



RESEARCH ARTICLE 

# Machine learning and uncertainty quantification to track and monitor natural frequencies in vibration-based SHM applied to offshore wind turbines

Maximillian Weil<sup>1</sup> , Carlos Sastre Jurado<sup>1,2</sup>, Wout Weijtjens<sup>1</sup>  and Christof Devriendt<sup>1</sup>

<sup>1</sup>OWI-Lab, AVRГ, Vrije Universiteit Brussel, Pleinlaan 2, 1050 Brussels, Belgium

<sup>2</sup>Geotechnical Laboratory, Ghent University, Technologiepark 68, 9052 Zwijnaarde, Belgium

**Corresponding author:** Maximillian Weil; Email: [maximillian.weil@vub.be](mailto:maximillian.weil@vub.be)

**Received:** 22 April 2024; **Revised:** 13 November 2024; **Accepted:** 12 December 2024


**Keywords:** structural health monitoring; natural frequency tracking; machine learning; uncertainty quantification; offshore wind turbines

## Abstract

Vibration-based structural health monitoring (SHM) of (large) infrastructure through operational modal analysis (OMA) is a commonly adopted strategy. This is typically a four-step process, comprising estimation, tracking, data normalization, and decision-making. These steps are essential to ensure structural modes are correctly identified, and results are normalized for environmental and operational variability (EOV). Other challenges, such as nonstructural modes in the OMA, for example, rotor harmonics in (offshore) wind turbines (OWTs), further complicate the process. Typically, these four steps are considered independently, making the method simple and robust, but rather limited in challenging applications, such as OWTs. Therefore, this study aims to combine tracking, data normalization, and decision-making through a single machine learning (ML) model. The presented SHM framework starts by identifying a “healthy” training dataset, representative of all relevant EOV, for all structural modes. Subsequently, operational and weather data are used for feature selection and a comparative analysis of ML models, leading to the selection of tree-based learners for natural frequency prediction. Uncertainty quantification (UQ) is introduced to identify out-of-distribution instances, crucial to guarantee low modeling error and ensure only high-fidelity structural modes are tracked. This study uses virtual ensembles for UQ through the variance between multiple truncated submodel predictions. Practical application to monopile-supported OWT data demonstrates the tracking abilities, separating structural modes from rotor dynamics. Control charts show improved decision-making compared to traditional reference-based methods. A synthetic dataset further confirms the approach’s robustness in identifying relevant natural frequency shifts. This study presents a comprehensive data-driven approach for vibration-based SHM.

## Impact Statement

This paper introduces a novel approach to vibration-based structural health monitoring (SHM). By integrating machine learning (ML) and uncertainty quantification (UQ), the study addresses significant challenges in the field, notably environmental and operational variability (EOV) and reliable observability of structural modes for offshore wind turbines (OWTs). The core innovation lies in the deeper embedding of ML models, trained on “healthy” data, and UQ, for improved mode tracking, through out-of-distribution detection, and EOV

 This research article was awarded an Open Materials badge for transparent practices. See the Data Availability Statement for details.

© The Author(s), 2025. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited.

normalization. This dual approach enhances the reliability and accuracy of SHM systems and enables distinguishing structural changes from operational noise. The methodology provides a more reliable and automated tool for decision-making in SHM strategies, especially for OWT operators.

## 1. Introduction

Vibration-based structural health monitoring (SHM) has emerged as a pivotal method for ensuring the structural integrity and optimizing the lifespan of various structures, ranging from buildings to bridges and offshore wind turbines (OWTs; Doebling et al., 1998; Peeters et al., 2001; Salawu, 1997; Weijtjens et al., 2016). Vibration-based SHM fundamentally relies on the observation and analysis of a structure's vibration characteristics, in this case, its modal parameters, which include natural frequencies, mode shapes, and damping ratios. These modal parameters, estimated through an operational modal analysis (OMA), are key indicators of a structure's health, as changes in them can imply damage or degradation (Magalhães et al., 2012). Numerous studies have linked frequency shifts in key structural modes to specific damage types, including scour depth in OWTs, underscoring the practical significance of such monitoring (Fallais et al., 2022; Prendergast et al., 2015; Weijtjens and Devriendt, 2017; Weil et al., 2023; Weinert et al., 2015).

Traditionally, vibration-based SHM that relies on modal parameter estimation (MPE) from an OMA consists of four consecutive steps (Bel-Hadj et al., 2024; Devriendt et al., 2014; Magalhães et al., 2012):

1. **Automated OMA:** Automatically extracts the modal parameters, such as frequency and damping ratios from ambient vibration data, as detailed in the study by Devriendt et al. (2014).
2. **Tracking:** Labels the identified structural modes within the large number of automated OMA results over time.
3. **Data normalization:** Adjusts tracked modal parameters, accounting for environmental and operational variability (EOV), such as temperature changes, enhancing accuracy, and reducing hidden damage states, often involving ML methods.
4. **Decision-making:** Analyzes normalized modal parameters against a healthy baseline to detect significant changes or anomalies, indicating potential damage.

This article builds upon these steps while revising several concepts. In particular, it explores how information from the data normalization process (step 3) can be integrated to improve both the tracking (step 2) and the decision-making (step 4). The goal of this research is to automate and enhance steps 2–4 by integrating recent data-driven methods, enabling vibration-based SHM to scale across numerous structures without requiring continuous expert oversight of all data.

The tracking step is usually performed through a reference-based strategy, involving manually selecting frequency limits for the modes of interest on an initialization dataset (Magalhães et al., 2012; Martins et al., 2014; Oliveira et al., 2018; Ubertini et al., 2016; Verboven et al., 2002). However, for OWTs, a challenge arises from the significant influence of EOV on the structural response. The EOV encompasses various factors, such as varying wind and wave conditions, operational states (parked, idling, rated, etc.), and temperature fluctuations, which can introduce significant variability in the measured natural frequencies (Farrar and Worden, 2012). Because of the large EOV of natural frequencies, a fixed set of selected limits are not adequate for reliable tracking of the structural modes over the entire operational window. Therefore, Weijtjens et al. (2016) suggested tracking the physical modes on a case-by-case basis, where different reference values are set for the different operational states (or cases) of the OWT. However, this method is subjective and requires time-consuming manual reference settings for each operational state. Alternatively, Pereira et al. (2023) introduced an automatic reference determination method using an extended modal assurance criterion (MACX). However, MACX requires mode shape estimation, impossible for single-sensor setups, as considered in Section 2.

Data normalization is crucial for reliable damage detection as it accounts for the EOV of the modal parameters, which can otherwise obscure variations caused by structural anomalies. Typically, this requires additional features to describe the environmental and operational conditions during which the OMA result is collected. For instance, in the context of OWTs, relevant features are typically obtained from the turbine's supervisory control and data acquisition (SCADA) data and meteorological (METEO) data (Weijtjens et al., 2016). Once these data are collected, various models can be used for the actual normalization, often linear regression (LR) suffices (Oliveira et al., 2018; Peeters and De Roeck, 2001b; Winkler et al., 2023). However, in particular for OWTs, the EOV can be nonlinear and discontinuous. The best example is the strong difference in structural dynamics between a parked and operational wind turbine, leading to the need for an ensemble of LR models (Weijtjens et al., 2016). To mitigate this, ML techniques, such as autoencoders (Weil et al., 2022), affinity propagation (Häckell and Rolfes, 2013), and ML regression models (Xiang et al., 2024), have been used for EOV normalization of natural frequencies. In this paper, several ML models are compared to predict natural frequencies under varying operational and environmental conditions for application to offshore wind.

In addition, recent experience with OWT data shows that several structural modes cannot be reliably tracked over the entire range of EOV (Oliveira et al., 2018; Xiang et al., 2024), for example, due to the presence of a rotor harmonic (cf. Section 2.1), an unfavorable sensor position, limited modal excitation or high values of damping. Often this concern is mitigated in the decision-making step by only considering modes that are available over the entire EOV window or by only considering the performance in a particular operational state, such as parked. However, this can lead to a loss of information as fewer modes will be considered for fewer timestamps.

It is proposed to leverage the ML model required for the data normalization step for model-based tracking. However, this leads to a chicken-and-egg problem as training an ML model requires a training dataset encompassing all relevant EOV, but creating this dataset through the smart tracking methodology requires an ML model. For this reason, an initialization step is introduced to construct an initial structural modes dataset through unsupervised clustering, subsequently corrected using physical knowledge of the structure and a human-in-the-loop. This initial structural modes dataset serves as the foundation for training various ML regression models to predict each structural mode for tracking and data normalization.

This research leverages the strengths of ML to better understand the complex interactions between SCADA data, METEO conditions, and modal parameters and bake engineering judgment into the selection of the training set. As such, after a human-in-the-loop initialization, a more reliable tracking of structural modes is achieved. Furthermore, recent advances in UQ within ML models (Maddox et al., 2019; Malinin, 2019; Mondal, 2021; Poggi et al., 2020) have been employed for out-of-distribution (OOD) detection. This approach ensures that the tracking is confined to periods where the operational and environmental conditions of the OWT are within the scope of the training data, thereby excluding modes that might be affected by non-structural components such as rotor dynamics. Consequently, this method enhances the coverage and reliability of tracking across all relevant structural modes, ensuring that the initial expert knowledge is effectively transferred to the model-based tracking process.

This paper is structured to explore the potential of ML and UQ to automate and improve vibration-based SHM strategies. Following this introduction, Section 2 presents the instrumented OWT setup and other data sources, encapsulating the available data and observed challenges. Subsequently, Section 3 gives an overview of the methodology separated into three parts. First, the creation of the initial structural modes dataset is discussed in Section 3.1. Second, these data, combined with operational and weather data, are used for feature selection, ML model training, comparison, and selection, as detailed in Section 3.2, along with a detailed overview of the UQ principles. Third, Section 3.3 discusses the implementation of ML and UQ in the SHM process for smart tracking, data normalization, and decision-making. Following the methodology, Section 4 details the results obtained by applying the novel SHM strategy to both empirical and synthetic data from an instrumented OWT. In Section 5, observed limitations and potential future works are identified. Finally, Section 6 gives the conclusion and

encapsulates the findings and their implications for SHM practices, with a particular emphasis on their application to OWTs.

## 2. Instrumented OWT

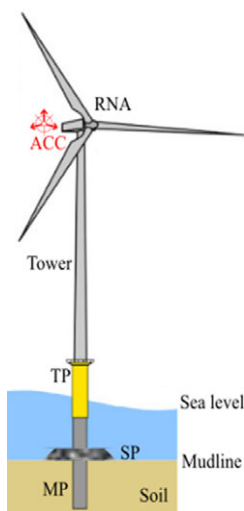
The developed vibration-based SHM methodology is applied to an instrumented monopile-supported OWT in the Belgian North Sea for a proof of concept. The OWT features a high-end accelerometer (ACC) placed in the rotor-nacelle assembly (RNA), as shown in Figure 1, crucial to capture high-fidelity vibration data that are key to accurate natural frequency analysis.

A comprehensive 15-month dataset comprising both SCADA and automated OMA results serves as the foundation for this study. The SCADA data detail the OWT's operational conditions, essential for EOVS normalization. Further, external METEO data sources, obtained from the “Meetnet Vlaamse Banken” (MVBC) (Flemish government, Agency for Maritime Services and Coast, 2023), complement the dataset with additional weather measurements, enhancing the environmental context for the SHM framework. Data synchronization poses a unique challenge due to varying sampling intervals across data sources. To align the 10-minute interval SCADA data with the 30-minute interval METEO data, a linear interpolation approach is employed for its simplicity and effectiveness in maintaining data continuity. The ACC vibration data are used to calculate the natural frequencies every 10 minutes through automated OMA as described in Section 3.1.1.

