Fault Detection in Wind Turbines Using Health Index Monitoring with Variational Autoencoders

Shun Wang, Yolanda Vidal, Francesc Pozo

Detailed Response to Editors and Reviewers

Dear Editors and Reviewers,

We sincerely thank you for your time, insightful comments, and constructive suggestions, which have been invaluable in refining our manuscript. We have carefully considered all the points raised and have carried out comprehensive revisions accordingly. We believe that these changes have significantly improved the quality, clarity, and rigor of the manuscript.

In the following, we provide a detailed point-by-point response to each comment, indicating where the corresponding revisions can be found in the updated manuscript. The changes made in response to the comments of the reviewers are highlighted in red. We believe that the quality of the revised paper is much better than the original version, and we hope that you find the revisions acceptable.

Point-to-Point Responses to the Reviewers

Response to Reviewer #1

Overall Evaluation: The paper entitled "Fault Detection in Wind Turbines Using Health Index Monitoring with Variational Autoencoders" deals with an interesting and timely topic, which is definitely adequate for the scientific objectives of the journal.

The quality of the presentation is in general high. Indeed, the rationale for the proposed methods is clearly explained. The test case is well discussed. The method is rigorously applied and the obtained results are convincing and sound. I have particularly appreciated the discussion of the icing vs. pitch imbalance case.

Despite the overall high quality of the paper, there are minor issues which in my opinion could be addressed in order to further improve the quality of the paper.

Authors' Response: We are truly grateful for your detailed and positive evaluation of our manuscript. Your constructive feedback has been instrumental in helping us improve the quality of the paper, and we have thoroughly revised the manuscript based on your specific suggestions. We believe that the changes have significantly enhanced the completeness and clarity of the manuscript.

We have carefully addressed your comments as detailed below.

Comment 1-1: In Table 1, aren't Crest Factor and Peak Factor the same quantity or am I wrong?

Authors' Response: Thank you for this careful observation. You are absolutely correct. Peak factor and crest factor are the same concept, referring to the ratio of a waveform's maximum (peak) value to its root mean square (RMS) value, both defined as $\frac{A_{\max}}{\text{RMS}}$. This redundancy was an oversight in our initial feature set definition, and we sincerely appreciate you bringing it to our attention.

In the revised manuscript, we have removed the redundant peak factor, resulting in a total of 19 features (12 time-domain and 7 frequency-domain features). To accommodate this change, the VAE architecture has been adjusted accordingly: the input and output layers now contain 266 neurons (19 features × 14 channels), while the encoder hidden layers (128, 64, 32 neurons), decoder hidden layers (32, 64, 128 neurons), and latent dimension (16) remain unchanged. This modification affects only the input/output dimensionality without altering the core network structure or learning capacity.

All experiments have been re-run with the corrected 19-feature set. The results confirm that the framework maintains its detection performance across all three fault scenarios, with detection accuracy, recall, precision, and F_1 scores remaining consistent with those originally reported. Table 1 has been updated in the revised manuscript to reflect the corrected feature set.

Comment 1-2: In Table 1, you employ the x symbol for the signal in the definition of Crest Factor and Clearance Factor, but in the caption of the Table you declare that the signal is indicated with a.

Table 1: Formulae for the selected features in time and frequency domains.

Domain	Features and Formulae	
Time domain	Standard deviation: $\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(a_i - \mu)^2}$	Maximum value: $a_{\max} = \max(\lvert a \rvert)$
	Minimum value: $a_{\min} = \min(a)$	Peak-to-peak value: $a_{pp} = \max(a) - \min(a)$
	Absolute mean: $a_{\text{mean}} = \frac{1}{n}\sum_{i=1}^{n}\lvert a_i \rvert$	Skewness: $S = \frac{1}{n}\sum_{i=1}^{n}\frac{(a_i-\mu)^3}{\sigma^3}$
	Kurtosis: $K = \frac{1}{n}\sum_{i=1}^{n}\frac{(a_i-\mu)^4}{\sigma^4} - 3$	Root mean square: $a_{\text{rms}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} a_i^2}$
	Waveform factor: $\text{WF} = \frac{a_{\text{rms}}}{a_{\text{mean}}}$	Impulse factor: $\text{IF} = \frac{a_{\max}}{a_{\text{mean}}}$
	Crest factor: $\text{CF} = \frac{a_{\max}}{a_{\text{rms}}}$	Clearance factor: $\text{CLF} = \frac{a_{\max}}{(\frac{1}{n}\sum_{i=1}^{n}\sqrt{\lvert a_i \rvert})^2}$
Frequency domain	Spectral mean: $Y_\mu = \frac{1}{m}\sum_{i=1}^{m}\lvert Y_i \rvert$	Spectral variance: $Y_{\sigma^2} = \frac{1}{m}\sum_{i=1}^{m}(\lvert Y_i \rvert - Y_\mu)^2$
	Spectral std. deviation: $Y_\sigma = \sqrt{Y_{\sigma^2}}$	Spectral entropy: $H_Y = -\sum_{i=1}^{m} p_i \log_2(p_i)$
	Spectral energy: $E_Y = \frac{1}{m}\sum_{i=1}^{m}\lvert Y_i \rvert^2$	Spectral skewness: $S_Y = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{\lvert Y_i \rvert - Y_\mu}{Y_\sigma}\right)^3$
	Spectral kurtosis: $K_Y = \frac{1}{m}\sum_{i=1}^{m}\frac{(\lvert Y_i \rvert - Y_\mu)^4}{{Y_\sigma}^4} - 3$	

Note: $\{a_i\}$ represents the time-domain vibration signal, n is the total number of sampling points, μ is the mean, σ is the standard deviation, a_{rms} is RMS, a_{mean} is absolute mean, and $a_{\max}$, $a_{\min}$ denote maximum and minimum values. $\{Y_i\}$ represents the magnitude of frequency components from FFT, m is the number of frequency bins, and $p_i = \frac{\lvert Y_i \rvert}{\sum_{k=1}^{m}\lvert Y_k \rvert}$.

Authors' Response: Thank you for catching this inconsistency. You are correct—this is a notational error. In the revised manuscript, we have corrected the formulas to use a_i consistently throughout Table 1:

- Crest factor: $\text{CF} = \frac{a_{\max}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n} a_i^2}} = \frac{a_{\max}}{a_{\text{rms}}}$
- Clearance factor: $\text{CLF} = \frac{a_{\max}}{(\frac{1}{n}\sum_{i=1}^{n}\sqrt{\lvert a_i \rvert})^2}$

We have ensured notational consistency throughout the revised manuscript.

Comment 1-3: The features in the frequency domain require more explanation. I guess that their computation passes through a Fourier Transform. How is it computed? On the sub-chunks indicated in Figure 8? Please clarify.

Authors' Response: Thank you for requesting this clarification. You are correct—the frequency-domain features are computed using the Fast Fourier Transform (FFT). We apologize for the insufficient detail in the original manuscript.

The FFT is applied to each 10-second sample (2000 data points at 200 Hz sampling rate). Specifically, the FFT converts the time-domain vibration signal $\{a_i\}_{i=1}^{n}$ into the frequency domain, yielding complex-valued frequency components $\{Y_i\}_{i=1}^{m}$, where m is the number of frequency bins. The magnitude spectrum $\lvert Y_i \rvert$ is then computed, and the normalized power distribution $p_i = \frac{\lvert Y_i \rvert}{\sum_{k=1}^{m}\lvert Y_k \rvert}$ is used for spectral entropy calculation. The seven frequency-domain features (spectral mean, variance, standard

deviation, entropy, energy, skewness, and kurtosis) are computed based on this magnitude spectrum, as defined in Table 1.

We have added the following clarification in Section 3.2 (Feature Extraction) of the revised manuscript:

> For frequency-domain feature extraction, the fast Fourier transform (FFT) is applied to each sample (2000 data points, corresponding to 10 seconds of operation at 200 Hz sampling rate). The FFT converts the time-domain vibration signal $\{a_i\}_{i=1}^{n}$ into the frequency domain, yielding complex-valued frequency components $\{Y_i\}_{i=1}^{m}$, where m is the number of frequency bins. The magnitude spectrum $|Y_i|$ is computed, and the normalized power distribution $p_i = \frac{|Y_i|}{\sum_{k=1}^{m} |Y_k|}$ is used for spectral entropy calculation. The seven frequency-domain features, including spectral mean, variance, standard deviation, entropy, energy, skewness, and kurtosis, are then computed based on this magnitude spectrum, as defined in Table 1.

