Structural Engineering and Mechanics, *Vol. 92, No. 2 (2024) 121-131*
https://doi.org/10.12989/sem.2024.92.2.121

Fault detection in blade pitch systems of floating wind turbines utilizing transformer architecture

Seongpil Cho[1], Sang-Woo Kim[1] and Hyo-Jin Kim*[2]

[1]*Department of Aeronautical and Astronautical Engineering, Korea Aerospace University, 76 Gonghangdaehak-ro, Deokyang-gu, Goyang, Gyeonggi 10540, Republic of Korea*
[2]*Department of Korean Medical Science, Kyung Hee University, 26 Kyungheedae-ro, Dongdaemun-gu, Seoul 02447, Republic of Korea*

(*Received August 4, 2024, Revised September 25, 2024, Accepted September 27, 2024*)

Abstract. This paper proposes a fault detection method for blade pitch systems of floating wind turbines using transformer-based deep-learning models. Transformers leverage self-attention mechanisms, efficiently process time-series data, and capture long-term dependencies more effectively than traditional recurrent neural networks (RNNs). The model was trained using normal operational data to detect anomalies through high reconstruction losses when encountering abnormal data. In this study, various fault conditions in a blade pitch system, including environmental load cases, were simulated using a detailed model of a spar-type floating wind turbine, the data collected from these simulations were used to train and test the transformer models. The model demonstrated superior fault-detection capabilities with high accuracy, precision, recall, and F1 scores. The results show that the proposed method successfully identifies faults and achieves high-performance metrics, outperforming existing traditional multi-layer perceptron (MLP) models and long short-term memory-autoencoder (LSTM-AE) models. This study highlights the potential of transformer models for real-time fault detection in wind turbines, contributing to more advanced condition-monitoring systems with minimal human intervention.

Keywords: blade pitch system; fault detection; floating wind turbine; prognostics and health management; sequential data; transformer

1. Introduction

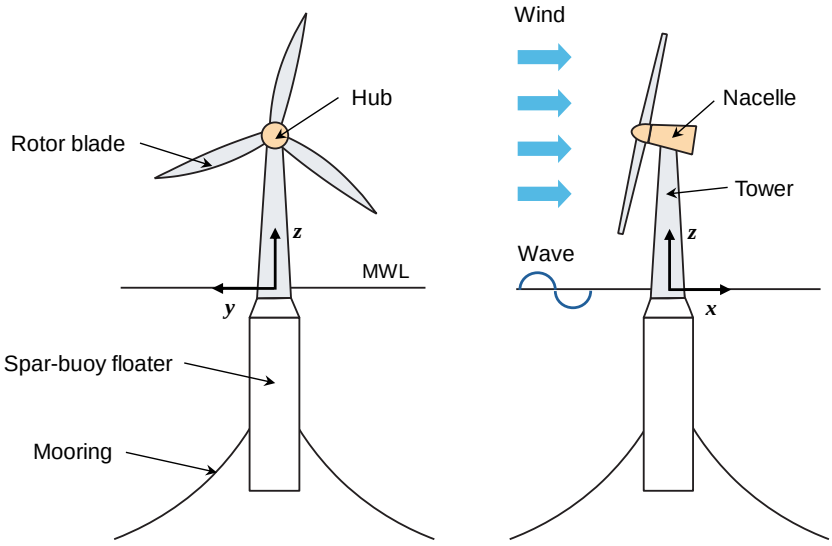
Offshore wind turbines are often subjected to harsh ocean environments characterized by wind, waves, and current loads, leading to significant maintenance costs, typically representing approximately 25-30% of the total life-cycle costs of turbines (Dinwoodie *et al.* 2013). The primary components of offshore wind turbines include towers, floaters, blades, hubs, rotor shafts, gearboxes, generators, yaw systems, and blade-pitch systems. While faults in any of these components can significantly affect the lifespan of a wind turbine, the blade pitch system is particularly prone to faults, considerably influencing the turbine's failure rate and downtime (Gayo 2011, Carroll *et al.* 2016). Faults in the blade pitch system alter the aerodynamic load on the blades, affecting the response of the support structure and tower (Cho *et al.* 2018, 2020), which can change the system characteristics of the turbine, impacting both the structural safety and power generation efficiency. Therefore, early detection and diagnosis of faults in the system are crucial for preventing extensive downtime and serious system failures and mitigating potential risks, accidents, and long-term damage to the turbine (Isermann 2006).

