial Neural Network

1 Introduction

Wind energy is one of the most widely used forms of renewable energy generation in the world, providing (alongside solar) more than 10% of the world's electricity according to Renewables 2022 Global Status Report [9]. For the generation to be optimal, the blades of the wind turbines (WTs) must be in operational condition, otherwise, the blade could fail due to the structural stresses caused by the wind hitting it, delamination of the fiberglass layers, amongst others detailed in [5], causing the disruption of operation to replace it or the exchange of the complete generator, incurring in significant economic losses. Thus, knowing the state of health (SoH) of the blades becomes a relevant indicator for reliability, safety, and economic reasons. This creates the need to monitor and diagnose the health state of the structure. Structural health monitoring (SHM) uses different techniques to accomplish this task. To diagnose the structures, the system needs data and as with most industrial machinery, WTs count with a Supervisory, Control, and Data Acquisition system or SCADA that measures relevant information about the structure (such as blade pitch, power generated, and wind speed) over a period of time. This information paired with different machine learning techniques allowed for several approaches to be raised. In Mylonas et al. [7] the authors introduce a machine learning-based strategy to use this information, paired with simulations

M. P. Limongelli et al. (Eds.): EVACES 2023, LNCE 433, pp. 458–467, 2023.
https://doi.org/10.1007/978-3-031-39117-0_47

of the blades, to infer the accumulated deterioration of the WTBs. For this, a conditional variational autoencoder was used to model the probability distribution of said wear. Another approach is postulated by Chandrasekhar et al. [1], where Gaussian Processes are trained to predict the edge frequency of WTBs based on the SCADA information of another of the blades combined with the ambient temperature, and then by calculating the residual error between the actual frequency and the prediction; and by putting a threshold to this error the blade is categorized as healthy or damaged. Another popular approach that tries to generate more information than the SCADA system is by instrumenting the structure to obtain, usually, acceleration responses that are analyzed and processed to obtain features that allow the identification of the SoH of the structure. This vibration-based SHM allows the calculation of new features that contain more information, and that can represent the different states of the structure in a better way. This allows for new types of analysis, such as modal analysis where the natural frequencies and mode shapes are used to characterize the healthy state, and by using artificial neural networks (ANNs) [3], statistical and/or modal damage detection [8] the same goal is achieved. Also, strategies based on outlier analysis, such as Tcherniak et al. [10], where Principal Component Analysis (PCA) was applied to represent the healthy state and uses Mahalanobis distance (MD) as a damage detection metric in a laboratory environment, and, in the same line of research, Ulriksen et al. [11] uses a similar strategy but in an operating WT. In the same manner, but without using PCA, Movsessian et al. [6] introduces an ANN that is trained to output this distance, and the damage detection is made by calculating the absolute logarithmic error between the output and the calculated MD. In another technique presented by Feijóo et al. [2], the authors used an autoencoder to reconstruct the healthy data obtained from accelerometer signals taken from an offshore wind turbine jacket replica. Instead of using dimensionality reduction techniques, such as PCA, the authors calculate damage-sensitive features (DSFs) by using statistical values (mean and standard deviation) and frequency features (spectral entropy) in different SoH of the structure, these states are artificially created by introducing cracks on certain parts of the jacket. The approach is similar to the previously mentioned, as the output is used to calculate the reconstruction error that, by using a threshold, classifies SoH.

In this paper, a different approach to the previously discussed is presented. Instead of training only with healthy data, acceleration responses of the blade with different types and levels of damage were collected, labeled, and then used to train the network. For this, DSFs are calculated in a similar fashion to [2], and then fed to an ANN that is trained to classify the input to their corresponding damage state.