Comment 1-4: How have you selected the division of the data sets indicated in Figure 6?

Authors' Response: Thank you for this important question. We recognize that the rationale behind our dataset division strategy was not sufficiently explained in the original manuscript.

The dataset division was determined based on the operational timeline shown in Figure 4 and guided by two key principles: (1) training and validation sets must consist exclusively of confirmed normal operation data to ensure the model learns only healthy turbine behavior, and (2) a chronological approach maintains temporal consistency.

As shown in Figure 4, the pitch drive coupling replacement on February 24, 2022, significantly altered the turbine's dynamic characteristics, effectively creating two distinct operational regimes that

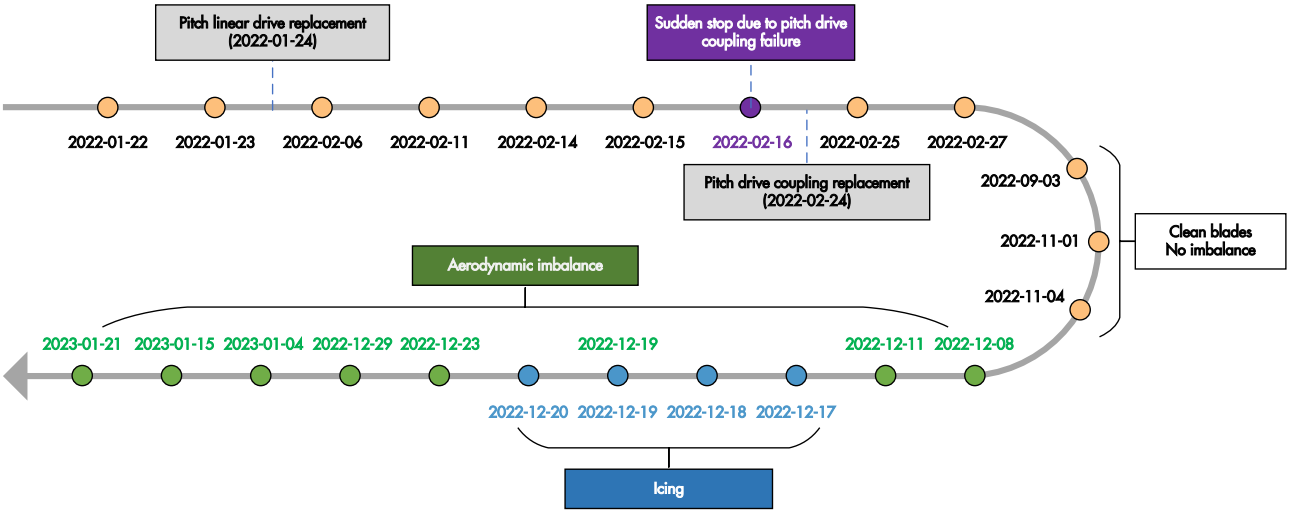

Figure 4: Timeline of failures and maintenance for Aventa AV-7 ETH Zurich research wind turbine.

require separate models. For pitch fault detection (pre-replacement period), we selected confirmed normal operation data from February 11 to February 15, 2022, allocating 106 timestamps for training and 43 timestamps for validation. For aerodynamic imbalance and icing detection (post-replacement period), only three dates after the coupling replacement are confirmed as normal operations without any fault events: September 3, 2022, November 1, 2022, and November 4, 2022. From these limited confirmed healthy operation periods, we used the data from September 3, 2022, along with a portion of the data from November 1, 2022, as the training set (17 timestamps total). The remaining portion of the data from November 1, 2022, was allocated to the validation set (5 timestamps), while the data from November 4, 2022, was reserved for the test set. This chronological division ensures that validation data reflects similar operational conditions while remaining independent for threshold establishment.

In contrast to the training and validation sets, test sets contain both normal and faulty operation periods, enabling comprehensive evaluation of the framework's ability to distinguish between healthy and faulty conditions. This mixed composition allows assessment of detection accuracy, false alarm rates, and the framework's robustness under realistic operational scenarios.

We have clarified this strategy in Section 4.1 of the revised manuscript by adding the following explanation:

> A fundamental principle of this division is that training and validation sets consist exclusively of data from confirmed healthy operational periods, ensuring the VAE learns an accurate representation of normal turbine behavior without contamination from fault signatures.
>
> The dataset division follows a chronological approach to maintain temporal consistency. Validation sets are selected from periods immediately following the training periods, ensuring they reflect similar operational conditions while remaining independent for threshold establishment. In contrast to the training and validation sets, test sets contain both normal and faulty operation periods, enabling comprehensive evaluation of the framework's ability to distinguish between healthy and faulty conditions. This mixed composition allows assessment of detection accuracy, false alarm rates, and the framework's robustness under realistic operational scenarios.

Comment 1-5: In the case of pitch drive failure (Section 4.3.1), I appreciate the results, given that this kind of fault is very difficult to detect in advance. Nevertheless, I consider too strong the statement "The framework demonstrates exceptional early warning capability for this progressive fault. As shown in Figure 9, an alarm is triggered at 14:32, providing a 2.5-hour lead time before the actual turbine shutdown at 17:12. This substantial lead time enables proactive maintenance planning

and minimizes unexpected downtime". An advance time of 2.5 hours is impractical for proactive maintenance planning. In my opinion, it would be more important to emphasize the robust detection of the aerodynamic imbalance because this kind of error, being substantially static and with a time-to-failure potentially infinite in line of principle, can be treated timely by minimizing the performance losses.

Authors' Response: We appreciate this insightful comment and agree with your perspective. You are correct that our original statement overstated the practical value of the 2.5-hour lead time for maintenance planning, and we thank you for this important clarification.

In the revised manuscript, we have modified Section 4.3.1 to provide a more balanced and realistic assessment:

> The framework demonstrates the capability to provide early warning for this progressive fault. As shown in Figure 9, an alarm is triggered at 14:32, providing a 2.5-hour lead time before the actual turbine shutdown at 17:12. While this lead time is insufficient for comprehensive maintenance planning in operational scenarios, the result validates the framework's ability to detect early signs of pitch drive degradation before complete failure occurs.

Furthermore, following your recommendation, we have added emphasis on the practical significance of aerodynamic imbalance detection in both the Results section and the Conclusion. The revised text highlights that:

> Unlike rapidly progressing mechanical failures such as pitch drive faults, aerodynamic imbalances represent static conditions with potentially indefinite time-to-failure. The framework's robust and accurate detection of such faults (achieving 100% accuracy with zero false alarms) offers significant practical value for operational wind farms. Early identification of aerodynamic imbalances enables timely corrective actions that minimize performance losses, prevent secondary damage to turbine components, and optimize energy production efficiency.

We would like to thank you once again for your positive and constructive review of our work, which has been invaluable in improving the quality of this paper. We have thoroughly addressed your comments and queries and sincerely hope that these revisions will receive your approval.

Response to Reviewer #2

Overall Evaluation: Thank you for writing this paper, which I've found interesting and well written. You will find several comments in the attached pdf, and I report hereafter the main ones.

Authors' Response: We are very grateful for your detailed and positive evaluation of our manuscript. Your constructive feedback has been instrumental in helping us improve the quality of the paper, and we have thoroughly revised the manuscript based on your specific suggestions. We believe that the changes have significantly enhanced the completeness and clarity of the manuscript.

Below, we address the main comments and detailed comments from the annotated PDF.

Response to Main Comments

Comment 2-1: The features that you have selected are completely general and have nothing specific of wind turbines. You should acknowledge that there are other studies that focused on correlating damages with wind turbine-specific features, such as: 1P, 3P, modes frequency, damping and mode shapes.

Authors' Response: We appreciate this observation and fully acknowledge this limitation of our feature selection approach.