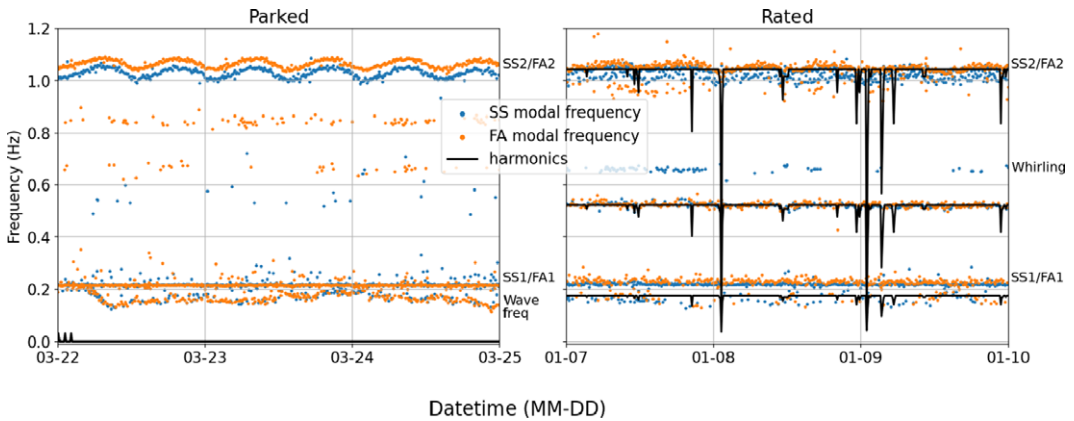
### 2.1. Observability of structural modes

The damage-sensitive structural modes that are crucial for SHM purposes are not always observable within the results of the automated OMA applied to an OWT. Under certain environmental and operational conditions, these structural modes may be obscured by other structural modes, such as whirling modes (Hansen, 2007), or nonstructural modes, such as wave frequencies or rotor harmonics from rotor dynamics (Weijtjens et al., 2014a). Additionally, it is possible that these modes are insufficiently excited under specific conditions, making them difficult to detect. In such cases, the OMA results for these damage-sensitive structural modes are considered unreliable for SHM decision-making and are therefore excluded from the proposed monitoring strategy.

This is illustrated below through the example of second-order modes in modern OWTs, where interactions occur between structural modes and those induced by the rotor, such as the 6P harmonic.



**Figure 1.** Monopile-supported OWT monitoring setup with one ACC at tower top.



**Figure 2.** SS (blue) and FA (orange) detected modes through automated OMA for both the parked (left) and rated (right) operational conditions, highlighting the interference of rotor harmonics for SS2 and FA2.

These harmonic frequencies depend on the rotor's rotational speed expressed in rotations per minute (RPM) and are defined by Equation (1).

$$f_{\text{harmonic}} = P * \frac{\text{RPM}}{60}; \quad P \in [1, 3, 6, 9, \dots]. \quad (1)$$

Figure 2 demonstrates the perturbation of the second-order side-side (SS2) and second-order fore-aft (FA2) modes due to 6P harmonic by comparing periods when the OWT was parked (left) and operational under rated conditions (right). In the parked state, the harmonic mode is at 0 Hz as the rotor is stationary. The SS1 and FA1 modes, visible at approximately 0.2 Hz, exhibit minimal variability, whereas the frequencies primarily below SS1 and FA1 correspond to wave frequencies. The SS2 and FA2 modes, just above 1 Hz, show the expected cyclic behavior due to changing tides for a monopile-supported OWT (Weijtjens et al., 2016).

Under rated conditions, SS1 remains consistent; however, FA1 is observed at a slightly higher frequency when rated and exhibits increased variability as expected from the literature (Song et al., 2023; van Vondelen et al., 2023; Weijtjens et al., 2014b). Additionally, a whirling mode is apparent between 0.6 and 0.8 Hz in the SS direction. The interaction between the 6P harmonic and second-order modes disrupts the expected cyclic pattern. It is, therefore, essential to differentiate periods with and without harmonic influence. Moreover, an automated process with minimal manual intervention is crucial considering the growing volume of monitored structures. In Song et al. (2023), second-order modes are evaluated solely during parked conditions, a state that persists for merely 1 week in an annual monitoring period. Given that second-order modes are more sensitive to some damages, such as scouring (Prendergast et al., 2015), implementing an automated strategy to consistently track these modes with high fidelity is desirable. Moreover, only intermittent evaluation raises questions on the variability with environmental conditions, assuming there are little other data available: Can one truly compare data, for example, from a few parked days in summer with some isolated days in winter?

The interaction between the 6P harmonic and second-order modes underscores the significant influence nonstructural modes can have on the observed structural modes. Harmonic perturbation in OMA has been extensively studied, and recent approaches have attempted to incorporate the harmonics directly into the models to mitigate their impact. Notably, van der Veen et al. (2013) and Greš et al. (2021) have built upon the foundational work of Mohanty and Rixen (2003) by integrating harmonic components into OMA models. However, these methods face challenges in a continuous monitoring context, particularly due to uncertainties in rotor speed data, the presence of slowly varying rotor speeds, and

the highly damped nature of these harmonics, as discussed in a previous study within our research group (Motte et al., 2015).

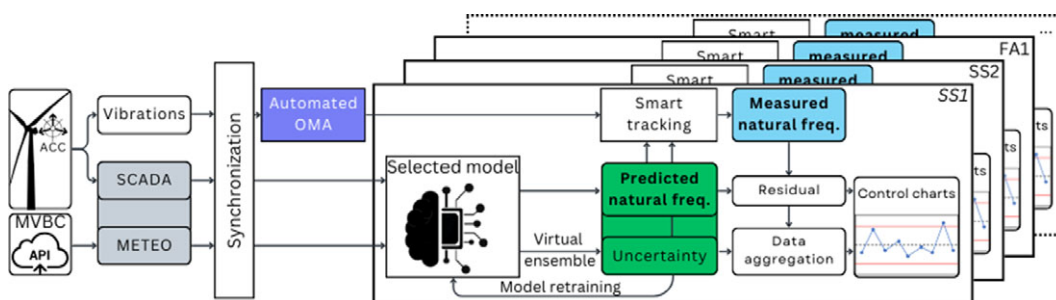
Further advancements were made by van Vondelen et al. (2023), who introduced an extension aimed at better localizing these harmonics. Despite this progress, these solutions remain limited to harmonic perturbations that can be explicitly modeled. However, perturbations in structural modes are not limited to harmonics alone; they can also arise from other factors such as changes in environmental conditions, operational states, or wave-induced forces.

If the structural modes used as damage-sensitive parameters in SHM strategies are not observed without perturbations, this can potentially lead to erroneous conclusions and decision-making based on faulty data. To address these challenges, this research proposes a methodology for reliably tracking structural modes under high-fidelity conditions only, ensuring that these modes are monitored in scenarios where perturbations are minimized. This approach aims to establish a robust SHM framework without the need for exhaustive characterization of every environmental and operational condition.

### 3. Methodology

This section outlines the methodology proposed in this research to automate and enhance the OMA-based SHM. The approach is illustrated through the flowchart in Figure 3, with the main novelty being the smart tracking approach, to automate the tracking of structural modes under high-fidelity conditions only. The methodology starts from the automated OMA, introduced by Devriendt et al. (2014). This three-step process incorporates vibration data preprocessing, MPE, and finally mode selection from the MPE output. Typically, the physical modes resulting from the mode selection are characterized by their natural frequency, damping ratio, cluster size (in the stabilization diagram), and mode shape (in multisensor setups). However, as previously detailed, obtaining reliable modes can be challenging for operational OWTs showing high EOV. Therefore, the new smart tracking method is introduced, involving the following main steps:

1. **Initialization:** On a selected initialization dataset, all the modes resulting from the MPE are stored in a database and used for the initialization process to identify the relevant structural modes. Additionally, engineering judgment is baked into the selection of the training set through a human-in-the-loop process, as detailed in Section 3.1.
2. **ML model preparation:** The identified initial structural modes are used in combination with data describing the environmental and operational conditions to train the ML model, after model and data selection. Section 3.2 describes the ML model preparation and UQ methods, required for the smart tracking process.



**Figure 3.** Flowchart showing the implemented natural frequency predictions combined with measures for uncertainty on the predictions, for enhanced tracking, data normalization and decision-making in the OMA-based SHM workflow.

3. **Smart tracking and decision-making:** The natural frequency predicted by the ML model and the uncertainty of the prediction are used to track the relevant structural modes identified in the initialization. As the ML model is used to make an a priori expected value, which is then used for tracking, we refer to the process as “smart tracking.” This results in a reliable mode tracking as shown in Section 3.3.1. The averaged residual between ML model prediction and automated OMA output for low uncertainty predictions is used as a normalized, damage-sensitive feature in the control charts, as described in Section 3.3.2.

The following paragraphs give a detailed overview of the initialization, the necessary steps for ML model preparation, and of all the steps involved in the SHM methodology.

### 3.1. Initial structural mode detection

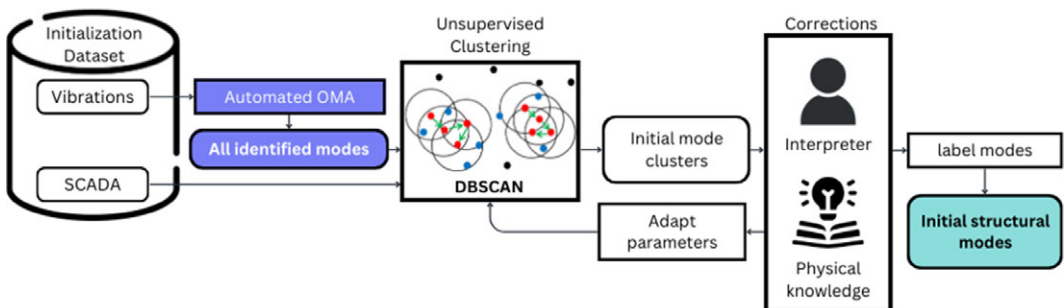
To overcome the rigidity of reference-based tracking when dealing with the EOv of OWT natural frequencies, an alternative model-based tracking methodology is introduced. However, as previously mentioned, a chicken-or-the-egg problem emerges. For smart tracking, a trained ML model is required, which in turn requires an appropriate training dataset, typically only available after tracking. In light of these challenges, an initialization process, as illustrated in Figure 4, is formulated and implemented. This workflow uses the density-based spatial clustering of applications with noise (DBSCAN) algorithm (Ester et al., 1996) to cluster all modes, identified by the automated OMA, across multiple dimensions and show the evolution of modal parameters over time for an initialization dataset. Next, leveraging system physical knowledge and an interpreter’s expertise, the parameters of DBSCAN are fine-tuned in a human-in-the-loop approach (Wu et al., 2022) to achieve optimal cluster formation, as detailed in Section 3.1.2.

Interference from extraneous modes may necessitate manual data selection within the DBSCAN identified clusters. This step is critical to guarantee that the initial set of structural modes contains only high-quality data based on engineering judgment, suitable for effective ML model training. Finally, the resulting clusters are labeled as the initial structural modes, serving as training data for ML models and, in turn, used for long-term mode tracking. This process of cleaning up the data and highlighting the automatically identified clusters of interest replaces the traditional manual setting of reference-based tracking. Note that this action serves a dual purpose as it provides the training dataset for the data normalization.

The following paragraphs give a detailed overview of all steps involved in the generation process of the initial structural modes.

#### 3.1.1. Automated OMA

Modal parameters are essential for vibration-based SHM and are determined through the automated OMA, a process that is executed on discrete segments of ACC data, specifically 10-minute intervals for



**Figure 4.** Flowchart illustrating the construction of the initial physical modes used for training the normalization and tracking models, as shown in Figure 5.

this study. Multiple algorithms exist for MPE, including frequency domain decomposition (FDD), least squares complex frequency domain (LSCF), and covariance-driven stochastic subspace identification (SSI-COV; Magalhães and Álvaro Cunha, 2011; Reynders, 2012). For this study, the methodology developed by Devriendt et al. (2014), utilizing the LSCF algorithm for MPE, is implemented to identify all modes from the vibrations in the initialization dataset (cf. Figure 4). As previously mentioned, all clusters identified in the MPE are kept for delayed DBSCAN tracking, as detailed in the subsequent section. This results in the natural frequency, damping ratio, and cluster size for every cluster of the stabilization diagram of 10-minute vibration data. Mode shapes are not considered, as the method applies to a single-sensor SHM setup. The calculated modal parameters form the basis for the next section, consisting of clustering the structural modes and removing spurious modes and non-structural effects.