2 Experimenal Setup and Damage States

In this study, a replica of a 5 kW WT is used with three 2.4-meter-long blades. The speed at which the replica rotates is controlled to simulate different operating wind speeds. On top of the simulated WT, a data acquisition system was

designed to capture acceleration data. It is composed of three strips of three accelerometers each placed along two of the three blades of the WT, as shown in Fig. 1. Each sensor is an ADXL-345 triaxial accelerometer from Analog Devices, controlled by an ESP32 that records and sends the acceleration data. The measurements were made in ten-minute intervals with the sampling frequency of the sensors set to 200 Hz. Vibration data was collected at different SoH of the blade by inducing damage or modifications to the physical system. This was made firstly by adding masses in specific sections of the blade to simulate events such as ice accumulation and secondly by introducing cuts at specific blade locations where cracks were previously observed during experimental tests conducted in a laboratory environment [4]. The position of the masses and the location and direction of the cuts are shown in Fig. 2. The combination between masses, cuts,

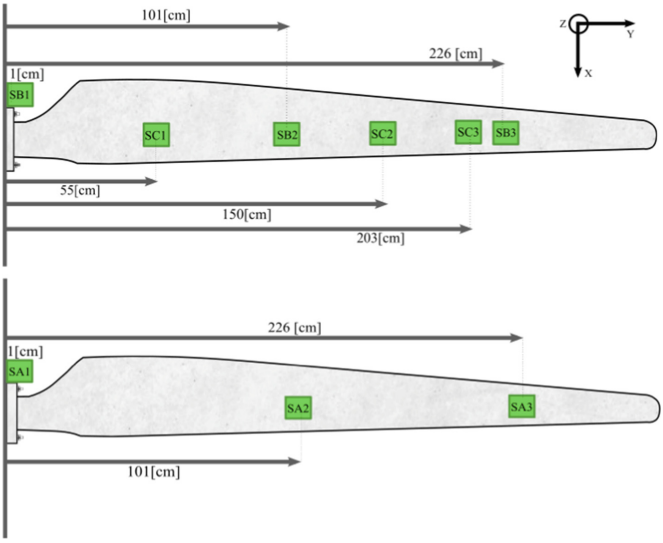

Fig. 1. Instrumentation of WT blades A (top) and B (bottom) with the respective distances from the base. The directions represent the measurement axes of the sensors.

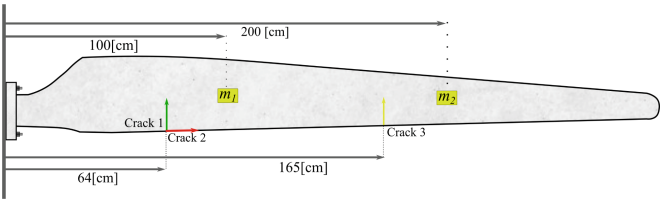

Fig. 2. Location of cuts and masses inflicted on the blade

and rotation speed receives a label of 'WT0#_i' where the '#' indicates the identification number of added mass or cuts and the sub-indices the velocity at which the WT is spinning. Table 1 shows three different SoH used for this report.

Table 1. Test Protocols

Condition	Test	Rotor Speed [RPM]	m_1 [g]	m_2 [g]	c_1 [cm]	c_2 [cm]	c_3 [cm]
Healthy	WT00_1	15	0	0	–	–	–
Healthy	WT00_2	30	0	0	–	–	–
Healthy	WT00_3	45	0	0	–	–	–
Healthy	WT00_4	60	0	0	–	–	–
Added Mass	WT01_1	15	120	120	–	–	–
Added Mass	WT01_2	45	120	120	–	–	–
Damage	WT03_1	15	0	0	2	–	–
Damage	WT03_2	30	0	0	2	–	–
Damage	WT03_3	45	0	0	2	–	–
Damage	WT03_4	60	0	0	2	–	–

3 Damage Diagnosis Approach

As shown in Table 1, the different SoH can be categorized, so the approach selected to diagnose the system is of a supervised classification problem, where the data is fed through a neural network to obtain the class it belongs to at the end. The chosen architecture is a multi-layer perceptron (MLP).

3.1 Data Preprocessing

For the input of the network, features have to be extracted from the responses obtained. As shown by Feijóo et al. [2], the mean (μ), standard deviation (σ), and spectral entropy (H) are representative features that contain enough information for an artificial neural network (ANN) to learn to differentiate the states. These features are calculated over 200 samples or one second of measurement.

3.2 Feature Vector

For the analysis, flapwise and edgewise accelerations (Z and X on Fig. 1, respectively) are relevant as there should be no significant acceleration along the longitudinal axis of the blades. So for six sensors, two axes each, and three features per axes, the input vector for the MLP is composed of these 36 features. The feature extraction is defined by the following equations.