Recent fault detection and diagnosis techniques predominantly rely on data-driven methods, which offer the advantage of detecting and diagnosing issues using only the data measured from the system (Kusiak and Verma 2011, Choi 2014, Bakdi *et al.* 2019). These fault detection and diagnosis techniques utilize a series of measurement data collected by sensors located in various components of wind turbine systems (Hameed *et al.* 2009, Márquez *et al.* 2012, Zhou *et al.* 2016). They apply statistical techniques, such as the multi-variable statistical approach (de Andrade Melani *et al.* 2021, Harrou *et al.* 2023, Shin *et al.* 2023), and deep learning theories, including multi-layer perceptrons (MLPs) (Zaher *et al.* 2009, Kusiak and Verma 2012, Wang *et al.* 2016, Chang *et al.* 2019, Cho *et al.* 2021), convolutional neural networks (CNNs) (Bach-Andersen *et al.* 2018, Jiang *et al.* 2018, Nguyen *et al.* 2021, Rahimilarki *et al.* 2022), radial basis functions (RBFs) (Dervilis *et al.* 2014), autoencoders (AEs) (Jiang *et al.* 2017, Finotti *et al.* 2022, Liu *et al.* 2023), and recurrent neural networks (RNNs) (Cho *et al.* 2021, Cui *et al.* 2021, Dhibi *et al.* 2022) to detect faults at the early stage.

Sensors output time-series signal data, making them suitable for RNNs because they can model sequential data utilizing information from previous time steps. However, RNNs struggle to capture long-term dependencies because of the vanishing gradient problem. To address this issue, long short-term memory (LSTM) (Hochreiter *et al.* 1997, Lei *et al.* 2019) and gated recurrent unit (GRU) (Cho *et al.* 2014) networks designed using memory cells that manage long-term dependencies by selectively including or excluding information. LSTMs process data sequentially,

*Corresponding author, Ph.D.
E-mail: hjinkim@khu.ac.kr

ISSN: 1225-4568 (Print), 1598-6217 (Online)



(a) (b)

Fig. 1 Schematic view of a spar-type floating wind turbine

retaining information over time, which helps capture dependencies within time-series data (Malhotra *et al.* 2015). However, this sequential processing results in slower training and inference owing to limited parallelism. Despite their gating mechanisms, LSTMs and GRUs struggle with long sequences and complex dependencies.

In contrast, transformers (Vaswani *et al.* 2017, Schlag *et al.* 2021, Wu 2023) enhance sequence processing using a self-attention mechanism that allows them to consider all the positions of the input sequence simultaneously. This parallel-processing capability significantly accelerates both training and inference, making transformers highly efficient for modern hardware. Unlike LSTMs, transformers do not process data sequentially, enabling them to capture long-range dependencies more effectively. Each layer of the transformer model employs multiple self-attention heads that independently focus on different parts of the sequence, thereby providing a comprehensive understanding of the entire input. This architecture facilitates faster computation and enhances the model's ability to scale with data and model size.

In condition monitoring, the acquisition and labeling of abnormal data pose significant challenges. Fault data are often rare and irregular, complicating the development of detection models. Additionally, ambiguity in labeling data, particularly when components show early signs of abnormality, can change the data characteristics. Therefore, focusing on fault detection using abundant normal data is crucial. The transformer can capture long-term dependencies, facilitating the real-time processing of time-series data. This method is trained solely on normal sequence data, similar to the AE or LSTM-AE (Nguyen *et al.* 2021, Cho 2024) approaches. When feeding an abnormal sequence into the trained model, a high reconstruction loss signals a fault—flagged if it exceeds a predefined threshold. This technique allows effective fault detection by training on normal data without requiring fault or failure data.

This study discusses the implementation of a transformer-based scheme for fault detection in the blade pitch system of spar-type floating wind turbines. Section 2 provides detailed information on the modeling and methodologies for wind turbines, and Section 3 outlines the fault modes, data collection procedures and steps involved in the structure of the proposed method. In Section 4, the concept of the transformer and learning procedure are described, with the performance of the proposed algorithm compared with that of conventional models. The final section explores future research directions.

Table 1 Properties of the National Renewable Energy Laboratory (NREL) 5-MW wind turbine (Jonkman *et al.* 2009)

Rated power, MW	5
Rotor orientation and configuration	Upwind, three blades, horizontal axis
Rotor diameter, m	126
Hub height from the mean water level, m	90
Cut-in, rated, cut-out wind speed, m/s	3, 11.4, 25
Cut-in, rated rotor speed, °/s	41.4, 72.6
Max pitch rate, °/s	8
Gearbox ratio	97