### 3.1.2. DBSCAN-based unsupervised clustering

The initialization process builds an initial structural modes dataset through unsupervised clustering. As previously mentioned, this is achieved through the DBSCAN algorithm (Ester et al., 1996) on an initialization dataset. The relevant clusters are then used to train ML models for the smart tracking methodology detailed in Section 3.3.1. The novel insight in this method is to add the time dimension to allow for slow variations of the natural frequency over time. This is done by adding the time difference between the different identified modal parameters as an additional feature to the DBSCAN algorithm. The clustering can thus be done on the frequency and time difference only, as done in a previous SHM research on a high-voltage transition tower (Bel-Hadj et al., 2024). However, due to EOv of natural frequencies in OWTs, this process required corrections to the cluster through physical knowledge of the structure and the supervision of an interpreter, as detailed in Section 3.1.3.

Before clustering, the search space must be transformed to allow for the correct clustering of the structural modes. This is done by scaling every dimension to transform the hyperspace on which the DBSCAN algorithm clusters the data. In this case, the input dimensions are the frequency ( $f$ ), cluster size from the automated OMA (size), and time ( $t$ ). Although new scaling parameters ( $\gamma$ , to scale the parameter  $\bullet$ ) are introduced, this enables the DBSCAN algorithm to accurately cluster the modes. Together with the inherent DBSCAN parameters:  $\epsilon$  (maximum distance between two points for them to be considered neighbors) and  $\text{minPts}$  (minimum number of neighbors a point needs to have to be considered a core point), these new parameters should be optimized through the corrections process detailed below. However, this optimization only needs to be run once for a population of similar structures, such as OWTs in an offshore wind farm, as the same parameters can be used. This methodology is implemented in a separate Python package (Weil, 2024c). The previous work on transmission towers contains an in-depth description of this approach (Bel-Hadj et al., 2024).

### 3.1.3. Cluster corrections

Following the initial mode clustering, corrections are made to ensure accurate mode labeling under the supervision of an interpreter, in a human-in-the-loop fashion, to ensure engineering judgment is added to the selection process. This involves both an optimization of the DBSCAN parameters and manual correction for remaining confounded modes. By training ML models on this curated data and using OOD detection through UQ (cf. Section 3.2.4), the engineering judgment is automatically transferred to new data during the tracking process, as detailed in Section 3.3.1. This approach facilitates the optimization of the clustering algorithm's parameters and ensures that the resulting clusters exclude obfuscated modes.

*Interpreter and physical knowledge:* Iterative adjustments are made to both the added parameters for the hyperspace scaling ( $\gamma$ ) and the inherent DBSCAN parameters ( $\epsilon$  and  $\text{minPts}$ ). Alternatively, the optimization of the latter can be automated through the use of hierarchical DBSCAN (McInnes and Healy, 2017), but this is outside the scope of the current research. The obtained optimal parameters are given for the FA and SS direction in Table 1. Using these parameters for the clustering yields a set of initial structural

**Table 1.** DBSCAN parameters for unsupervised clustering in FA and SS directions

Parameter	SS direction	FA direction
$\epsilon$	4	5
minPts	100	100
$\gamma_f$	200	200
$\gamma_{size}$	2	2
$\gamma_t$	$\frac{1}{5000}$	$\frac{1}{5000}$

mode clusters. However, these can still have non-structural components. Therefore a manual data selection based on engineering judgment is performed on these clusters.

*Manual data selection:* There are instances where modes resulting from the automated OMA cannot be trusted, for example, due to nonstructural mode interference. In these scenarios, only the data representing the mode's expected behavior, as informed by prior knowledge of the engineer, can be manually isolated from the clusters. Such a process can readily be implanted with a graphical user interface, reducing the process to the engineer encircling the desirable results on the initialization dataset only. An implementation in Python is made available at Weil (2024a). When sufficient data are gathered to represent all relevant EOV, the data are used to train the ML models for smart tracking, leveraging OOD and UQ of the ML predictions to automatically remove modes with nonstructural interference on new arriving data (cf. Section 3.3.1).

*Reproducibility:* While the human-in-the-loop approach ensures that expert engineering judgment is incorporated into the mode selection process, it introduces a challenge to reproducibility. Although the DBSCAN parameters can easily be transferred, different interpreters may make slightly different decisions when selecting data, potentially leading to variations in the final outcome. Therefore, different solutions are proposed to enhance reproducibility:

- Detailed guidelines and criteria for decision-making based on the monitored structure should be established and documented, ensuring consistency across different users. In future research, this could be incorporated within the graphical user interface for manual data selection. In the case of the current research, different rules are outlined below:
  - The clusters resulting from the DBSCAN implementation are used as an initial set from which to select the structural modes.
  - Only modes identified with a damping ratio lower than 5% are considered (Van Der Tempel, 2006). This threshold can be set higher for FA modes in the rated operational state (Devriendt and Weijtjens, 2017).
  - Only modes identified with a cluster size higher than 5 are considered (Devriendt et al., 2014).
  - Modes from the rotor dynamics are known to have a one-on-one relationship with the RPM (cf. Equation (1)) and are removed through the Campbell diagram (Jahani et al., 2022).
  - It is known that the natural frequencies of structural modes do not jump abruptly over large frequency ranges. Therefore, these jumps are probably caused by perturbations in the OMA result and are not considered (Weijtjens et al., 2016).
  - The selection performed on the SS2 mode is made only when the mode exhibits the expected dependency to the tidal levels for monopile-supported OWTs (Weijtjens et al., 2016).
- Maintaining a detailed log of the human-in-the-loop adjustments, including the specific parameters and data selections made, can help in tracking decisions and understanding their impact on the final results.
  - In the current research, only corrections based on damping ratio and cluster size are made on the clusters identified for the FA1 and SS1 mode, while only five time periods exhibiting the expected dependency to the tidal levels and have no perturbation from extraneous modes are

used as the final SS2 cluster. The data selected for SS2 and the operational states represented in the dataset are later described in the charts of Figure 8.

### 3.2. ML model for natural frequency prediction under changing conditions

This section focuses on the development and implementation of ML models to predict the natural frequencies of the different structural modes under varying operational and environmental conditions, as shown in Equation (2). The ML models are then used to split the natural frequencies into a prediction based on environmental and operational data and a residual, later used as damage-sensitive feature for SHM.

$$f_m(t) = \hat{f}_m(\mathbf{x}(t)) + \epsilon_m(t) \quad \forall m. \quad (2)$$

In this equation,  $f_m(t)$  represents the natural frequency for mode  $m$  in a time interval  $t$ ,  $\hat{f}_m(\mathbf{x}(t))$  denotes the ML model's prediction based on input features  $\mathbf{x}(t)$ , and  $\epsilon_m(t)$  is the residual. The initial structural modes dataset, established in Section 3.1, is split into a train (80%) and test (20%) dataset through the train–test–split method (Géron, 2019) and serves as the foundation for training the various ML regression models, as illustrated in Figure 5. The following paragraphs detail the processes involved in data preprocessing and synchronization, as well as the training, evaluation, and selection of optimal ML models, and finally, the implementation of the models for natural frequency prediction and UQ.

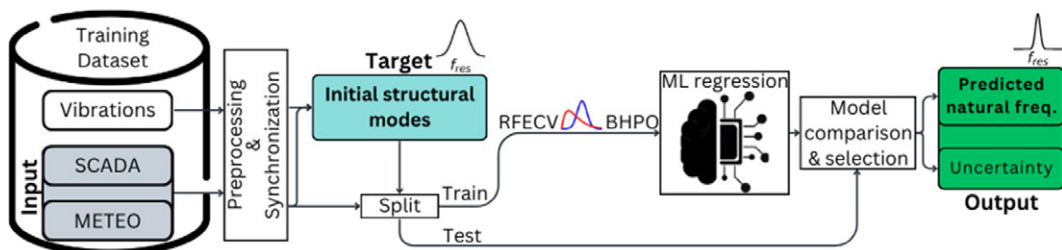
#### 3.2.1. Data preprocessing and synchronization

In the ML workflow, the data preparation step always precedes the ML model development (Géron, 2019). In this case, the data preparation step involves preprocessing and synchronizing the data collected from the different sources.

The preprocessing step is crucial in the ML workflow to ensure the model interprets the data accurately (Al-jabery et al., 2020). The preprocessing involves encoding circular data, including angles with a  $360^\circ$  range, such as yaw and wind direction, as well as temporal data (e.g., time of day, month of the year). The challenge with these data is that, for example, an angle of  $350^\circ$  is closer to  $10^\circ$  than to an angle of  $300^\circ$ . To enable ML models to understand circular relationships, the sine and cosine transformations are applied to encode the circular data (Van Wyk, 2022). In the studied case, this includes the wind direction and the yaw angle or orientation of the OWT nacelle.

Additionally, preprocessing involves data scaling or normalization, such as standardization or min–max normalization, to prevent the magnitude of the data from skewing the ML predictions. However, this step is model-dependent; tree-based learners, for instance, do not require data scaling due to their inherent handling of feature scales (Hehn et al., 2020). The data normalization used in this research is the min–max normalization, performed on the selected data during the model comparison on the input of certain models. The used minimum and maximum values are given in Appendix A.

Data synchronization is then undertaken to align input and target data across corresponding time-stamps and interpolate data with different sampling frequencies, resulting in a consolidated dataset ready for ML training and testing.



**Figure 5.** Flowchart illustrating the ML model construction and selection process for smart tracking and EOv normalization.

### 3.2.2. Feature selection

The next step in the ML workflow is feature selection, determining the most relevant features to predict the natural frequencies. As stated in Guyon and Elisseeff (2003), the objective of the feature selection process is three-fold: to improve the prediction performance of the model, to make the model faster and more cost-effective, and to better understand the parameters influencing the natural frequencies.

Feature selection is performed using recursive feature elimination with cross-validation (RFECV) on the training data for automatic tuning of the number of features selected. RFECV consists of fitting a model and recursively removing the weakest feature, according to the model's feature importance, until a specified amount of features are obtained (Guyon et al., 2002; Sanz et al., 2018). The addition of cross-validation (CV) identifies the best number of features within a k-fold CV loop.

In this case, the XGBoost regression model (Chen and Guestrin, 2016) is selected for its execution speed, good predictive performance, and its built-in feature importance metric. The RFECV is implemented using the scikit-learn Python package (Pedregosa et al., 2011), with XGBRegressor as the estimator from the XGBoost library with standard hyperparameters. The RFECV is configured with the following parameters:  $\text{step} = 1$ ,  $\text{cv} = 5$ ,  $\text{stepscoring} = \text{"neg\_mean\_squared\_error"}$ ,  $\text{min\_features\_to\_select} = 1$ . The results of applying this RFECV to the selected SS2 data are summarized in Table 2, where the last column indicates whether each feature is selected.

Table 2 provides an overview of the feature selection from METEO and SCADA data, used as ML model inputs for predicting natural frequencies of OWTs (i.e., the targets). Only input variables with over

**Table 2.** Overview of the available data with the source of the data, the measurement units, and availability

	Source	Variable	Unit	Availability (%)	Selected
Input	METEO	Wave height	mm	100	Yes
		Tide-TAW	m	50 → 99 <sup>a</sup>	Yes
		Sea water temperature	°C	94	Yes
		Air pressure	Pa	99	Yes
		10% highest waves	mm	100	Yes
		Average wave period	s	100	No
		Air temperature	°C	100	No
		Height waves (period >10 s)	mm	100	No
		Average wind direction	°	100	No
		Max 3 s wind gust (10 m)	mm	99	No
		Wind speed (10 m)	m/s	99	No
		Relative humidity	%	78	No
	SCADA	1% wave height	mm	13	No
		Pitch	°	97	Yes
		Power	MW	97	Yes
		Wind speed	m/s	97	Yes
		Yaw	°	97	Yes
		Wind direction	°	97	Yes
		RPM	min <sup>-1</sup>	97	Yes
Target	IoT ACC	SS1	Hz	87	—
		SS2 (parked)	Hz	4 <sup>b</sup>	—
		SS2 (selected)	Hz	7.4	—
		FA1	Hz	41	—

Note: The last column states if the features are selected by the RFECV.

<sup>a</sup>Pulled from a second source to increase availability.