You are correct that previous research has demonstrated the effectiveness of physics-informed, wind turbine-specific features for fault detection. Physics-informed features such as rotor frequency (1P), blade passing frequency (3P for three-bladed turbines), structural modal frequencies, damping ratios, and mode shapes for detecting specific fault mechanisms [Bertelè et al., 2018, Riva et al., 2016, Cacciola et al., 2016, Chen and Griffith, 2023] can directly reveal specific fault mechanisms. These turbine-specific features offer superior physical interpretability and targeted sensitivity to particular fault types.

Our choice of general statistical and spectral features was driven by the objective of creating a framework with broader applicability that does not require detailed system identification, turbine-specific modeling, or prior knowledge of operational parameters. This approach enables straightforward application across different turbine models and configurations. However, we recognize that this generality comes at the cost of reduced physical interpretability.

In the revised manuscript, we have added the following acknowledgment in Section 3.2 (Feature Extraction):

> It is important to acknowledge that the features selected in this study are general statistical and spectral indicators rather than wind turbine-specific characteristics. Previous research has demonstrated the effectiveness of physics-informed features such as rotor frequency (1P), blade passing frequency (3P for three-bladed turbines), structural modal

> frequencies, damping ratios, and mode shapes for detecting specific fault mechanisms [Bertelè et al., 2018, Riva et al., 2016, Cacciola et al., 2016, Chen and Griffith, 2023]. These turbine-specific features can directly reveal whether, for example, a mass imbalance has increased 1P amplitude or whether aerodynamic asymmetries are affecting 3P harmonics, offering superior physical interpretability. Our choice of general features avoids the need for detailed system identification or turbine-specific modeling, enabling broader applicability across different turbine models.

We have also included the relevant references in the revised manuscript.

Comment 2-2: All the models presented in this paper are trained on 1 wind turbine during 1 measurement champaign. Therefore, they will not perform as well on other copies of the same turbine, or even after a few years.

Authors' Response: Thank you for raising this critical point regarding model generalization. We acknowledge that the models presented in this work are trained on data from a single turbine, as provided by the challenge dataset. This limitation is inherent to the available data rather than a methodological constraint of the proposed framework.

However, the framework is designed with scalability in mind and can be extended to multi-turbine scenarios. If data from multiple turbines within a wind farm were available, a more general or *universal* model could be developed by training on combined data from several turbines. This universal model would capture shared operational characteristics across turbines while accounting for inter-turbine variability. Once such a universal model is established, it could be efficiently fine-tuned for a specific turbine using only a small amount of its own healthy operational data and a few additional training epochs. In this way, the model would adapt to each turbine's unique behavioral patterns without requiring full retraining from scratch, significantly improving efficiency and scalability in practical wind farm applications.

This transfer learning strategy has been successfully demonstrated in similar industrial condition monitoring applications, where pre-trained models are adapted to new equipment with minimal site-specific data. The semi-supervised nature of our VAE-based framework is particularly well-suited for this approach, as the learned representations of normal operational patterns can serve as a strong initialization for new turbines.

In the revised manuscript, we have added explicit acknowledgment of this limitation and directions for future work in the Conclusion section:

> While the framework demonstrates promising detection capability, several limitations exist. The models were trained on data from a single turbine during specific measurement campaigns, and generalization to other turbines or extended operational periods

requires further validation. The framework provides anomaly detection without detailed fault diagnosis, and the use of general statistical features limits physical interpretability compared to wind turbine-specific features. Additionally, as a static baseline approach, the framework would require periodic retraining or integration with alarm management systems for long-term deployment to handle gradual turbine aging while maintaining sensitivity to new faults. Future work should address these limitations through validation across multiple turbines and extended operational periods, extension to fault localization and identification capabilities, exploration of hybrid approaches combining general features with physics-informed characteristics, and investigation of adaptive learning strategies for sustained operational deployment.

Comment 2-3: The threshold for detecting failures is not only a function of the turbine and features, but also of the machine learning model (see Fig. 10). Therefore, it lacks generality and might require expert tuning.

Authors' Response: Thank you for this comment. The detection threshold is specific to the combination of the turbine, features, and machine learning model, as illustrated by the different baseline methods in Figure 10. However, we would like to clarify two important points:

First, regarding the concern about "expert tuning": We emphasize that our threshold determination process is fully automated and requires no manual intervention. As described in Section 3.5 and illustrated in Figure 2, the threshold is systematically established as the maximum EWMA value observed across the training and validation datasets, both of which contain only confirmed healthy operational data. This is a systematic, rule-based process that requires no manual adjustment or subjective judgment.

Second, regarding "lack of generality": We acknowledge that the specific threshold value is inherently model-dependent. This characteristic, however, is fundamental to data-driven anomaly detection systems that learn from historical data. Different models capture different aspects of the underlying data structure, naturally resulting in different health index distributions, as shown in Figure 10. The key distinction is that while the threshold value itself is application-specific, the methodology for establishing it is generalizable. Our framework provides a systematic, end-to-end pipeline encompassing feature extraction, VAE-based modeling, EWMA smoothing, and automated threshold determination. This complete methodology can be consistently applied to any turbine, with each application establishing its calibration-specific threshold through the same automated process.

In summary, the framework delivers a complete, automated solution from raw vibration data to fault detection, where threshold establishment is an integral component rather than a manual tuning step.

Comment 2-4: Stemming from the previous point, I'm getting the impression that you have conveniently selected the training time and threshold to detect the failures that you knew were there.

Authors' Response: Thank you for raising this concern. We understand this concern and appreciate the opportunity to clarify our methodology. We wish to emphasize that our framework was developed and applied following a strict protocol that prevents any optimization based on prior knowledge of fault events. Specifically:

Dataset division based on temporal criteria: Our data partitioning was governed by two objective constraints: temporal chronology and confirmed operational status from the dataset (Figure 4). Training and validation sets were selected exclusively from confirmed normal operation periods in chronological order. This selection was constrained by operational reality rather than optimized for detection performance. The pitch drive coupling replacement on February 24, 2022, fundamentally altered the turbine's dynamics, requiring separate models for pre- and post-replacement periods. For the post-replacement period, only three dates were documented as fault-free: September 3, November 1, and November 4, 2022. We used all available normal data from these periods, allocating them chronologically: September 3 and a portion of November 1 for training, with the remaining portion of November 1 for validation. This division was determined purely by temporal sequence and data availability, not by optimizing detection performance on known faults.

Fixed threshold determination: The detection threshold follows a systematic rule established independently of test data. It is defined as the maximum EWMA value observed across the training and validation datasets. This threshold is computed once using only healthy operational data and remains fixed throughout all testing. No information from fault events or test performance is used in this process. The threshold therefore represents a genuine baseline of normal behavior rather than any optimization.

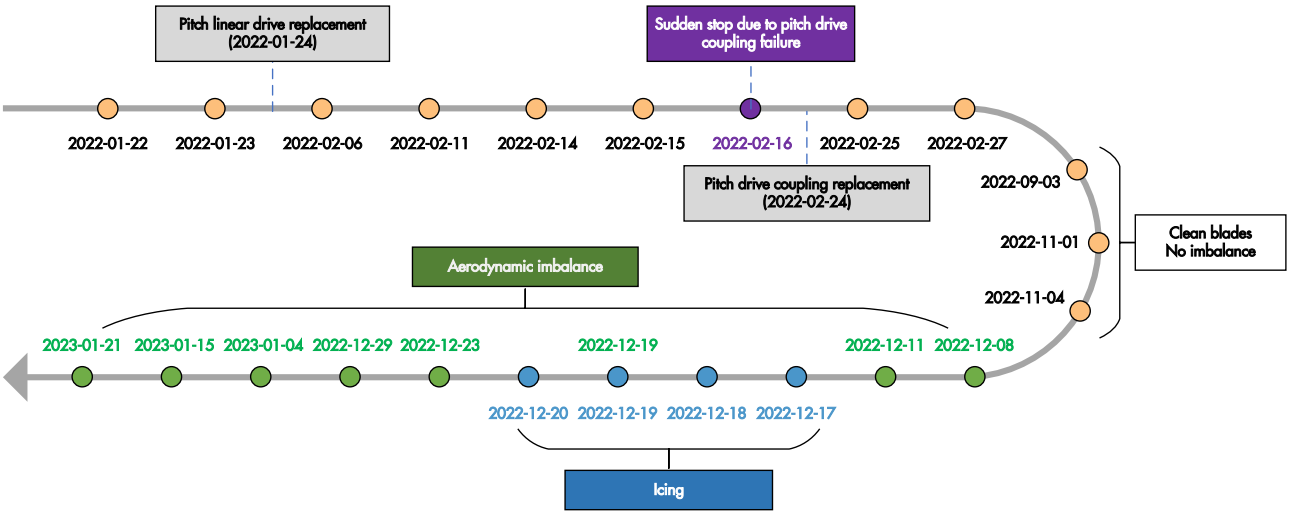

Figure 4: Timeline of failures and maintenance for Aventa AV-7 ETH Zurich research wind turbine.