First, a set of measurements is defined by Eq. 1

$$A_t^{s,d} = [a_{t_1}^{s,d}, ..., a_{t_k}^{s,d}] \tag{1}$$

where $a_{t_1}^{s,d}$ is the acceleration of the sensor s in the direction d at time t, and $k = 200$ as it is the window for calculating the features. Then, the sets of sensors is defined in Eq. 2

$$A_t = [A_t^{1,X}, A_t^{1,Z}, ..., A_t^{6,X}, A_t^{6,Z}] \tag{2}$$

where t represents the instant (or second) of the samples. After the sets of sensors, the feature extraction is defined as the function shown in Eq. 3 where the three features are extracted in each window.

$$F(A_{t,s,d}) = [\mu(A_t^{s,d}), \sigma(A_t^{s,d}), H(A_t^{s,d})] \tag{3}$$

Then the feature vector can be defined the following way

$$\mathbf{v}_t = [F(A_{t,1,X})), F(A_{t,1,Z}), F(A_{t,2,X}), ..., F(A_{t,6,X}), F(A_{t,6,Z})] \tag{4}$$

So for the same second, the features of the sensors are organized from one to six or, by using the nomenclature of Fig. 1 from A1 to B3. Finally, each feature is normalized independently using Z-Score as shown in Eq. 5.

$$f_n = \frac{f - \mu_f}{\sigma_f} \tag{5}$$

where f_n is the normalized feature, μ_f is the mean value of the feature in the SoH, and σ_f is the standard deviation of the feature in said SoH. After the normalization process, the vectors can be projected into 2D using UMAP. This is done to understand the distribution of the points on their own dimensionality as UMAP tries to maintain the topological characteristics of the sample when it projects. Figure 3 shows that the classes are mostly separated, but there are some vectors that are on top of others making the problem more complex.

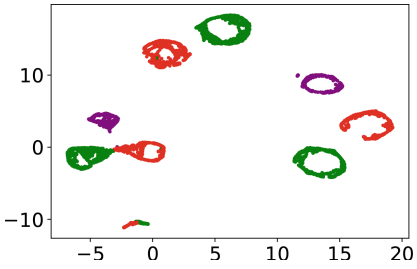

Fig. 3. 2D UMAP projection of 5000 feature vectors, in green WT00, in purple WT01, and in red WT03. Some overlap can be seen in the lower left corner between WT00 and WT03 vectors.

3.3 MLP Architecture

The proposed architecture of the MLP is composed of three fully connected layers. As discussed in the previous section, the input layer has 36 input neurons, the first hidden layer of 128 neurons, a second hidden layer of 32 neurons, and at the end the output layer with three neurons (one per category); Fig. 4 shows the summarised architecture. Due to the nature of the signals, even after normalization, they have positive and negative values, this is why the chosen activation function for both hidden layers is hyperbolic tangent (tanh), as the function takes both positive and negative values. At the output, softmax activation is used to obtain the probability of the input belonging to a certain label. Before the training process, the data is divided into two sets of training and testing with an 80%/20% split. For training, the loss function (also known as the cost function), as is modeled as a multi-label classification problem, is cross-entropy paired Stochastic Gradient Descent (SGD) as the optimizer. Finally, the network was trained for 5 epochs with a batch size of 64.

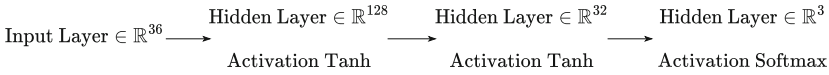

Fig. 4. Proposed MLP Architecture

3.4 Metrics

To measure the performance of the network, a few metrics were used, these are; Accuracy, Precision, Recall, F1-Score, and the confusion matrix. All of the metrics are defined in relation to the amount of correctly classified samples or True Positives (TP) and True Negatives (TN) and the incorrectly classified samples False Positives (FP) and False Negatives (FN). The metrics are defined in the following equations.

$$Accuracy = \frac{TP + TN}{TN + FP + FN + TP} \tag{6}$$

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$F1 = 2\frac{Precision \cdot Recall}{Precision + Recall} \tag{9}$$

4 Results and Discussion

Before the architecture was fixed, several parameters were changed and tested. This was made by training different combinations of the number of neurons per layer and the number of layers and keeping the best-performing based on convergence time. This architecture, shown in Fig. 4, was then trained and modified according to the experiment being performed. First, the classification using all SoH.