<sup>b</sup>Even though 4% seems enough to detect slow evolving damages in SHM, this is concentrated in two regions of data for the entire monitoring period, sometimes leading to months without accurate data to rely on.

90% availability are considered to ensure adequate coverage of the ML model. However, the tidal-level data were initially available only 50% of the time, this is problematic as tidal level is known to influence higher order structural modes of OWTs (Weijtjens et al., 2016). By supplementing this with data from another nearby weather station, availability was increased to 99%.

The data selected by RFECV include parameters that are known to affect the natural frequencies of OWTs. First, all SCADA features are selected. Parameters such as pitch, power, RPM, and wind speed describe the turbine's operational condition, which directly influences its dynamics (Jahani et al., 2022). Yaw and wind direction can impact dynamics, particularly in cases of misalignment (Song et al., 2023). Not all features from the weather station are selected; however, the tidal level, wave height, and 10% highest waves are included due to their significant influence on modal frequencies. Sea water temperature affects modal parameters, contributing to seasonal variability, while air pressure influences the interaction between wind and structure, affecting dynamics (Weijtjens et al., 2016). Among the parameters not selected, many are correlated with the selected features. However, average wave period, identified as relevant by Xiang et al. (2024), is not included in the selected features in this case.

The SS1 and FA1 modes show high (87%) and moderate (41%) availability, but the SS2 mode has high interference from nonstructural modes and could only be considered when the OWT is parked. But this translates into low availability (4%) concentrated in two regions of the year, leading to long periods without reliable data. Therefore, a manual data selection, as detailed in Section 3.1.3, is implemented to train a model on structural modal data only for predicting the SS2 frequency. The ML model can then be used for smart tracking, normalization, and anomaly detection.

### 3.2.3. Model optimization, comparison, and selection

Once the features are selected, a model comparison is performed to guide the ML model selection. Model performance is evaluated on the test data using two key metrics: the mean squared error (MSE) and the coefficient of determination ( $R^2$ ), as defined in Equations (3) and (4), respectively. These metrics provide insights into the accuracy and goodness of fit for each model.

$$\text{MSE}_m = \frac{1}{n} \sum_{t=1}^n (f_m(t) - \hat{f}_m(\mathbf{x}(t)))^2 \quad \forall m, \quad (3)$$

$$R_m^2 = 1 - \frac{\sum_{t=1}^n (f_m(t) - \hat{f}_m(\mathbf{x}(t)))^2}{\sum_{t=1}^n (f_m(t) - \bar{f}_m)^2} \quad \forall m. \quad (4)$$

In these equations,  $n$  denotes the number of samples considered,  $f_m(t)$  is the target value, the natural frequency of mode  $m$  in this case, for timestamp  $t$ , while  $\hat{f}_m(\mathbf{x}(t))$  represents the ML model prediction based on input  $\mathbf{x}(t)$ , and  $\bar{f}_m$  the average value of  $f_m(t)$  over the  $n$  considered samples. For this study, eight ML models, categorized into three distinct types, are considered to provide a comprehensive comparison of their predictive capabilities.

LR models are first considered, as these were previously used by Magalhães and Álvaro Cunha (2011) and Peeters and De Roeck (2001a). Both a traditional and multiple linear regression (MLR) are implemented. In the case of MLR, one LR is trained for each operational condition of the OWT (e.g., parked, idling, etc.) based on SCADA data (Weijtjens et al., 2016). The MLR extension reduces MSE and increases  $R^2$  for all three modes considered, when compared to LR, as shown in Table 3. However, it requires manually splitting the data according to operational conditions, which other ML models do not require. Despite their limitations, LRs are useful as benchmarks for more advanced ML models due to their simplicity and interpretability.

Next, tree-based learners are implemented, including random forest (RF), which uses the Bagging principle (Altman and Krzywinski, 2017), and gradient boosting methods (Friedman, 2001) such as eXtreme gradient boosting (XGB; Chen and Guestrin, 2016) and categorical boosting (CB; Prokhorenkova

**Table 3.** Results obtained for the model comparison for OWT natural frequency predictions on the test data after BHPO. The metric of the best performing model is highlighted in the table as bold-faced.

		SS1		SS2		FA1	
Type	Model	MSE	$R^2$	MSE	$R^2$	MSE	$R^2$
Ref. values: $\bar{f}_{m_r}$		9.15e−06	0.00	2.01e−04	0.00	6.05e−05	0.00
Lin.	LR	8.62e−06	0.06	4.72e−05	0.77	5.16e−05	0.15
	MLR	8.00e−06	0.11	4.44e−05	0.78	<b>4.94e−05</b>	0.16
Tree-based	RF	<b>7.88e−06</b>	<b>0.14</b>	<b>3.72e−05</b>	<b>0.81</b>	4.99e−05	<b>0.17</b>
	XGB	7.96e−06	0.13	3.85e−05	<b>0.81</b>	5.03e−05	<b>0.17</b>
	CB	<b>7.88e−06</b>	<b>0.14</b>	3.73e−05	<b>0.81</b>	5.01e−05	<b>0.17</b>
ANN	1 hidden	8.12e−06	0.11	3.90e−05	0.80	5.05e−05	0.16
	2 hidden	8.03e−06	0.12	3.95e−05	0.80	5.06e−05	0.16
	3 hidden	7.98e−06	0.13	3.95e−05	0.80	5.05e−05	0.16

et al., 2018). These models capture complex, nonlinear relationships in the data, with XGB focusing on regularization, efficient tree-building, and pruning to enhance performance, while CB emphasizes efficiency and accuracy through ordered boosting and symmetric trees.

Finally, artificial neural networks (ANNs) with one, two, and three hidden layers are explored for predicting natural frequencies. Data and target normalization are applied to improve training efficiency and model performance (Arnekvist et al., 2020).

Hyperparameter optimization is crucial for enhancing the performance of tree-based learners and ANNs. Bayesian hyperParameter optimization (BHPO) as implemented in the Hyperopt Python package (Bergstra et al., 2015) is used to systematically fine-tune models, aiming to achieve the lowest possible MSE on the selected data. A full overview of optimized parameters is provided in Appendix B.

The optimized models' performances, based on test data predictions and targets, are detailed in Table 3.

Table 3 presents a comparative overview of the different ML models applied to predict monopile-supported OWT natural frequencies SS1, SS2, and FA1. First a reference value of the MSE and  $R^2$  are given, by calculating them for a hypothetical model that would give the mean natural frequency of the training data as prediction, replacing  $\hat{f}_m(\mathbf{x}(t))$  by  $\bar{f}_{m_r}$  in Equations (3) and (4). Subsequently, MSE and  $R^2$  values are calculated for each model. In general, the  $R^2$  values for the SS2 mode are much higher than for both FA1 and SS1 regardless of the model. This is because of the high EOv of SS2, mainly driven by the tidal level and wave height, which is well caught by the models.

The relatively low  $R^2$  values observed for FA1 and SS1 indicate that the input variables used in the ML models cannot explain much of the variability in these modes. Thus, the low  $R^2$  values for FA1 and SS1 suggest inherently low variability in these modes rather than poor model performance. It is important to note that  $R^2$  serves as an effective metric to compare models predicting the same variable but is less suitable to compare models predicting different variables due to inherent differences in the variability of the data to predict.

This higher variability captured by the model when compared to simply predicting the mean frequency is also observed by comparing the MSE of the models to the reference value. SS2 shows a greater reduction in MSE compared to FA1 and SS1. SS1 shows the lowest MSE value, but by comparing to the reference value, this can be interpreted as an inherent low variability in the natural frequency of this mode.

When comparing the models, the LR model shows in general a comparatively higher MSE and lower  $R^2$ . Applying MLR greatly improves model performance, especially for SS1, where  $R^2$  almost doubles, and MLR even outperforms the other models in terms of MSE for the FA1 frequency predictions. However, MLR requires the definition of operational cases. Tree-based models and ANNs generally outperform the linear models without prior knowledge of the operational cases, with tree-based models

slightly outperforming ANNs. This aligns with expectations set by Grinsztajn et al. (2022) regarding the efficacy of tree-based algorithms for tabular data. In general, RF and CB perform best on all three natural frequencies. In the end, CB is selected because the UQ methodology, as described in Section 3.2.4, is already implemented in the Catboost Python package (Dorogush et al., 2018).

#### 3.2.4. UQ

Once the models are trained, a prediction of the natural frequencies can be made and used to remove the identified EOV, isolating the residual  $\epsilon_m(t)$  as given in Equation (2). Assuming an additive error model the residual  $\epsilon_m(t)$  can be further decomposed to isolate noise and modeling error components, as shown in Equation (5).

$$\epsilon_m(t) = v_m(t) + \zeta_m(t) + \delta_m(t) \quad \forall m \quad (5)$$

The first term  $v_m(t)$  denotes the noise present in the natural frequency data. The second term  $\zeta_m(t)$  represents the ML modeling error. This consists of EOV in the natural frequency that the model could not adequately predict. Finally, the third term  $\delta_m(t)$  is the damage-sensitive part of the residual, independent of EOV, which remains zero as long as the structure's state is unchanged and gradually increases with anomaly. However, it is not straightforward to dissociate the term  $\delta_m(t)$  from  $v_m(t)$  and  $\zeta_m(t)$ .

The noise component  $v_m(t)$  is assumed to follow a zero-mean Gaussian distribution, allowing for its reduction through signal averaging (Hassan and Anwar, 2010). In prior work (Weil et al., 2023), the frequentist confidence interval (CI; Dekking et al., 2006) was used to determine the number of residuals to average to reduce the influence of the process noise below a certain threshold. However, as stated in Hespanhol et al. (2019), the Bayesian credible intervals (BCI) are a more appropriate measure for this purpose. Through BCIs, it is found that by averaging residuals  $\epsilon_m(t)$  over 1-week timestamps, the noise in  $\epsilon_m(t)$  is reduced to below 1%, increasing the probability of detecting genuine shifts, despite the inherent noise in the natural frequency. However,  $\zeta_m(t)$  and  $\delta_m(t)$  could still vary within this time window. To address this, UQ methods for OOD detection are applied to ensure that the inputs to the ML model are drawn from the training distribution. By focusing only on reliable predictions, the contribution of  $\zeta_m(t)$  to the residual is minimized, as later shown in Section 3.3.2. Consequently, if a shift in the averaged residual is detected during periods of low uncertainty, it can be attributed to  $\delta_m(t)$ , indicating the detection of an anomaly.

This strategy requires incorporating UQ into the predictions. However, deterministic point predictors, like those discussed in Section 3.2.3, while straightforward to train and implement with strong predictive capabilities, do not provide inherent UQ. Fortunately, recent advancements in UQ methodologies now enable uncertainty estimation for both tree-based models (Malinin et al., 2020; Mondal, 2021) and ANNs (Gawlikowski et al., 2021; Pearce et al., 2020). These techniques allow for more robust predictions by quantifying uncertainty, making them suitable for improving the reliability of the proposed SHM framework.

Uncertainty can be divided into either aleatoric, relating to variability which is due to inherently random effects (Hüllermeier and Waegeman, 2021), or epistemic, stemming from incomplete model knowledge (Bastani and Alur, 2024). Aleatoric or data uncertainty arises from physical phenomena that are random by nature. Therefore, the aleatoric uncertainty cannot be easily quantified and eliminated during modeling calibration (Wang et al., 2023). On the other hand, epistemic, model, or knowledge uncertainty arises from inadequate knowledge of the model to explain the data from inputs from regions either far from the training data or sparsely covered by it. This uncertainty is reducible by providing more knowledge about the problem to the ML model (e.g., providing more and new training data or optimizing the model's hyperparameters). Various empirical UQ techniques exist, as detailed in Poggi et al. (2020) and Mondal (2021), offering methods to approximate the epistemic uncertainty such as Bayesian neural networks (BNN) and Bayesian ensembles. Because BNNs entail significantly higher computational costs compared to point-wise predictors (Bai and Chandra, 2023), it is decided to explore Bayesian ensembles.