Uniform application: The complete framework was applied identically to all three fault scenarios. The same feature extraction, VAE architecture, EWMA parameter, and threshold methodology were used throughout. No parameters were adjusted based on knowledge of specific fault characteristics. The test sets include complete fault event periods as provided in the dataset, without selective inclusion or exclusion of difficult cases.

In summary, our methodology follows the fundamental principle of semi-supervised learning: the model learns exclusively from normal operational data and is evaluated on unseen conditions including both normal and faulty states. The results represent genuine detection capability rather than fitting to known fault events.

Comment 2-5: Please specify how you have computed the frequency domain features.

Authors' Response: Thank you for requesting this clarification. We apologize for the insufficient detail in the original manuscript.

The fast Fourier transform (FFT) is applied to each 10-second sample (2000 data points at 200 Hz sampling rate). Specifically, the FFT converts the time-domain vibration signal $\{a_i\}_{i=1}^{n}$ into the frequency domain, yielding complex-valued frequency components $\{Y_i\}_{i=1}^{m}$, where m is the number of frequency bins. The magnitude spectrum $|Y_i|$ is then computed, and the normalized power distribution $p_i = \frac{|Y_i|}{\sum_{k=1}^{m}|Y_k|}$ is used for spectral entropy calculation. The seven frequency-domain features (spectral mean, variance, standard deviation, entropy, energy, skewness, and kurtosis) are computed based on this magnitude spectrum, as defined in Table 1. This feature extraction process is performed independently for each sample, yielding a feature vector that is then input to the VAE model for reconstruction error calculation.

We have added the following clarification in Section 3.2 (Feature Extraction) of the revised manuscript:

> For frequency-domain feature extraction, the fast Fourier transform (FFT) is applied to each sample (2000 data points, corresponding to 10 seconds of operation at 200 Hz sampling rate). The FFT converts the time-domain vibration signal $\{a_i\}_{i=1}^{n}$ into the frequency domain, yielding complex-valued frequency components $\{Y_i\}_{i=1}^{m}$, where m is the number of frequency bins. The magnitude spectrum $|Y_i|$ is computed, and the normalized power distribution $p_i = \frac{|Y_i|}{\sum_{k=1}^{m}|Y_k|}$ is used for spectral entropy calculation. The seven frequency-domain features, including spectral mean, variance, standard deviation, entropy, energy, skewness, and kurtosis, are then computed based on this magnitude spectrum, as defined in Table 1.

Comment 2-6: I agree with the other reviewers that little can be done by detecting problems 2.5 hours in advance. This is not necessarily an issue of your framework, because maybe the failure was simply not there before.

Authors' Response: We appreciate this insightful comment and agree with your perspective. You are correct that our original statement overstated the practical value of the 2.5-hour lead time for maintenance planning, and we thank you for this important clarification.

In the revised manuscript, we have modified Section 4.3.1 to provide a more balanced and realistic assessment:

> The framework demonstrates the capability to provide early warning for this progressive fault. As shown in Figure 9, an alarm is triggered at 14:32, providing a 2.5-hour lead time before the actual turbine shutdown at 17:12. While this lead time is insufficient for comprehensive maintenance planning in operational scenarios, the result validates the framework's ability to detect early signs of pitch drive degradation before complete failure occurs.

Response to Detailed Comments in Annotated PDF

Page 1, Line 19: *"And sometimes also hydrodynamic."*

Authors' Response: Thank you for this valuable addition. We have revised the sentence to include hydrodynamic effects: "Wind turbines operate in highly dynamic and stressful conditions, experiencing aerodynamic, gravitational, centrifugal, gyroscopic, and hydrodynamic loads."

Page 2, Line 49: *"I thought the opposite. If more data are available, I should be able to train a better model with the same parametrization."*

Authors' Response: Thank you for pointing this out. You are absolutely correct that more training samples improve model performance. Our statement was poorly phrased. The issue we intended to address is the challenge of high dimensionality rather than sample size. Using raw multi-channel signals directly creates an input space with thousands of features, leading to two key problems: first, an increased computational burden for training and inference; second, with finite training samples, models can easily find spurious correlations in this high-dimensional space and overfit to noise rather than learning true patterns of healthy operation—a phenomenon known as the *curse of dimensionality*. This is the primary motivation for our feature engineering step: reducing dimensionality from thousands of raw data points to 266 robust, physically meaningful features, mitigates both computational burden and overfitting risk while preserving essential information.

We have revised the text to clarify this distinction between sample size and feature dimensionality:

The high dimensionality of raw multi-channel vibration signals leads to increased computational burden and can cause overfitting when training samples are limited, as models may capture spurious correlations rather than true operational patterns. Feature extraction addresses these challenges by reducing the input space to a manageable set of physically interpretable features.

Page 5, Line 123: *"Please motivate the usage of a normal distribution."*

Authors' Response: Thank you for this important question. The choice of a standard normal distribution $\mathcal{N}(0, I)$ as the prior for the latent space is standard practice in VAE formulations and is motivated by several key reasons. First, it provides a mathematically tractable reference distribution that enables closed-form computation of the KL divergence term in the loss function, which is essential for efficient training. Second, it ensures the latent space is continuous and well-structured, preventing the encoder from learning disconnected clusters that would make sampling and interpolation problematic. Finally, this choice has been extensively validated both theoretically and empirically across numerous VAE applications.

We have added this explanation in the revised manuscript:

To ensure the latent space is continuous and well-structured, it is regularized to approximate a multivariate standard normal distribution $\mathcal{N}(0, I)$, a standard choice in VAE formulations that enables tractable optimization through closed-form Kullback-Leibler (KL) divergence computation [Doersch, 2016].

Page 5, Line 133: *"What I like about these features is that they are very general, but I don't like that they have nothing to do with wind turbine. These features will not say if a mass imbalance has caused the amplitude of the 1P loading to grow, if a certain rotor speed is causing a resonance or other physical phenomena. Furthermore, they depend on how the PSD is computed, which you should explain. See for example "A machine-learning-based approach for active monitoring of blade pitch misalignment in wind turbines" and other papers by Cacciola and Bottasso."*

Authors' Response: Thank you for this detailed comment. This concern regarding the generality of features versus wind turbine-specific characteristics has been addressed in our response to Comment 2-1, where we acknowledge this limitation and discuss the trade-off between broader applicability and physical interpretability. We have added relevant discussion and citations (including the suggested reference by Cacciola and Bottasso) in Section 3.2 of the revised manuscript.

Regarding the computation of frequency-domain features, this has been clarified in our response to Comment 2-5. We have added detailed explanation of the FFT-based feature extraction process in

Section 3.2, including how the magnitude spectrum is computed and used for calculating the seven frequency-domain features.

Page 6, Table 1: *"F makes me think at frequency, which is obviously not the case. Maybe you can rename it Y? And then its time-domain equivalent could be y."*

Authors' Response: Thank you for this excellent suggestion to improve notation clarity. We agree that F could be misinterpreted as frequency rather than Fourier coefficient magnitude.

We have revised the notation throughout the manuscript: frequency-domain magnitude components are now denoted as Y_i instead of F_i, with corresponding spectral features using subscript Y (e.g., Y_μ for spectral mean, Y_{σ^2} for spectral variance). For the time-domain signal, we have retained the notation a_i rather than adopting y_i, as a explicitly indicates that these are acceleration measurements from vibration sensors, which enhances physical clarity. Table 1 and all related description have been updated accordingly.