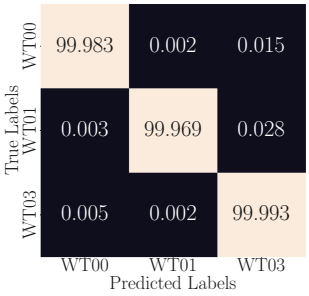

Fig. 5. Confusion Matrix of the trained MLP calculated with testing data, all values are percentages

Table 2. Classification Metrics of the proposed methodology

Class	Precision	Recall	F1-Score	Number of Test Samples
WT00	100.0	100.0	100.0	192225
WT01	100.0	100.0	100.0	67339
WT03	100.0	100.0	100.0	166492

The network achieves 100% accuracy, this result also repeats on the different metrics presented in Table 2. In addition to the metrics, the confusion matrix in Fig. 5 shows how well the network classifies the data and how many samples it confuses and with which label. For WT00 a 0.017% or 32 misclassified samples, 21 for WT01, and for WT03 11 samples. From these results, this methodology has an auspicious future for damage detection, as all metrics are extremely good with two different types of damage and healthy data. This could be attributed to the conditions in which the replica was operated, as the rotation speed was fixed and not dependent on the wind. Also, the temperature variability was not significant during the tests. In order to check if these environmental and operational variables (EOV) are relevant to the classification, the training, and testing were repeated, including them in the input vector.

4.1 EOV Effect

As previously explained, the network takes three features calculated with the acceleration obtained from the sensors. In addition to those vectors, the values

of the mean rotor speed and mean temperature (over the same 200 samples) were added to them. This means that the input changes to 38, but all the rest of hyperparameters are maintained.

Table 3. Classification metrics of the MLP trained with EOVs

Class	Precision	Recall	F1-Score	Number of Test Samples
WT00	100.0	100.0	100.0	60242
WT01	100.0	100.0	100.0	34548
WT03	99.9	100.0	100.0	25098

With these new features, the results are maintained from the previous as shown by the confusion matrix Fig. 6 and the metrics in Table 3. With the addition of EOVs, the confused samples are reduced, but not in a significant manner. This means that the approach works only with the vibrational data and the effects that EOVs have on the features are not meaningful enough to change them, and the variables themselves do not have relevant information for the classification. But, this methodology has to be validated with a working WT in regular operational conditions.

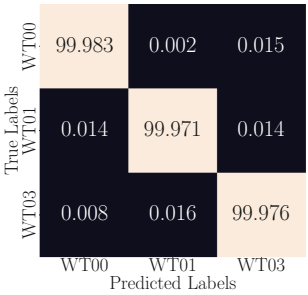

Fig. 6. Confusion Matrix of the trained MLP trained with EOVs, all values are percentages

4.2 Network Generalization

In addition to the network knowing the classification of the damage, it is important to know how the network will respond when a new state is presented to it, this is why a binary experiment was performed. In this case scenario, the labels of the classes were changed to healthy or damaged (true or false), instead of the corresponding 'WT0# label, to create a binary classification problem. This comes with the change of activation function and the number of neurons in the

output layer. The neurons are changed to two, one per class, and the activation function is changed to sigmoid. This is done to allow for the training of both neurons, as the healthy data enters the network, the 'healthy' neuron trains to converge to '1' and the 'damaged' neuron converges to '0' and vice-versa. For this, the training and testing dataset are built using different SoHs. The training dataset is composed of WT00 and WT01 and the testing dataset is made up from WT00 and WT03.