Bayesian ensembles consist of an ensemble or collection of  $n_{BE}$  models with different configurations that all make predictions on the same data. The final prediction is an average of the predictions of all

models, and epistemic uncertainty is quantified by measuring the variance or diversity of these predictions (Chipman et al., 2006). As it has been shown in Section 3.2.3 that tree-based learners outperform ANNs for the analyzed tabular data, this research focuses on Bayesian ensembles constructed from tree-based models for UQ (Chipman et al., 2006; Duan et al., 2020). The construction of  $n_{BE}$  different models of the Bayesian ensemble is detailed by Malinin et al. (2020) for gradient boosted decision trees (GBDTs), using stochastic gradient Langevin dynamics. However, the creation, storage, and prediction of these Bayesian ensembles can be computation- and memory-intensive, as it requires training and storing of the  $n_{BE}$  models.

Therefore, Malinin et al. (2020) offer a practical solution to the computational demands of these Bayesian ensembles through virtual ensembles. By recognizing that GBDTs are inherently an ensemble of individual decision trees, a virtual ensemble can be constructed from a single GBDT model. This virtual ensemble comprises  $n_{BE}$  truncated submodels, each representing a stage in the GBDT's creation process. The final proposed strategy uses the virtual ensemble as implemented in the Catboost Python package (Prokhorenkova, 2020) to quantify the knowledge uncertainty on the frequency predictions from a CB regression model.

### 3.3. Integration of ML models and UQ in OMA-based SHM

This paragraph details the implementation of the ML models for natural frequency predictions combined with associated UQ within the SHM methodology as previously illustrated in Figure 3.

#### 3.3.1. Smart tracking

Traditional mode tracking from OMA often relies on reference thresholds. However, as demonstrated in Xiang et al. (2024), model-based strategies employing RF and XGB models offer an alternative. Instead of setting static upper and lower thresholds on the natural frequency for tracking, only results  $f_i(t)$  that are sufficiently close to the ML model predictions  $\hat{f}_m(\mathbf{x}(t))_m$  are tracked as mode  $m$ . The proposed methodology integrates prediction uncertainties obtained through UQ to assess the validity of the model itself given the prevailing EOV. This translates the tracking strategy into the rules given in Equation (6).

$$U_m(\mathbf{x}(t)) \leq \tau_{u_m} \wedge |\hat{f}_m(\mathbf{x}(t)) - f_i(t)| \leq r_{f_m} \Rightarrow f_i(t) \in m. \quad (6)$$

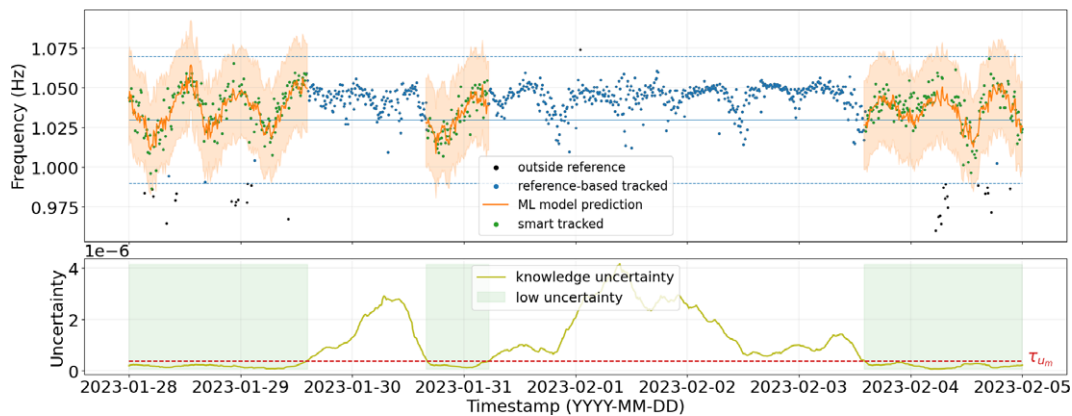
In this equation,  $U_m(\mathbf{x}(t))$  denotes the knowledge uncertainty obtained from the virtual ensemble for input variables  $\mathbf{x}(t)$ . The uncertainty threshold  $\tau_{u_m}$  is chosen as the 90th percentile of uncertainty from the training data  $U_{m_r}$ . The tracking range is set as a  $3\sigma$  band around the error predictions from the training data, or  $r_{f_m} = 3\sigma(\hat{f}_m - f_m)$ .

The concept is illustrated in Figure 6 in comparison to a classic reference-based tracking of a monopile-supported OWT's SS2 mode. The traditional reference-based tracking bounds (blue) make no distinction in data quality and require a wide tracking range to accommodate the EOV. Meanwhile, in smart tracking, only results close to the ML model predictions are used, and only when uncertainty is lower than the uncertainty threshold  $\tau_{u_m}$  (orange). The resulting smart tracked modes (green) are then stored as values for the structural mode for further analysis.

Figure 6 also illustrates the advantageous use of UQ in the smart tracking principle. The SS2 mode is expected to exhibit cyclic variations corresponding to tidal levels. However, discrepancies in this cyclic pattern, indicative of rotor harmonic interference, are flagged by a high uncertainty of the virtual ensemble. Meanwhile, (a simple) reference-based tracking would have still considered these results, potentially obfuscating changes in the natural frequency.

#### 3.3.2. Decision-making

The final step in the vibration-based SHM framework is the decision-making process. In the case of natural frequency monitoring, frequency shifts after normalization are used to determine the structural health of the structure. In the proposed methodology, this is obtained by calculating the residual  $\epsilon_m(t)$  and

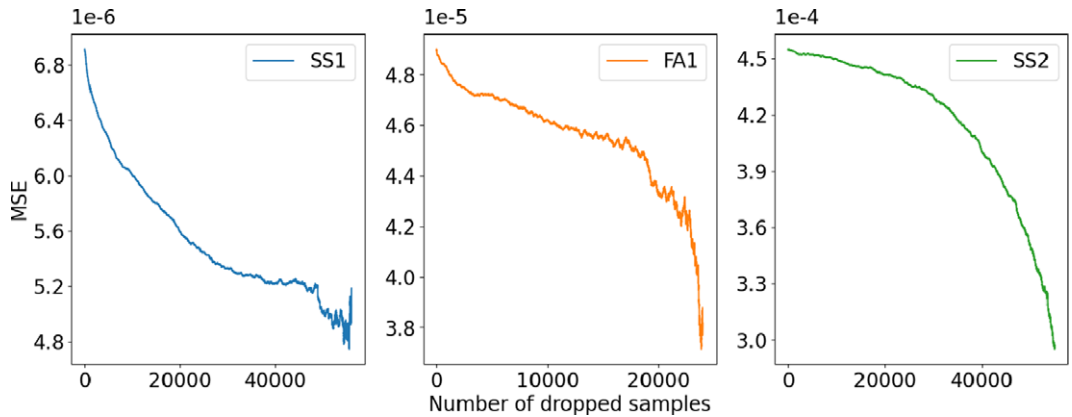


**Figure 6.** Conceptual illustration contrasting smart tracking (based on ML and UQ) with traditional reference-based tracking for an OWT's SS2 mode.

separating parameter  $\delta_m(t)$  as defined in Equation (5). A substantial deviation in  $\delta_m(t)$  for any monitored frequency  $f_m(t)$  should trigger a structural health alert.

However, as stated in Section 3.2.4, the parameter  $\delta_m(t)$  cannot be directly measured. Therefore, an average of the residual  $\epsilon_m(t)$  is considered to remove  $v_m(t)$ , and only low uncertainty periods are considered, to ensure a low  $\xi_m(t)$ . This latter relationship is highlighted by the sparsification error curves (SECs) as proposed by Poggi et al. (2020). SECs are drawn by sequentially removing the prediction for timestamp  $t$  with the highest knowledge uncertainty  $\max(U_m(t))$  from the predictions and then recalculating the MSE (cf. Equation (3)) on the new dataset with one uncertain sample less. The SECs depicted in Figure 7 for modes SS1, SS2, and FA1, illustrate a general MSE reduction with the exclusion of high-uncertainty samples, thereby refining the predictive accuracy and reducing modeling error  $\xi_m(t)$ .

Finally, either a distribution of the residuals over a time period or a control chart of the averaged residuals is drawn to monitor the changes in the residual for low-uncertainty predictions. Both the distribution and control charts are shown in the results of Section 4. The frequency shift that should trigger an alarm depends on the use case. For instance, Prendergast et al. (2015) associated a 5% shift in the SS1 mode with significant scour. Alternatively, it is possible to draw the limits on the frequency changes



**Figure 7.** Sparsification error curves for the physical modes of an OWT, obtained by sequentially removing the most uncertain predictions and showing the effect on the model performance through MSE.

through physics-based models as demonstrated in the study by Weil et al. (2023) for the SS1 mode of a monopile-supported OWT. However, this is currently not incorporated into this research but will be explored in a subsequent study.

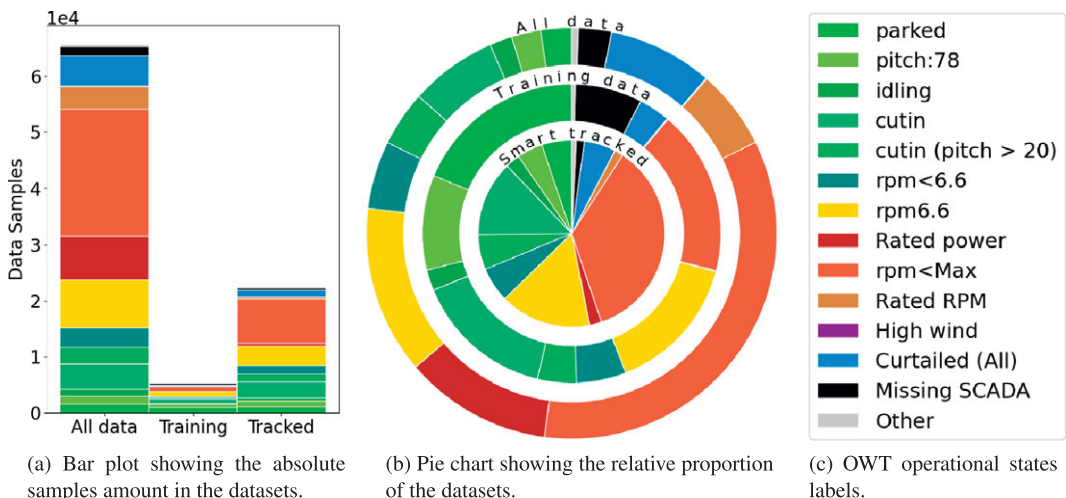
## 4. Results

This section presents the results obtained from applying the novel vibration-based SHM framework as shown in the flowchart of Figure 3 and detailed in Section 3 on both the 15-month OWT dataset described in Section 2 and a synthetic dataset. This evaluation focuses on the ML models' predictive performance and the efficacy of the UQ in enhancing the SHM process.

### 4.1. SS2 smart tracking results

The smart tracking methodology, as detailed in Section 3.3.1, is evaluated by applying it to the SS2 mode, where rotor harmonic interference, particularly at rated RPM, present notable tracking challenges. By applying the smart tracking methodology to the data, it is possible to remove most of the OMA output with rotor harmonic interference, as evidenced by the changes in the data distribution depicted in Figure 8. Figure 8a and 8b, respectively, provides a comprehensive breakdown of the absolute and relative quantities of data throughout the operational states for the total, training and SS2 tracked datasets. Figure 8c gives the legend for the OWT operational states. A detailed description of the operational states is given in Appendix C.

The training data, carefully selected to include only operational cases with anticipated SS2 modal behavior, intentionally omits data from the rated power (dark red) and rated RPM (brown) conditions, known for their harmonic interference. As a result, the smart tracking process significantly diminishes the prevalence of these states in the dataset, from a substantial 20% down to a mere 3%. This reduction affirms that the remaining 97% of the SS2 modal data used for decision-making is predominantly free of harmonic distortion. Moreover, the smart-tracked dataset constitutes a significant 34% of the entire dataset, offering a more comprehensive temporal representation than the limited 4% derived solely from parked and idling states (cf. Table 2). Additionally, the considered OWT was parked or idling for only 2 weeks during the entire 15-month period, while smart tracking 34% of data is more evenly spread over the monitoring period.



**Figure 8.** Data description plots showing the proportion of the operational states (8c) for all the data, the training data and the smart tracked data for SS2 both in absolute (8a) and relative (8b) terms.