Page 6, Line 137: *"Why not using x from the beginning?"*

Authors' Response: Thank you for this question. In Table 1, we chose to present each feature using its conventional statistical or signal processing notation (e.g., σ for standard deviation, $a_{\max}$ for maximum value, Y_μ for spectral mean) rather than generic indexing like $x_1, x_2, ..., x_{19}$. This approach makes the physical and statistical meaning of each feature immediately clear to readers, which is important for understanding what aspects of the vibration signal are being captured. Once these features are extracted and concatenated into a vector for input to the VAE, we adopt the standard machine learning notation $\boldsymbol{x} \in \mathbb{R}^N$ to represent the complete feature vector, following conventional VAE terminology.

Page 7, Line 147: *"This reminds me of the 2 papers by Deepali Singh, where she used Mixture Density Networks to predict fatigue loads. Might be worth a citation."*

Authors' Response: Thank you for this valuable reference suggestion. We have added citations to her papers in the revised manuscript.

Page 7, Line 149: *"No, the VAE is trained to ..."*

Authors' Response: Thank you for this correction. We have revised the sentence to: "The VAE is trained to maximize the evidence lower bound (ELBO) by minimizing the combined loss of reconstruction error and KL divergence."

Page 8, Algorithm 1: *"Is there a for each keyword? For each epoch and batch? Surely this step should be out of the loops. Is it the Hadamard product? Anyway, please write it."*

Authors' Response: Thank you for these careful observations. We have revised Algorithm 1 to address all the issues you identified.

First, regarding the feature extraction step: this should be outside the training loops. In our actual implementation, features are extracted once from all raw signals before training begins. We have added a preprocessing step in the revised algorithm to clarify this.

Second, regarding the loop structure: Yes, the algorithm has two nested loops—an outer loop over epochs and an inner loop over mini-batches within each epoch. This is the standard training procedure for neural networks. The revised algorithm now clearly shows this two-level structure.

Third, regarding the Hadamard product: Yes, the notation represented element-wise multiplication. We have added corresponding description with the standard notation $\odot$ to make this explicit.

The revised Algorithm 1 is shown below and has been updated in the manuscript:

Algorithm 1 Training procedure for the proposed fault detection method

Input: Vibration signal dataset, learning rate α, batch size B, epochs N_{epochs}
Preprocess: Extract features from raw signals to obtain dataset $\mathcal{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$ where each $\boldsymbol{x}_i \in \mathbb{R}^N$ is a feature vector with $N = 266$
for epoch $= 1$ **to** N_{epochs} **do**
    Shuffle $\mathcal{X}$ and partition into mini-batches of size B
    for each mini-batch $\{\boldsymbol{x}_i\}_{i=1}^{B}$ from $\mathcal{X}$ **do**
        $\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i \leftarrow Q_\phi(\boldsymbol{x}_i)$ for $i = 1, \ldots, B$
        $\boldsymbol{z}_i \leftarrow \boldsymbol{\mu}_i + \boldsymbol{\sigma}_i \odot \boldsymbol{\varepsilon}_i$ where $\boldsymbol{\varepsilon}_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
        $\hat{\boldsymbol{x}}_i \leftarrow P_\theta(\boldsymbol{z}_i)$ for $i = 1, \ldots, B$
        Compute batch loss $\mathcal{L} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{KL}}$ over the mini-batch
        Update θ, ϕ using Adam optimizer
    end for
end for
Return: Trained parameters θ^*, ϕ^*

Page 8, Line 175: *"(VAE architecture) An hyper-parameters tuning would have been appreciated."*

Authors' Response: Thank you for this suggestion. The VAE architecture (encoder hidden layers: 128-64-32, latent space: 16) was determined through preliminary experiments and follows established design principles for VAEs: the encoder progressively compresses information through decreasing layer sizes, and the latent dimension is chosen to be substantially smaller than the input dimension ($M \ll N$) to encourage meaningful compression while retaining sufficient representational capacity.

In response to your comment, we have added hyperparameter analysis in the Section 4.4 of the revised manuscript. Specifically, we conducted experiments investigating the impact of latent space dimension on detection performance for aerodynamic imbalance and icing detection. We tested latent dimensions of 2, 4, 8, 16, 32, and 64, evaluating their effect on accuracy, recall, precision, and F_1 score. As shown in Table 5 (included below and in the revised manuscript), latent dimensions of 8 and above consistently achieve perfect performance. The slightly lower performance at dimension 4

for aerodynamic imbalance (99.81% accuracy) suggests that extremely small latent spaces may have insufficient capacity to capture normal operational complexity. The consistent perfect performance across dimensions 8 to 64 demonstrates that the framework is robust to this hyperparameter within a reasonable range. This robustness is particularly valuable for practical deployment, as it reduces the need for extensive hyperparameter tuning and suggests the method can generalize well across different operational scenarios. Additionally, we analyzed the sensitivity of the EWMA smoothing parameter λ in Section 4.4.2, showing performance across values from 0.15 to 0.45. These analyses demonstrate that our hyperparameter choices are well-justified and the framework exhibits robustness across reasonable parameter ranges.

Table 5: Performance comparison of different latent space dimensions for Model 2.

Latent Dimension	Aerodynamic Imbalance				Icing Events			
	Accuracy	Recall	Precision	F_1	Accuracy	Recall	Precision	F_1
2	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
4	99.81%	99.81%	100.00%	99.90%	100.00%	100.00%	100.00%	100.00%
8	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
16	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
32	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
64	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%

We have added a new subsection "Hyperparameter Analysis" (Section 4.5) in the revised manuscript to present these findings.

Page 11, Table 2: *"This table is missing the power."*

Authors' Response: Thank you for this comment. Table 2 lists only the vibration signal channels used for feature extraction and fault detection. While power and other SCADA data are available in the dataset, our framework focuses exclusively on vibration-based monitoring. Power data was used only for preprocessing to identify and remove non-operational periods (Section 4.2), but not as model inputs. Therefore, Table 2 correctly represents the channels used in our analysis.

Page 12, Figure 5: *"Is there also erosion? See for example "Jens Visbech - Aerodynamic effects of leading-edge erosion in wind farm flow modeling""*

Authors' Response: Thank you for this reference and observation. Based on the dataset documentation, the fault events analyzed in this study were specifically roughness tape application (simulating aerodynamic imbalance) and icing, with no explicit indication of blade erosion.

However, the connection you've drawn between erosion and the experiment is precisely the rationale behind the simulated imbalance test. This condition was designed to serve as a practical proxy

for various real-world blade surface degradation phenomena, with leading-edge erosion being an example. Both icing and erosion degrade the aerodynamic profile of the blades, leading to performance loss and altered vibration patterns. Therefore, the framework's successful detection of the roughness tape-induced imbalance strongly suggests its potential applicability for detecting the onset of actual leading-edge erosion, as both would generate similar changes in the turbine's vibration signature.

To better integrate this important context and acknowledge your valuable feedback, we have revised Section 4.1 (Experimental Dataset Description). A new sentence has been added to clarify that the simulated aerodynamic imbalance is representative of real-world surface degradation issues. To support this clarification and credit your suggestion, we have also incorporated a citation to the paper by Visbech et al. Visbech et al. [2023] that you kindly recommended.

Page 13, Line 247: *"mass"*

Authors' Response: Thank you for pointing this out. We have corrected "rotational balance" to "mass balance" in the revised manuscript.

Page 14, Line 255: *"depends"*

Authors' Response: Thank you for this correction. We have revised "depend" to "depends" in the manuscript.

Page 14, Figure 7: *"How have you removed the outliers from the accelerometers listed in Table 2?"*

Authors' Response: Thank you for requesting this clarification. The outlier removal process uses the turbine's power output as a filtering criterion to ensure that only vibration data from valid, healthy operational periods are used. We did not remove outliers from the accelerometer signals directly.

The procedure is as follows: we first identify any 10-minute timestamp where the corresponding power output is anomalous (e.g., zero or negative), as this indicates a non-operational state or a data logging error. Once a timestamp is flagged, the entire corresponding 10-minute block of vibration data from all 14 accelerometer channels is discarded. This approach ensures the integrity of our training and validation datasets.