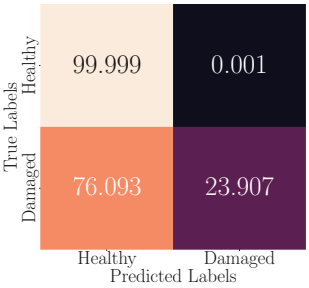

Fig. 7. Generalization experiment confusion matrix, all values are percentages

Table 4. Classification metrics for the generalization experiment

Class	Precision	Recall	F1-Score	Number of Test Samples
Healthy	60.0	100.0	75.2	288338
Damaged	100.0	23.9	38.6	249738

With this, the results show that the network is able to classify the healthy state of the blade almost to a 100%. This is expected as the healthy state does not change with respect to the features, but the damaged state behaves in a completely different manner, as the metrics in Table 4 and the confusion matrix in Fig. 7 show, the damaged state does not generalize in a useful manner, this can be seen in Fig. 3 as the clusters in purple (WT01) are separated from the clusters in red (WT03) so there is no overlap for the MLP to interpret both as damages.

5 Conclusions

This paper presented a supervised machine-learning methodology for damage identification in wind turbine blades (WTBs) that is capable of differentiating between types of damage and correctly classifying them with an accuracy of 100%. The proposed approach has advantages compared to previous techniques as it includes more information on the label, apart from the distinction between healthy and damaged states. The methodology was validated using experimental data from a 2.4-meter-long wind turbine blade, which was tested with added

masses and with artificially-induced damage. In addition, the effects of environmental and operational variables (EOV) were also investigated. The proposed methodology proves to be a promising strategy for structural health monitoring (SHM) for WTBs. In addition to the high accuracy, it is remarkable the amount of data needed for the diagnosis, as the features are calculated over 200 samples, and the sampling frequency is 200 Hz, meaning that the network only needs one second of measurements for the diagnosis.

In regard to the generalization experiment, the representation can be improved by training with more types of damages, as the features locate different damages in different regions of the feature vector space and similar damages should be closer. Finally, the next step for this methodology is to be validated with data obtained from an instrumented blade on a working wind farm and explore how the features and performance of the network differ from the controlled setup.

References

1. Chandrasekhar, K., Stevanovic, N., Cross, E.J., Dervilis, N., Worden, K.: Damage detection in operational wind turbine blades using a new approach based on machine learning. Renew. Energy **168**, 1249–1264 (2021)
2. Feijóo, M.D.C., Zambrano, Y., Vidal, Y., Tutivén, C.: Unsupervised damage detection for offshore jacket wind turbine foundations based on an autoencoder neural network. Sensors **21**(10), 3333 (2021)
3. Gantasala, S., Luneno, J.C., Aidanpää, J.O.: Identification of ice mass accumulated on wind turbine blades using its natural frequencies. Wind Eng. **42**(1), 66–84 (2018)
4. Jaramillo, F., Gutiérrez, J.M., Orchard, M., Guarini, M., Astroza, R.: A Bayesian approach for fatigue damage diagnosis and prognosis of wind turbine blades. Mech. Syst. Signal Process. **174**, 109067 (2022)
5. Li, D., Ho, S.C.M., Song, G., Ren, L., Li, H.: A review of damage detection methods for wind turbine blades. Smart Mater. Struct. **24**(3), 033001 (2015)
6. Movsessian, A., Cava, D.G., Tcherniak, D.: An artificial neural network methodology for damage detection: demonstration on an operating wind turbine blade. Mech. Syst. Signal Process. **159**, 107766 (2021)
7. Mylonas, C., Abdallah, I., Chatzi, E.: Conditional variational autoencoders for probabilistic wind turbine blade fatigue estimation using supervisory, control, and data acquisition data. Wind Energy **24**(10), 1122–1139 (2021)
8. Ou, Y., Chatzi, E.N., Dertimanis, V.K., Spiridonakos, M.D.: Vibration-based experimental damage detection of a small-scale wind turbine blade. Struct. Health Monit. **16**(1), 79–96 (2017)
9. REN 21 Steering Committee: Renewables 2022 global status report (2022)
10. Tcherniak, D., Mølgaard, L.: Vibration-based SHM system: application to wind turbine blades. J. Phys. Conf. Ser. **628** (2015). https://doi.org/10.1088/1742-6596/628/1/012072
11. Ulriksen, M.D., Tcherniak, D., Damkilde, L.: Damage detection in an operating vestas v27 wind turbine blade by use of outlier analysis. In: 2015 IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems (EESMS) Proceedings, pp. 50–55. IEEE (2015)