#### 4.2. Control charts for decision-making

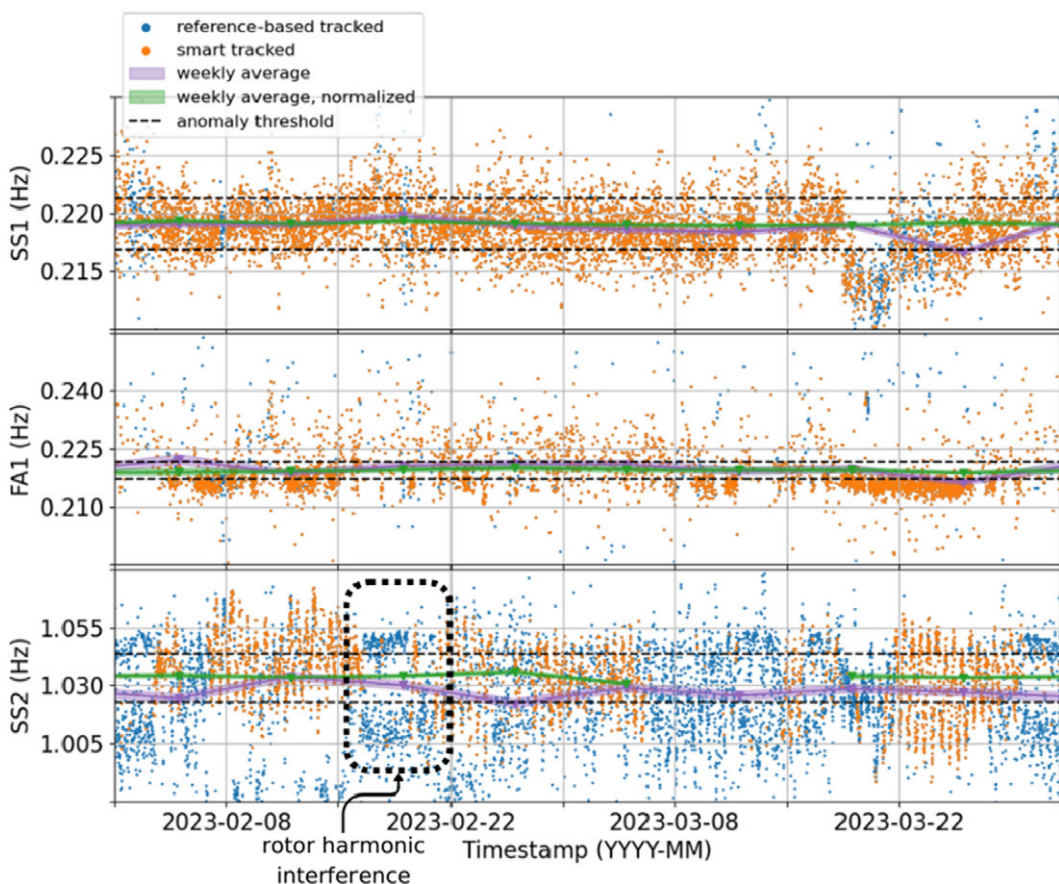
The control charts shown in Figure 9 depict the reference-based tracked modes in blue and show the result of averaging the data over 1 week in purple. After applying the smart tracking methodology, the orange points are tracked. The residual is then calculated for these modes and averaged over a week, as the weekly average, normalized, shown in green. Additionally, the mean training frequency is added to the residual to bring it to the level of the measured frequencies.

For all three considered modes, the green line, representing the normalized and averaged natural frequencies for low uncertainty predictions, remains stable around the mean, as expected from healthy data when the variations in natural frequencies  $f_m(t)$  are driven by modeled EOV  $\hat{f}_m(\mathbf{x}(t))$  and noise.

For the reference-based tracking, shown in purple, the process noise is removed through the weekly averaging. However, the EOV remains, showing variations in the blue line that are not attributed to changes in the structure. This highlights the importance of normalizing the natural frequency for EOV.

The smart tracking discards most of the modes that are at risk of being influenced by rotor harmonics. An example of rotor harmonic interference is highlighted as a dashed box in Figure 9. When too much data are not tracked, the normalized weekly average cannot accurately be calculated, this is visible as gaps over time appear in the smart tracked results.

Currently, a data-driven limit can be set on the averaging and normalized natural frequency. In this case, a 1% shift is used as an anomaly threshold in Figure 9 to trigger anomaly detection.



**Figure 9.** Control charts for monitoring the OWT natural frequencies after weekly data averaging (purple), using the reference-based tracking (blue) compared to the ML- and UQ-based tracking (orange) and normalization (green).

### 4.3. Validation of damage detection

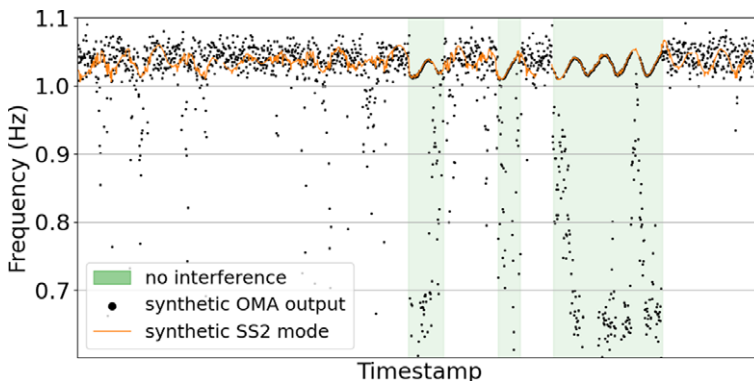
Although the control charts show that it is possible to monitor the shifts in the different natural frequencies of the OWT, no anomaly has been reported on the OWT, and thus no significant shift in the natural frequency during the monitoring period. Therefore, a synthetic dataset, based on the real data, with different introduced natural frequency shifts is constructed to quantify the added value of the method over the reference-based methodology. The dataset is created to mimic the interactions between the rotor harmonic and the SS2 mode as outlined in Section 2.1.

To construct these data, a simple LR is trained on the selected SS2 data subset, when no harmonic interference occurs, to learn simple relationships between the different environmental and operational conditions and the SS2 mode. The LR is then used to generate cyclical changing data with some overlaid Gaussian noise, as expected from the OMA process for the SS2 mode. This is shown as the orange line in Figure 10. This represents the part of the physical mode that would be obtained by the OMA if no rotor harmonic interference occurred. To simulate the rotor harmonic interference, the actual RPM data for the full monitoring period are used to generate data representing the harmonic with added noise (representing both the noise in the data and the RPM process). This is generated as  $h_P = P * \frac{\text{RPM}}{60} + v$  with  $P = 6$  for the 6P harmonic. The harmonic data and structural mode are overlaid and when both occur with a difference smaller than 0.2 Hz, the structural mode is removed as it is considered to be “clouded” by the harmonic. This is the final synthetic OMA output and is illustrated in Figure 10 for a period with and without harmonic interaction.

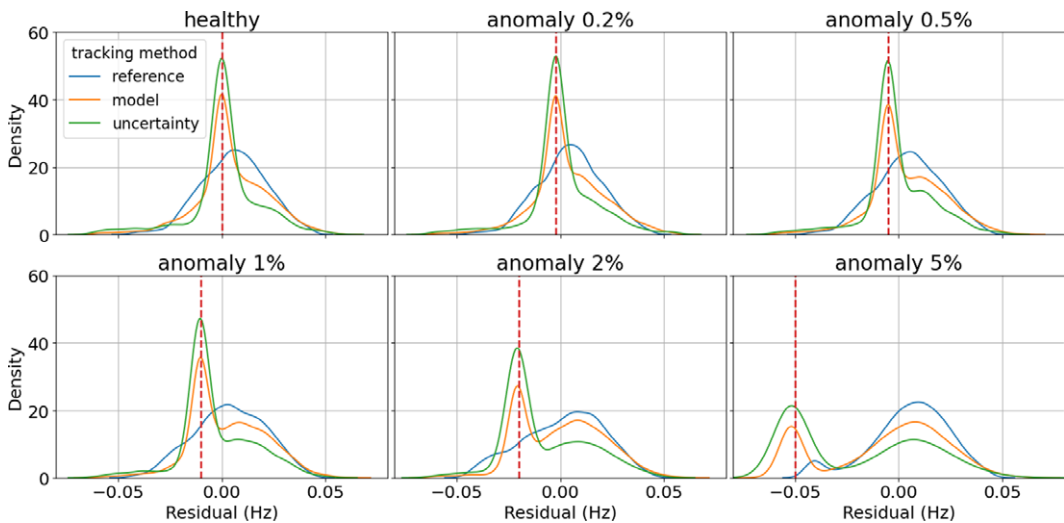
Figure 10 illustrates the harmonic interference issue in the synthetic data. During the vast majority of time, the SS2 mode is obfuscated by nearby harmonics, rendering it nearly impossible to reliably track the mode. Only when the rotor speed drops, highlighted as green areas in the plot, the SS2 mode reappears in the automated OMA. From an SHM perspective, only these periods highlighted in green contribute to a reliable assessment.

Once the dataset is created, the last month of data is repeated five times with an increasing shift in the physical part of the SS2 frequency, as given by the LR. These shifts start at  $-0.2\%$  and range up to  $-5\%$ . The repeated datasets of 1 month with increasing shifts form the anomalous data.

Three different methods are compared for anomaly detection. The first method uses reference-based tracking and simply measures the residual as the deviation of the natural frequency from the mean (taken from the selected SS2 data)  $\varepsilon_{\text{SS2}}(t) = f_{\text{SS2}}(t) - \mu_{\text{SS2}}$ . This method does not normalize the natural frequency and simply serves as a benchmark for the anomaly detection methodology. The second method uses a point-wise ML predictor, CatBoostRegressor, in this case, (Dorogush et al., 2018), to get a prediction of the SS2 mode every 10 minutes. If an OMA mode is detected with frequency inside the



**Figure 10.** Creation process of the synthetic dataset, with the synthetic SS2 mode created through an LR (orange) and the synthetic OMA output generated by overlaying the 6P harmonic (black), showing periods with and without (green) interference.



**Figure 11.** Residuals of three methods on a synthetic dataset with an increasing introduced anomaly on the physical mode with harmonic interaction.

$3\sigma$  band, it is tracked and used for anomaly detection, regardless of the ML model inputs. The residuals of the tracked modes and predictions are then taken as a measure of anomaly. Finally, the smart tracking methodology, as introduced in Section 3.3.1 combining ML prediction and UQ is used to remove OOD predictions from the tracking. Again, the residual is taken as a measure of anomaly but only for the predictions with low uncertainty.

For all three methodologies, the residual distribution for the repeated month of data with synthetic anomalies, as shown in Figure 11, is used as a metric to judge the anomaly detection capabilities of the method. Already from the healthy data, it is clear that introducing the ML model creates a narrower distribution around 0 Hz, showing a higher confidence in the “normality” of the data. The method introducing the uncertainty shows a slightly narrower peak as it removes part of the noise introduced by the harmonic interaction. When introducing the anomaly, it is clear that both ML- and uncertainty-methods follow the shift in the data, while the reference-based method only shows a shift with a 5% anomaly. Furthermore, the uncertainty-based method consistently shows a higher peak around the anomaly than the ML-based method alone, highlighting the improvement in the anomaly detection capabilities.

## 5. Future work

The proposed method lays the groundwork to improve and automate steps in the existing vibration-based SHM approach. However, future work as identified in this paragraph is required to further advance the method.

### 5.1. Tracking the whirling mode: OOD detection limitations

A final analysis to better understand the OOD detection based on the UQ is made on the OWT whirling mode, a flap- and edgewise mode of the rotor blades (Hansen, 2007). This mode is chosen because this mode has a rotor harmonic crossing for a specific RPM range ( $[7, 7.5]$  RPM) within an operational state, namely “rpm < MAX” as defined in Figure 8c. As a test, the whirling mode, visible at RPM > 6, is selected on a 6-month data subset, omitting the RPM range of rotor harmonic interaction, to train the ML model of the proposed SHM strategy.