In the revised manuscript, we have modified Section 4.2.1 to clearly describe this filtering method:

> To ensure that the training and validation datasets accurately represent true healthy operating conditions, the turbine's power output is used as the primary indicator of its operational state. Any timestamp where the corresponding power measurement is anomalous (e.g., zero or negative values) is flagged as invalid. Subsequently, for any timestamp flagged, the associated vibration data is discarded from the analysis. Discarding these timestamps ensures that the dataset more accurately represents the normal operating con-

ditions of the wind turbines.

Page 15, Line 274: *"This sounds like you are looking at the raw sensor output, before applying gain and offset to convert to m/s^2. Are the gain and offset for this database written anywhere?*

Authors' Response: Thank you for this comment. According to the dataset documentation, the vibration data appear to be pre-processed sensor outputs rather than raw voltage signals. The specific gain and offset calibration parameters used to convert raw sensor readings to acceleration units (m/s^2) are not explicitly provided in the dataset documentation.

The exclusively positive values we observed are likely due to a DC offset in the data acquisition or processing chain. Our detrending approach (subtracting the mean from each sample) effectively removes any constant offset, ensuring the signals oscillate around zero regardless of the original calibration. Therefore, while the absolute calibration values are unknown, our framework is inherently robust to this type of data artifact. The features used for model training are based on the dynamic behavior of the signal, which is unaffected by this preprocessing step.

Page 17, Line 309: *"Good job for predicting the upcoming failure, but I doubt that much can be done in 2.5 hours."*

Authors' Response: We appreciate this insightful comment and agree with your perspective. You are correct that our original statement overstated the practical value of the 2.5-hour lead time for maintenance planning, and we thank you for this important clarification.

In the revised manuscript, we have modified Section 4.3.1 to provide a more balanced and realistic assessment:

> The framework demonstrates the capability to provide early warning for this progressive fault. As shown in Figure 9, an alarm is triggered at 14:32, providing a 2.5-hour lead time before the actual turbine shutdown at 17:12. While this lead time is insufficient for comprehensive maintenance planning in operational scenarios, the result validates the framework's ability to detect early signs of pitch drive degradation before complete failure occurs.

Page 17, Figure 9: *"As much as I want to like this figure, and the next, I can't help but think that this threshold is based on a conveniently short time history. What will happen if I analyze a whole year of healthy data? Will the threshold get higher and then failures will be hidden? It feels like you have set this threshold exactly to detect this failure."*

Authors' Response: Thank you for this comment. We want to clarify that that our procedure was designed to prevent selection bias. The training and validation sets were established based on a strict

chronological split, using all available data that was documented as 'healthy' in the operational logs prior to the fault event. The threshold was then automatically set as the maximum health index value from this healthy data. This process simulates a real-world monitoring scenario where future data is unknown and was not "set exactly to detect this failure" with the benefit of hindsight.

We acknowledge that the available pre-fault healthy data for this specific event spans a relatively short period. This was, however, the entirety of the continuous healthy data provided in the public dataset immediately preceding the failure.

Regarding your question about a longer baseline, our framework's robustness is supported by two key points:

- A key strength of a robust health index is that it should remain stable during all healthy operational phases. If a model has successfully learned the true patterns of normal operation, then additional healthy data—even from a longer period with more environmental and operational variety—should not significantly elevate the HI baseline. Our framework is designed to achieve this stability through robust feature engineering and EWMA smoothing, which filters out transient noise. Therefore, introducing a longer period of healthy data should primarily serve to increase confidence in the established threshold, rather than substantially raising it.
- This principle is strongly supported by the evidence in our analysis. As shown in Figure 9, there is a very large margin of safety, with the fault-induced HI (peaking over 0.06) being around double the threshold (0.035) derived from the available healthy data. This significant separation demonstrates that the detection is not a borderline case. It provides strong evidence that the fault would be clearly detected even in the presence of minor baseline fluctuations that a longer dataset might introduce.

Page 18, Figure 10: *"Why this figure has a different color scheme? So, the threshold for the health index doesn't only depend on the data, but also on the machine learning model. Feels like it's extremely specific and carefully tuned to detect exactly what you want to see."*

Authors' Response: Thank you for these observations. Regarding the color scheme, we apologize for the inconsistency. We have standardized the colors across all figures in the revised manuscript for better visual coherence and readability.

Regarding your critical point about the threshold: the threshold is dependent on the specific machine learning model being used. This is an important and inherent characteristic of comparing different anomaly detection algorithms. Each model (AE, VAE, Deep SVDD, etc.) has a different architecture and objective function, causing it to learn a unique representation of the "normal" data. Consequently, each model produces anomaly scores (e.g., reconstruction errors or decision function values) on a different scale. It is therefore methodologically necessary to establish a unique, model-

specific threshold for each one. Applying a single, universal threshold across all models would be an invalid comparison.

The crucial point, which we have now made more explicit in the manuscript, is that while the value of the threshold is model-specific, the procedure for determining it was systematic and applied consistently to all methods. For every model, the threshold was automatically set as the maximum EWMA value from the exact same set of pre-fault healthy data. This rigorous process was performed before evaluating any of the models on the test data, ensuring a fair comparison and preventing any "tuning" based on the fault outcomes. We have added transparency about this methodology in Section 4.1 (see detailed response to Main Comment 2-4) to clarify that while the threshold is model-dependent, the selection process was rigorous and not arbitrarily tuned to the test results.

Page 18, Line 319: *"What's going to happen if I apply your algorithm over 10 years? Will it start flagging damages as soon as there is enough blade erosion? That is, will the alarm stay active for years?"*

Authors' Response: Thank you for this question about long-term deployment behavior.

You are correct that if blade erosion develops gradually over years, the framework would detect the anomaly once erosion-induced vibration changes exceed the learned baseline. The key concern you raise is whether the alarm would remain continuously active, which would indeed occur under the current design. This highlights an important characteristic of the framework: it operates as an anomaly detector that flags deviations from the learned healthy baseline, but does not distinguish between newly detected anomalies and persistent, unresolved conditions.

This behavior is not unique to our framework but represents a fundamental challenge for static baseline anomaly detection systems. In practical industrial deployment, such systems are typically integrated with maintenance management platforms that track identified issues, manage alarm states, and suppress redundant notifications for known conditions. Additionally, for gradual degradation that remains within acceptable operational limits, periodic model retraining can update the baseline to reflect the current state while maintaining sensitivity to new or accelerating faults.

We acknowledge this limitation in the revised Conclusion section, noting that long-term deployment scenarios would benefit from integration with alarm management systems and periodic baseline adaptation strategies to distinguish between persistent degradation and emerging faults.

Page 19, Figure 11: *"(Figure 11) What happened here? It looks like the blade is fine again."*

Authors' Response: Thank you for this observation. We understand Figure 11 may appear to suggest the blade returned to normal condition, but this is not the case. The aerodynamic imbalance (induced by roughness tape) was continuously present throughout the entire period from December 08, 2022, to January 21, 2023. The variation in HI values does not indicate recovery but rather reflects

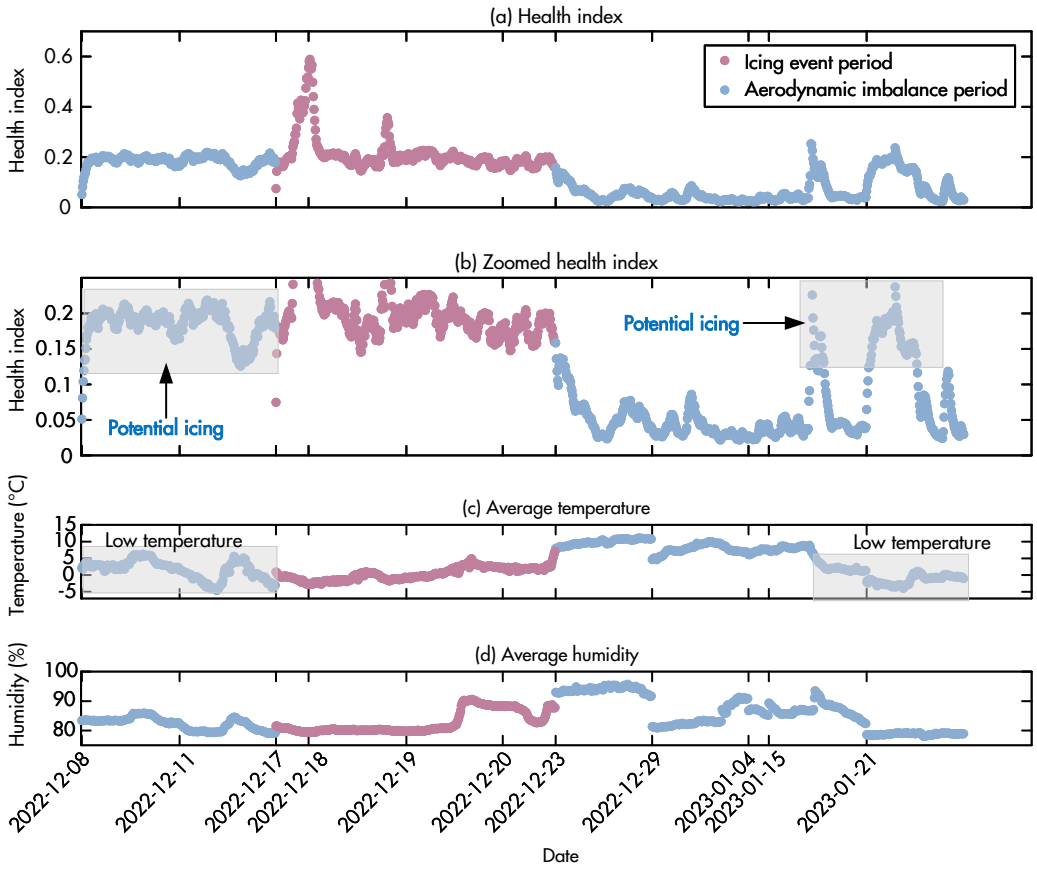

Figure 15: (a) Comparison of HI values for icing events and aerodynamic imbalance periods, (b) zoomed HI values, (c) temperature data trends, and (d) humidity data trends during the monitoring period.

the potential co-occurrence of icing with the aerodynamic imbalance during certain sub-periods.

As illustrated in Figure 15 and discussed in Section 4.6, the HI values show distinct patterns:

- **December 8-11, 2022**: Elevated HI values comparable to confirmed icing events, despite no icing being documented in the dataset for this period. Environmental data (low temperature and high humidity in Figure 15 (c-d)) suggest conditions conducive to ice formation. We hypothesize that natural icing may have occurred during this period in combination with the roughness tape, producing higher HI values.
- **December 17-20, 2022**: Documented icing period with high HI values, representing the combined effect of confirmed icing and roughness tape.
- **January 15 and 21, 2023**: Additional HI elevations coinciding with low temperature periods (Figure 15(c)), suggesting potential intermittent icing episodes during the aerodynamic imbalance period.
- **Other periods**: Lower but still elevated HI values (above threshold), reflecting aerodynamic imbalance from roughness tape alone.

Crucially, the HI values never return to the baseline (below threshold) during the entire period,

confirming continuous fault presence. The variation in HI magnitude reflects whether icing is co-occurring with the aerodynamic imbalance, not the resolution of the fault. Higher HI values occur when both fault types are present simultaneously, while lower values correspond to aerodynamic imbalance alone.

Our framework successfully identified these patterns, detecting not only the documented icing event but also potential undocumented icing episodes based on HI signatures and environmental correlation. This demonstrates the framework's capability to detect complex, overlapping fault conditions. We have clarified this interpretation in Section 4.6 to emphasize the continuous nature of the fault condition throughout the monitoring period.

Page 19, Line 336: *"This conclusion is based on exactly 1 measurement champaign."*

Authors' Response: You are absolutely correct. This is an important limitation addressed in our response to Main Comment 2-2. In the revised manuscript, we have added explicit acknowledgment of this limitation and directions for future work in the Conclusion section:

> While the framework demonstrates promising detection capability, several limitations exist. The models were trained on data from a single turbine during specific measurement campaigns, and generalization to other turbines or extended operational periods requires further validation. The framework provides anomaly detection without detailed fault diagnosis, and the use of general statistical features limits physical interpretability compared to wind turbine-specific features. Additionally, as a static baseline approach, the framework would require periodic retraining or integration with alarm management systems for long-term deployment to handle gradual turbine aging while maintaining sensitivity to new faults. Future work should address these limitations through validation across multiple turbines and extended operational periods, extension to fault localization and identification capabilities, exploration of hybrid approaches combining general features with physics-informed characteristics, and investigation of adaptive learning strategies for sustained operational deployment.

Page 21, Table 4: *"0.49 should be 49% right?"*

Authors' Response: Thank you for this question. The value 0.49 represents 49% (expressed as a decimal fraction between 0 and 1). We have revised the table to use percentage format consistently (e.g., 49% instead of 0.49) for better clarity and readability in the revised manuscript.

Page 23, Line 380: *"It would be nice to have an idea of how long it takes to train the proposed model."*

Authors' Response: Thank you for this suggestion. We have included training times in Section

4.5 (Computational Efficiency) and Table 7. As stated in the text, model training times vary by task reflecting dataset size: 8.17 seconds for pitch drive failure detection (106 training timestamps), and 1.61 seconds for both aerodynamic imbalance and icing detection (17 training timestamps). All models were trained for 100 epochs on an NVIDIA RTX 4060 GPU.

Page 24, Table 6: *"It takes less time to train on some days than to predict one 10 minutes dataset?"*

Authors' Response: Thank you for this observation. The training time appears shorter than inference time due to the small dataset size for aerodynamic imbalance and icing detection (only 17 timestamps). Training for 100 epochs on such limited data takes only 1.61 seconds. The inference time (2.56 seconds per 10-minute dataset) includes not just model prediction but also feature extraction (2.06 seconds), model loading, and HI computation, which explains why it appears longer.

Page 24, Table 6: *"(100 epochs) This seems short. How does the MSE look like?"*

Authors' Response: Thank you for this question regarding training convergence. While 100 epochs may appear short, our experiments show that both models converge effectively within this range.

To demonstrate this, we have added training convergence curves to the revised manuscript in Appendix A (Figure A1). As shown in the figure, the total loss decreases rapidly in the first 10-30 epochs and stabilizes after approximately 40-80 epochs for both models. The loss curves plateau well before reaching 100 epochs, indicating that the models have successfully learned the normal operational patterns without requiring additional training iterations.

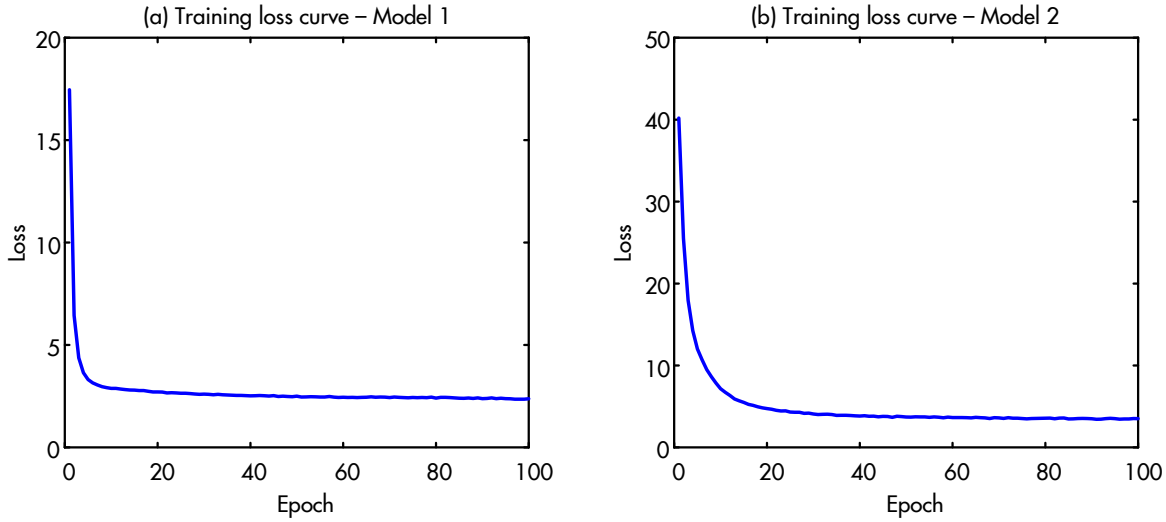

Figure A1: Training convergence curves for (a) Model 1 (pitch fault detection) and (b) Model 2 (aerodynamic imbalance and icing detection), showing the total loss over 100 training epochs.