Following the ML model training, the whirling mode is tracked using the methodology described in Section 3.3.1. In this first case, the whirling mode was accurately tracked, but the  $[7, 7.5]$  RPM was included in the smart tracked mode. This highlights that this range was not considered as OOD from the training data by the UQ as it can be interpolated, and the ML model yields good predictions for this operational state.

However, when the selected training data was limited to high RPM data only ( $\text{RPM} > 8.5$ ), the whirling mode was tracked for high RPM only. This example illustrates the limitations of the method, which only excludes OOD data when it cannot accurately be interpolated by the model from the training data. This could not be solved by tweaking the ML model hyperparameters. However, further investigation using other UQ methods (ANN-based or the full Bayesian ensemble applied to RF or XGB) could be explored.

## 5.2. Physics-informed decision criteria

By integrating a physics-based model to interpret the shifts in the natural frequency with respect to specific damaging scenarios, it is possible to elevate the SHM methodology from damage detection to damage quantification. This was previously done in Weil et al. (2023) for quantification of scouring depth based on SS1 frequency shift. By integrating this workflow on all the tracked modes, more different scenarios can be considered with a higher precision.

## 6. Conclusion

This study has introduced a novel vibration-based SHM methodology, integrating ML and UQ to automate and refine natural frequency tracking and anomaly detection, especially for OWTs. The methodology used a single ML model, trained on data from healthy operational states, to perform both mode tracking and normalization of EOV. Additionally, the application of UQ to the ML predictions enabled the distinction of OOD cases, which is crucial for addressing the interference of rotor harmonics.

A robust SHM framework was established, starting with the creation of a representative dataset for ML model training, that encompasses all environmental and operational conditions when the automated OMA outputs structural modes with no interference. Feature selection was performed using operational and METEO data. From a comparison of different ML models, tree-based learners emerged as the optimal choice, with Catboost being selected for its built-in UQ capabilities. The concept of a virtual ensemble provided a tool for UQ, giving an estimate of knowledge uncertainty, for improved mode tracking.

This study has shown the efficacy of the proposed smart tracking strategy based on ML models and UQ, through the practical application of the proposed methodology on a monitored monopile-supported OWT. It successfully distinguished between true structural modes and those affected by harmonic interference for the SS2 mode. Reducing the timestamps with interference from a substantial 20% to a mere 3%, while maintaining 34% of the original dataset. The study also explored the use of control charts as a practical tool for decision-making. These charts showcased the comparative effectiveness of the proposed smart tracking methodology against traditional reference-based tracking. The application on a synthetic dataset, created to simulate shifts in natural frequency that can be hidden by rotor harmonic interactions, further validated the methodology's robustness in detecting anomalies when compared to reference- and model-based implementations.

However, the study encountered challenges with OOD detection during the tracking of the whirling mode in specific RPM ranges. This highlights that the current methodology cannot exclude regions that can be interpolated from the training data by the ML model. This limitation highlights the potential for further research, including the exploration of alternative UQ methods to refine the detection of OOD samples.

In conclusion, the proposed methodology demonstrates significant improvements in SHM processes for OWTs, particularly in natural frequency tracking and anomaly detection, especially when monitoring

second-order modes. Future research will aim at combining this data-driven approach with physics-based models, to interpret the detected frequency shifts, enhancing the vibration-based SHM capabilities from damage detection to damage quantification.

**Data availability statement.** In this study, both public and proprietary data were used. The public data were downloaded from the Meetnet Vlaamse Banken (MVBC) provided by Flemish government, Agency for Maritime Services and Coast (2023). The proprietary data were provided by Parkwind under license. Due to these conditions, the proprietary data cannot be made publicly available. Details of the data and the conditions under which they were used are documented internally and can be accessed with permission from Parkwind. The developed Python packages for DBSCAN-based mode clustering is available at <https://doi.org/10.5281/zenodo.10523150> or [https://github.com/OWI-Lab/oma\\_clustering](https://github.com/OWI-Lab/oma_clustering) and the developed Python package for manual data selection is available at [https://github.com/OWI-Lab/data\\_selector](https://github.com/OWI-Lab/data_selector). Additionally, a Python package to directly access the weather data of the MVBC through the API was developed and published to pip (Weil, 2024b).

**Acknowledgments.** The authors would like to acknowledge the support of the Belgian Ministry of Economic Affairs through the ETF WINDSOIL project and the support of VLAIO through the De Blauwe Cluster SBO SOILTWIN project, as well as Parkwind for permission to use their data.

**Author contribution.** Conceptualization: M.W., C.S.J., and W.W. Methodology: M.W. and W.W. Funding acquisition: C.D. Data curation: M.W. and W.W. Data visualisation: M.W. Writing original draft: M.W., C.S.J., and W.W. All authors approved the final submitted draft.

**Funding statement.** This research was supported by grants from the ETF WINDSOIL.

**Competing interest.** The authors declare no competing interests exist.

**Ethical standard.** The research meets all ethical guidelines, including adherence to the legal requirements of the study country.

## References

- Al-jabery K, Obafemi-Ajayi T, Olbricht G and Wunsch D (2020) Data preprocessing. In *Computational Learning Approaches to Data Analytics in Biomedical Applications*. Cambridge, MA: Academic Press, pp. 7–27.
- Altman N and Krzywinski M (2017) Ensemble methods: Bagging and random forests. *Nature Methods* 14(10), 933–935.
- Arnekvist I, Carvalho JF, Kragic D and Stork JA (2020) The effect of target normalization and momentum on dying relu. Available at <https://arxiv.org/abs/2005.06195>
- Bai G and Chandra R (2023) Gradient boosting Bayesian neural networks via langevin MCMC. *Neurocomputing* 558, 126726.
- Bastani O and Alur R (2024) Cis 7000: Trustworthy machine learning (spring 2024). Lecture 14: Aleatoric vs. Epistemic Uncertainty. Available at <https://www.seas.upenn.edu/~obastani/cis7000/spring2024/docs/lecture14.pdf>.
- Bel-Hadj Y, Weil M, Weijtjens W and Devriendt C (2024) Experimental validation of automated OMA and mode tracking for structural health monitoring of transmission towers. *Structural Health Monitoring*. <https://journals.sagepub.com/doi/abs/10.1177/14759217241249048>
- Bergstra J, Komer B, Eliasmith C, Yamins D and Cox DD (2015) Hyperopt: A python library for model selection and hyperparameter optimization. *Computational Science & Discovery* 8(1), 014008.
- Chen T and Guestrin C (2016) XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*. New York, NY: ACM, pp. 785–794.
- Chipman H, George E and McCulloch R (2006) Bayesian ensemble learning. *Advances in Neural Information Processing Systems* 19, pp. 265–272.
- Dekking F, Kraaikamp C, Lopuhaä H and Meester L (2006) *A Modern Introduction to Probability and Statistics*. New York, NY: Springer.
- Devriendt C, Magalhães F, Weijtjens W, De Sitter G, Cunha A and Guillaume P (2014) Structural health monitoring of offshore wind turbines using automated operational modal analysis. *Structural Health Monitoring* 13(6), 644–659.
- Devriendt C and Weijtjens W (2017) Damping of offshore wind turbines. In *Offshore Wind Energy*, PO.085.
- Doebeling SW, Farrar CR, Prime MB, et al. (1998) A summary review of vibration-based damage identification methods. *Shock and Vibration Digest* 30(2), 91–105.
- Dorogush AV, Ershov V and Gulin A (2018) Catboost: Gradient boosting with categorical features support. Available at <https://arxiv.org/abs/1810.11363>.
- Duan T, Anand A, Ding DY, Thai KK, Basu S, Ng A and Schuler A (2020) NGBoost: Natural gradient boosting for probabilistic prediction. In *International Conference on Machine Learning*, PMLR, pp. 2690–2700.
- Ester M, Kriegl H-P, Sander J and Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, AAAI Press, Portland Oregon, pp. 226–231.

- Fallais D, Winkler K, Jurado CS, Weijtjens W, Stuyts B and Devriendt C** (2022) Farm wide sensitivity assessments of resonant frequencies of integrated offshore wind turbine finite element models. *Journal of Physics: Conference Series* 2265, 042053.
- Farrar CR and Worden K** (2012) Data normalisation. In *Structural Health Monitoring: A Machine Learning Perspective*. Wiley, Hoboken, New Jersey, United States, pp. 403–438.
- Flemish Government, Agency for Maritime Services and Coast** (2023) Meetnet Vlaamse Banken. Available at <https://meetnetvlaamsebanken.be/> (accessed 09 October 2023).
- Friedman JH** (2001) Greedy function approximation: A gradient boosting machine. *Annals of Statistics* 29(5), 1189–1232.
- Gawlikowski J, Tassi CRN, Ali M, Lee J, Humt M, Feng J, Kruspe A, Triebel R, Jung P, Roscher R, et al.** (2021) A survey of uncertainty in deep neural networks. Available at <https://arxiv.org/abs/2107.03342>.
- Géron A** (2019) *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd Edn. Sebastopol, CA: O'Reilly Media, Inc.
- Gré S, Döhler M, Andersen P and Mevel L** (2021) Kalman filter-based subspace identification for operational modal analysis under unmeasured periodic excitation. *Mechanical Systems and Signal Processing* 146, 106996.
- Grinsztajn L, Oyallon E and Varoquaux G** (2022) Why do tree-based models still outperform deep learning on typical tabular data? *Advances in Neural Information Processing Systems* 35, 507–520.
- Guyon I and Elisseeff A** (2003) An introduction to variable and feature selection. *Journal of Machine Learning Research* 3(Mar), 1157–1182.
- Guyon I, Weston J, Barnhill S and Vapnik V** (2002) Gene selection for cancer classification using support vector machines. *Machine Learning* 46, 389–422.
- Häckell MW and Rolfes R** (2013) Monitoring a 5mw offshore wind energy converter—Condition parameters and triangulation based extraction of modal parameters. *Mechanical Systems and Signal Processing* 40(1), 322–343.
- Hansen MH** (2007) Aeroelastic instability problems for wind turbines. *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology* 10(6), 551–577.
- Hassan U and Anwar MS** (2010) Reducing noise by repetition: Introduction to signal averaging. *European Journal of Physics* 31(3), 453.
- Hehn TM, Kooij JFP and Hamprecht FA** (2020) End-to-end learning of decision trees and forests. *International Journal of Computer Vision* 128(4), 997–1011.
- Hespanhol L, Vallio CS, Costa LM and Saragiotto BT** (2019) Understanding and interpreting confidence and credible intervals around effect estimates. *Brazilian Journal of Physical Therapy* 23(4), 290–301.
- Hüllermeier E and Waegeman W** (2021) Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning* 110(3), 457–506.
- Jahani K, Langlois RG and Afagh FF** (2022) Structural dynamics of offshore wind turbines: A review. *Ocean Engineering* 251, 111136.
- Maddox WJ, Izmailov P, Garipov T, Vetrov DP and Wilson AG** (2019) A simple baseline for Bayesian uncertainty in deep learning. *Advances in Neural Information Processing Systems* 32, pp. 13153–13164.
- Magalhães F and Cunha Á** (2011) Explaining operational modal analysis with data from an arch bridge. *Mechanical Systems and Signal Processing* 25(5), 1431–1450.
- Magalhães F, Cunha Á and Caetano E** (2012) Vibration based structural health monitoring of an arch bridge: From automated oma to damage detection. *Mechanical Systems and Signal Processing* 28, 212–228.
- Malinin A** (2019) *Uncertainty Estimation in Deep Learning with Application to Spoken Language Assessment*. PhD thesis, University of Cambridge, UK.
- Malinin A, Prokhorenkova L and Ustimenko A** (2020) Uncertainty in gradient boosting via ensembles. Available at <https://arxiv.org/abs/2006.10562>
- Martins N, Caetano E, Diord S, Magalhães F and Cunha Á** (2014) Dynamic monitoring of a stadium suspension roof: Wind and temperature influence on modal parameters and structural response. *Engineering Structures* 59, 80–94.
- McInnes L and Healy J** (2017) Accelerated hierarchical density based clustering. In *Data Mining Workshops (ICDMW), 2017 IEEE International Conference on*. Piscataway, NJ: IEEE, pp. 33–42.
- Mohanty P and Rixen DJ** (2003) Operational dynamic testing in the presence of harmonic excitation. In *Computational Fluid and Solid Mechanics, Volume 2: Proceedings Second MIT Conference on Computational Fluid and Solid Mechanics*. Elsevier, Amsterdam, pp. 477–481.
- Mondal A** (2021) Uncertainty estimation. GitHub repository. Available at <https://github.com/akashmondal1810/UncertaintyEstimation> (accessed on 18 April 2024).
- Motte K, Weijtjens W, Devriendt C and Guillaume P** (2015) Operational modal analysis in the presence of harmonic excitations: A review. In *Dynamics of Civil Structures, Volume 2: Proceedings of the 33rd IMAC, A Conference and Exposition on Structural Dynamics*. Cham: Springer, pp. 379–395.
- Oliveira G, Magalhães F, Cunha A and Caetano E** (2018) Vibration-based damage detection in a wind turbine using 1 year of data. *Structural Control and Health Monitoring, Volume 25*.
- Pearce, T, Leibfried F and Brintrup A** (2020) Uncertainty in neural networks: Approximately Bayesian ensembling. In *International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 234–244.

- Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M and Duchesnay E (2011) Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12, 2825–2830.
- Peeters B and De Roeck G (2001a) One year monitoring of the z24 bridge: Environmental influences versus damage effects. In *Proceedings of IMAC-XVIII*. San Antonio, TX: Springer, pp. 1570–1579.
- Peeters B and De Roeck G (2001b) Stochastic system identification for operational modal analysis: A review. *Journal of Dynamic Systems, Measurement, and Control* 123(4), 659–667.
- Peeters B, Maeck J and De Roeck G (2001) Vibration-based damage detection in civil engineering: Excitation sources and temperature effects. *Smart Materials and Structures* 10(3), 518.
- Pereira S, Pacheco J, Pimenta F, Moutinho C, Cunha A and Magalhães F (2023) Contributions for enhanced tracking of (onshore) wind turbines modal parameters. *Engineering Structures* 274, 115120.
- Poggi M, Aleotti F, Tosi F and Mattoccia S (2020) On the uncertainty of self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, WA, USA, pp. 3227–3237.
- Prendergast LJ, Gavin K and Doherty P (2015) An investigation into the effect of scour on the natural frequency of an offshore wind turbine. *Ocean Engineering* 101, 1–11.
- Prokhorenkova L (2020) Tutorial: Uncertainty estimation with catboost. *Towards Data Science*. Available at <https://towardsdatascience.com/tutorial-uncertainty-estimation-with-catboost-255805ff217e> (accessed 15 February 2024).
- Prokhorenkova L, Gusev G, Vorobev A, Dorogush AV and Gulin A (2018) Catboost: Unbiased boosting with categorical features. *Advances in Neural Information Processing Systems* 31, pp. 6638–6648.
- Reynders E (2012) System identification methods for (operational) modal analysis: Review and comparison. *Archives of Computational Methods in Engineering* 19(1), 51–124.
- Salawu O (1997) Detection of structural damage through changes in frequency: A review. *Engineering Structures* 19(9), 718–723.
- Sanz H, Valim C, Vegas E, Oller JM and Reverter F (2018) SVM-RFE: Selection and visualization of the most relevant features through non-linear kernels. *BMC Bioinformatics* 19(1), 1–18.
- Song M, Mehr NP, Moaveni B, Hines E, Ebrahimian H and Bajric A (2023) One year monitoring of an offshore wind turbine: Variability of modal parameters to ambient and operational conditions. *Engineering Structures* 297, 117022.
- Ubertini F, Comanducci G and Cavalagli N (2016) Vibration-based structural health monitoring of a historic bell-tower using output-only measurements and multivariate statistical analysis. *Structural Health Monitoring* 15(4), 438–457.
- Van Der Tempel J (2006) *Design of support structures for offshore wind turbines*. Published and distributed by the author.
- van der Veen GJ, Van Wingerden J, Fleming P, Scholbrock A and Verhaegen M (2013) Global data-driven modeling of wind turbines in the presence of turbulence. *Control Engineering Practice* 21(4), 441–454.
- van Vondelen AA, Iliopoulos A, Navalkar ST, van der Hoek DC and van Wingerden J-W (2023) Modal analysis of an operational offshore wind turbine using enhanced kalman filter-based subspace identification. *Wind Energy* 26(9), 923–945.
- Van Wyk A (2022) Encoding cyclical features for deep learning. Kaggle notebooks. Available at <https://www.kaggle.com/code/avanwyk/encoding-cyclical-features-for-deep-learning> (accessed 18 April 2024).
- Verboven P, Parloo E, Guillaume P and Van Overmeire M (2002) Autonomous structural health monitoring—part i: Modal parameter estimation and tracking. *Mechanical Systems and Signal Processing* 16(4), 637–657.
- Wang C, Nord TS, Ziemer G and Li G (2023) Towards uncertainty and sensitivity analysis for modal parameters identification during ice–structure interaction. *Ocean Engineering* 277, 114224.
- Weijtens W and Devriendt C (2017) Temporal evolution of stiffness for offshore monopile foundations. In *Offshore Wind Energy*, London, United Kingdom, PO.080.
- Weijtens W, Lataire J, Devriendt C and Guillaume P (2014a) Dealing with periodical loads and harmonics in operational modal analysis using time-varying transmissibility functions. *Mechanical Systems and Signal Processing* 49(1), 154–164.
- Weijtens W, Shirzadeh R, De Sitter G and Devriendt C (2014b) Classifying resonant frequencies and damping values of an offshore wind turbine on a monopile foundation for different operational conditions. In *EWEA Annual Event*, Fira de Barcelona Gran Via, Barcelona, Spain.
- Weijtens W, Verbelen T, De Sitter G and Devriendt C (2016) Foundation structural health monitoring of an offshore wind turbine —A full-scale case study. *Structural Health Monitoring* 15(4), 389–402.
- Weil M (2024a) data\_selector. Github repository. Available at [https://github.com/OWI-Lab/data\\_selector](https://github.com/OWI-Lab/data_selector) (accessed 18 April 2024).
- Weil M (2024b) mvbc. Python package. Available at <https://pypi.org/project/mvbc/> Version 0.0.7.
- Weil M (2024c) Owi-lab/oma\_clustering: Zenodo registration. Available at <https://doi.org/10.5281/zenodo.10523150> (accessed 18 April 2024).
- Weil M, Sastre Jurado C, Chaar M, Winkler K, Weijtens W and Devriendt C (2023) Quantifiable scour detection for offshore wind turbines using resonance frequency monitoring and a digital twin. *Proceedings of the Fourteenth International Workshop on Structural Health Monitoring (IWSHM)*, September 12–14.
- Weil M, Weijtens W and Devriendt C (2022) Autoencoder and Mahalanobis distance for novelty detection in structural health monitoring data of an offshore wind turbine. *Journal of Physics: Conference Series* 2265(3), 032076.
- Weinert J, Smolka U, Schumann B and Chen PW (2015) Detecting critical scour developments at monopile foundations under operating conditions. In *EWEA Annual Event*. UK: Loughborough University.

**Winkler K, Weil M, Sastre Jurado C, Stuyts B, Weijtjens W and Devriendt C** (2023) Quantifying the effect of rock armour scour protection on eigenfrequencies of a monopile supported OHVS. In *Journal of Physics: Conference Series* (Vol. 2626, No. 1, p. 012039). IOP Publishing.

**Wu X, Xiao L, Sun Y, Zhang J, Ma T and He L** (2022) A survey of human-in-the-loop for machine learning. *Future Generation Computer Systems* 135, 364–381.

**Xiang Z-Q, Wang J-T, Wang W, Pan J-W, Liu J-F, Le Z-J and Cai X-Y** (2024) Vibration-based health monitoring of the offshore wind turbine tower using machine learning with Bayesian optimisation. *Ocean Engineering* 292, 116513.

### Appendix A. Data normalization through min–max normalization

As mentioned in Section 3.2.1, this research incorporates min–max normalization for the selected data during the model comparison on the input of the artificial neural networks (ANNs). The min–max normalization is defined by Equation (7), and the used minimum and maximum values are given in Table A1.

$$x'_i(t) = \frac{x_i(t) - x_{i,\min}}{x_{i,\max} - x_{i,\min}} \quad (7)$$

where:

- $x_i(t)$  is the original value at timestamp  $t$ .
- $x_{i,\min}$  is the minimum value of the feature.
- $x_{i,\max}$  is the maximum value of the feature.
- $x'_i(t)$  is the normalized value.

**Table A1.** Min and max values for min–max normalization

Parameter	Min value	Max value
10% highest waves	17.000	257.667
Wave height	13.000	201.667
Sea water temperature	6.300	20.800
Tide-TAW	−2.000	501.000
Air pressure	993.300	1038.667
RPM	0.000	10.445
Pitch	−3.945	90.840
Power	−145.728	9524.993
sin(Yaw)	−0.999	0.999
cos(Yaw)	−1.000	1.000
sin(Wind direction)	−0.999	1.000
cos(Wind direction)	−1.000	1.000

Appendix B. Hyperparameter optimization

The settings and the results of the hyperparameter optimization using the Hyperopt Python package (Bergstra et al., 2015) are given for all the considered machine learning models and natural frequencies in Table B1. For every algorithm optimization, a limit of 50 trials or 2 hours is applied.

Table B1. Settings and results of the hyperparameter optimization using the Hyperopt Python package

Model	Hyperparameter	Hyperspace	SS1	SS2	FA1
RF	n estimators	hp.uniformint(10, 1000)	956	472	339
	max depth	hp.uniformint(2, 16)	8	13	11
	min samples split	hp.uniformint(2, 10)	8	3	3
	min samples leaf	hp.uniformint(1, 5)	4	4	5
XGB	n estimators	hp.uniformint(10, 1000)	522	766	241
	max depth	hp.uniformint(2, 16)	5	4	3
	learning rate	hp.uniform(0.001, 0.5)	0.019	0.037	0.052
	colasample bytree	hp.uniform(0.1, 1)	0.380	0.534	0.931
CB	iterations	hp.uniformint(10, 1000)	114	434	549
	depth	hp.uniformint(2, 16)	8	9	6
	learning rate	hp.uniformint(0.001, 0.5)	0.223	0.051	0.019
	l2 leaf reg	hp.uniform(1, 10)	9.269	8.748	4.607
ANN1	units layer1	hp.uniformint(32, 256)	235	150	247
	batch normalization	hp.choice([False, True])	False	False	False
	learning rate	hp.uniform(0.0001, 0.01)	0.003	0.002	0.005
ANN2	units layer1	hp.uniformint(32, 256)	149	85	139
	units layer2	hp.uniformint(32, 256)	254	57	126
	batch normalization	hp.choice([False, True])	False	False	False
	learning rate	hp.uniform(0.0001, 0.01)	0.004	0.001	0.003
ANN3	units layer1	hp.uniform(32, 256)	191	156	126
	units layer2	hp.uniform(32, 256)	163	117	140
	units layer3	hp.uniform(32, 256)	33	43	76
	batch normalization	hp.choice([False, True])	False	False	False
	learning rate	hp.uniform(0.0001, 0.01)	0.001	0.001	0.003

### Appendix C. Operational states definitions

This appendix provides a detailed description of each operational state as shown in Figure 8. Table C1 lists the specific operational states and includes a brief description for each, clarifying the operational state.

**Table C1.** Description of the operational states of Figure 8

Operational state	Description
Parked	Turbine parked, pitched at >87, rotating at less than 0.4 rpm
Pitch:78	Turbine pitched at 78°
Idling	Generic idling (no constraint on pitch)
Cutin	Turbine speeding up to operational speeds
Cutin (pitch >20)	Turbine speeding up to operational speeds with pitch >20
rpm < 6.6	Turbine rotating up to 6.6 rpm
rpm6.6	Turbine rotating at 6.6 rpm
Rated power	Turbine at rated power
rpm < max	Turbine speeding up to rated RPM
Rated RPM	Turbine rotating at 10.4 rpm or 10.445 rpm
High wind	Turbine reducing output power at extreme wind speeds
Curtailed (All)	Catchall for data points that fall out of the normal behavior
Missing SCADA	No SCADA data available
Other	All data not labeled in the previous operational states