The smooth convergence and stable plateau demonstrate effective learning and proper regularization of the latent space. Extended training beyond 100 epochs showed no further improvement, confirming that 100 epochs is sufficient for this application.

We would like to thank you once again for your positive and constructive review of our work, which has been invaluable in improving the quality of this paper. We have thoroughly addressed your comments and queries and sincerely hope that these revisions will receive your approval.

Response to Reviewer #3

Overall Evaluation: Thanks to the authors for a very clear and well-written paper. I enjoyed reading it. I only have a few minor comments related to the method.

Authors' Response: We are very grateful for your detailed and positive evaluation of our manuscript. Your constructive feedback has been instrumental in helping us improve the quality of the paper, and we have thoroughly revised the manuscript based on your specific suggestions. We believe that the changes have significantly enhanced the completeness and clarity of the manuscript.

We have carefully addressed your comment as detailed below.

Comment 3-1: It would be useful to clearly explain in the overview the motivation for using a probabilistic model for this application and not a deterministic approach, which is often simpler. It is briefly mentioned in Sec. 3.3, but could appear earlier in the text. You could highlight some other benefits, such as better generalization, being less prone to overfitting, dealing with noise in the data, etc. In relation to that, could you comment in the result section on what it is about VAE that makes it a robust tool and performs better than the other methods you have compared it to?

Authors' Response: Thank you for this insightful suggestion. Following your suggestion, we have revised the manuscript by adding explicit discussion of VAE advantages earlier in the text.

In Section 2.2 (Related Work): We added an explanation that VAEs are particularly suitable for wind turbine applications because vibration signals exhibit inherent variability even during healthy operation due to load fluctuations, speed variations, and environmental factors. Unlike deterministic autoencoders that learn fixed mappings and may overfit, VAEs model underlying distributions, providing better generalization and noise robustness.

In Section 4.3.4 (Performance metrics): We added a brief discussion noting that the VAE's probabilistic framework and KL regularization contribute to its stable performance with zero false alarms. This stability is particularly evident in aerodynamic imbalance and icing detection, which achieved perfect detection despite limited training data.

These revisions clarify the theoretical motivation and connect it to the observed experimental advantages.

Comment 3-2: Some more details on the feature extraction process would be appreciated, especially for the frequency domain terms.

Authors' Response: Thank you for requesting this clarification. We apologize for the insufficient detail in the original manuscript.

The FFT is applied to each 10-second sample (2000 data points at 200 Hz sampling rate). Specifically, the FFT converts the time-domain vibration signal $\{a_i\}_{i=1}^{n}$ into the frequency domain, yielding complex-valued frequency components $\{Y_i\}_{i=1}^{m}$, where m is the number of frequency bins. The mag-

nitude spectrum $|Y_i|$ is then computed, and the normalized power distribution $p_i = \frac{|Y_i|}{\sum_{k=1}^{m} |Y_k|}$ is used for spectral entropy calculation. The seven frequency-domain features (spectral mean, variance, standard deviation, entropy, energy, skewness, and kurtosis) are computed based on this magnitude spectrum, as defined in Table 1.

We have added the following clarification in Section 3.2 (Feature Extraction) of the revised manuscript:

> For frequency-domain feature extraction, the fast Fourier transform (FFT) is applied to each sample (2000 data points, corresponding to 10 seconds of operation at 200 Hz sampling rate). The FFT converts the time-domain vibration signal $\{a_i\}_{i=1}^{n}$ into the frequency domain, yielding complex-valued frequency components $\{Y_i\}_{i=1}^{m}$, where m is the number of frequency bins. The magnitude spectrum $|Y_i|$ is computed, and the normalized power distribution $p_i = \frac{|Y_i|}{\sum_{k=1}^{m} |Y_k|}$ is used for spectral entropy calculation. The seven frequency-domain features, including spectral mean, variance, standard deviation, entropy, energy, skewness, and kurtosis, are then computed based on this magnitude spectrum, as defined in Table 1.

Comment 3-3: It would be good to differentiate between the vector and scalar notations by using bold for vectors, for example. Introduce the feature, output, and the latent space with mathematically rigorous notations (For example, instead of x, it is more complete to introduce it as $\boldsymbol{x} \in \mathbb{R}^N$). Similarly, $\boldsymbol{z} \in \mathbb{R}^M$, where M<N.

Authors' Response: Thank you for this valuable suggestion to improve the mathematical rigor and clarity of our notation. We have revised the notation throughout the manuscript as follows:

- Feature vectors are now denoted as $\boldsymbol{x} \in \mathbb{R}^N$, where $N = 266$ represents the total number of features (19 features from each of 14 channels).
- Latent vectors are denoted as $\boldsymbol{z} \in \mathbb{R}^M$, where $M = 16$ is the latent space dimension, with $M \ll N$.
- Reconstructed feature vectors are denoted as $\hat{\boldsymbol{x}} \in \mathbb{R}^N$.
- The encoder mapping is expressed as $Q_\phi : \mathbb{R}^N \to \mathbb{R}^M \times \mathbb{R}^M$ (outputting mean and standard deviation vectors).
- The decoder mapping is expressed as $P_\theta : \mathbb{R}^M \to \mathbb{R}^N$.
- Scalar quantities (such as individual time-domain signal points a_i, frequency components Y_i, and scalar loss values) remain in regular font to distinguish them from vectors.

Additionally, we have updated the frequency-domain notation in Table 1 from F_i to Y_i (in response to Reviewer 2's comment) to avoid confusion with frequency. In the feature extraction phase (Table 1), a_i represents individual scalar sampling points in the time-domain vibration signal, and Y_i represents

individual frequency components. These scalar notations are retained in regular font. When the extracted features are concatenated into the input for the VAE, they form the feature vector $\boldsymbol{x} \in \mathbb{R}^N$, which is consistently represented in bold throughout the modeling sections.

These revisions have been implemented in Section 3.2 (Feature Extraction), Section 3.3 (Variational Autoencoder), Table 1, Algorithm 1, and all related equations and descriptions throughout the manuscript.

Comment 3-4: Line 146: Reference variational inference.

Authors' Response: Thank you for this suggestion. We have added appropriate references to variational inference at the relevant location in the revised manuscript.

We would like to thank you once again for your positive and constructive review of our work, which has been invaluable in improving the quality of this paper. We have thoroughly addressed your comments and queries and sincerely hope that these revisions will receive your approval.

References

Marta Bertelè, Carlo L Bottasso, and Stefano Cacciola. Automatic detection and correction of pitch misalignment in wind turbine rotors. *Wind Energy Science*, 3(2):791–803, 2018.

Stefano Cacciola, I Munduate Agud, and Carlo Luigi Bottasso. Detection of rotor imbalance, including root cause, severity and location. In *Journal of Physics: Conference Series*, volume 753, page 072003. IOP Publishing, 2016.

Yuanchang Chen and D Todd Griffith. Blade mass imbalance identification and estimation for three-bladed wind turbine rotor based on modal analysis. *Mechanical Systems and Signal Processing*, 197:110341, 2023.

Carl Doersch. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*, 2016.

Riccardo Riva, Stefano Cacciola, and Carlo Luigi Bottasso. Periodic stability analysis of wind turbines operating in turbulent wind conditions. *Wind Energy Science*, 1(2):177–203, 2016.

Jens Visbech, Tuhfe Göçmen, Özge Sinem Özçakmak, Alexander Meyer Forsting, Ásta Hannesdóttir, and Pierre-Elouan Réthoré. Aerodynamic effects of leading edge erosion in wind farm flow modeling. *Wind Energy Science Discussions*, 2023:1–24, 2023.