2. Case study model and methodology

2.1 Floating wind turbine and blade pitch systems

The wind turbine, supported by a spar-type floating wind turbine, was modeled to include a rotor, tower, nacelle, floater, and mooring line. As illustrated in Fig. 1, this numerical model is based on the National Renewable Energy Laboratory (NREL) 5-MW offshore wind turbine (Jonkman *et al.* 2009), OC3-Hywind floater (Jonkman 2010), and three catenary mooring cables. Tables 1 and 2 present the characteristics of the NREL 5 MW wind turbine

Table 2 Properties for the OC3-Hywind floater (Jonkman 2010)

Water depth, m	320
Draft, m	120
Diameter above taper, m	6.5
Diameter below taper, m	9.4
Center of mass, m	(0, 0, −89.9115)
Mass, including ballast, kg	7.466×10^{6}
Mass moment of inertia (I_{xx} and I_{yy}), kg·m^2	4.229×10^{9}
Mass moment of inertia (I_{zz}), kg·m^2	1.642×10^{8}

and the OC3-Hywind floater model, respectively.

Specifically, the blade pitch system plays a crucial role in the rotor system by adjusting the aerodynamic load and rotor speed. Among the different types of blade pitch systems, which include both electric and hydraulic systems, hydraulic pitch systems are particularly well-suited for large wind turbines (5-10 MW) that experience high aerodynamic loads due to their high stiffness and appropriate damping characteristics (Lu *et al.* 2009). The hydraulic pitch system comprises a hydraulic pump, accumulator, directional control valve, fluid storage tank, and hydraulic cylinder. Fig. 2 represents a hydraulic system used for controlling the pitch of a wind turbine blade. The system operates through a hydraulic cylinder, which adjusts the blade's pitch by moving a piston. Pressurized fluid is directed into either side of the cylinder, controlling the piston's movement, and thus adjusting the angle of the wind turbine blade. The pressures on both sides of the piston, p_A and p_B, control the direction of piston movement, while the system ensures precise blade positioning through a rotary encoder that measures the blade's angular position.

The system also includes a directional control valve, which determines the flow of hydraulic fluid into the A-side or B-side of the cylinder, based on an input signal, u. This valve's function is crucial for making real-time adjustments to the blade angle, responding to varying wind conditions. A linear variable differential transformer (LVDT) sensor measures the piston's displacement, offering feedback to the control system to help regulate blade position accurately. This feedback ensures that the pitch control system can react promptly and with precision to external environmental changes.

Additionally, the hydraulic circuit contains a pump, accumulator, and pressure relief valve. The pump supplies the system with hydraulic pressure, and the accumulator stores this pressure for consistent piston movement. The pressure relief valve ensures system safety by diverting excess pressure back to the reservoir when needed. The overall purpose of this system is to regulate wind turbine performance by adjusting the blade pitch, either to optimize energy capture or reduce aerodynamic forces during high winds, protecting the turbine from potential damage.

2.2 Multi-physics simulations and environmental conditions

A dynamic response analysis of a floating wind turbine model was performed using Simo-Riflex, a code for time-domain numerical simulations involving aero-hydro-servo-elastic simulations. Simo (SINTEF Ocean 2018a) calculates the hydrodynamic forces and moments acting on a floating body based on potential flow theory and Morrison viscous drag. Riflex (SINTEF Ocean 2018b) performs a structural analysis of elastic components such as blades, shafts, towers, and mooring systems using a beam-element-based finite element method. The nonlinear beam elements in Riflex account for large deformations, and the blades and tower can be modeled with varying cross-sections along their lengths. The time-domain analysis involved a coupled simulation with Simo, incorporating mass, damping, and stiffness matrices. The external loads include the aerodynamic forces, wave loads, and torque from the rotor. Riflex also calculates the aerodynamic forces and moments on the blades using blade element momentum theory, including tower shadow effects and dynamic stall. Consequently, Simo-Riflex integrates the structural dynamics, hydrodynamics, and aerodynamic models with the baseline control system, which includes torque and blade pitch controllers, to conduct the simulation. Fig. 3 illustrates the data exchange between different simulation tools (Simo, Riflex, TurbSim) and a Java-based control system used for modeling and controlling a floating wind turbine. Simo is responsible for simulating the hydrodynamic behavior of the floating platform, accounting for wave history and floater hydrodynamic loads. These motions are then passed to Riflex, which performs structural analysis to assess the platform's response. At the same time, TurbSim provides wind field data to Riflex, allowing it to incorporate aerodynamic forces and simulate the turbine's behavior under changing wind conditions, particularly in terms of rotor speed and blade pitch angle.

The control system, written in Java, consists of the Baseline controller and the Blade pitch system, which interact with Riflex. Riflex sends the current rotor speed and blade pitch angle data to the Baseline controller, which

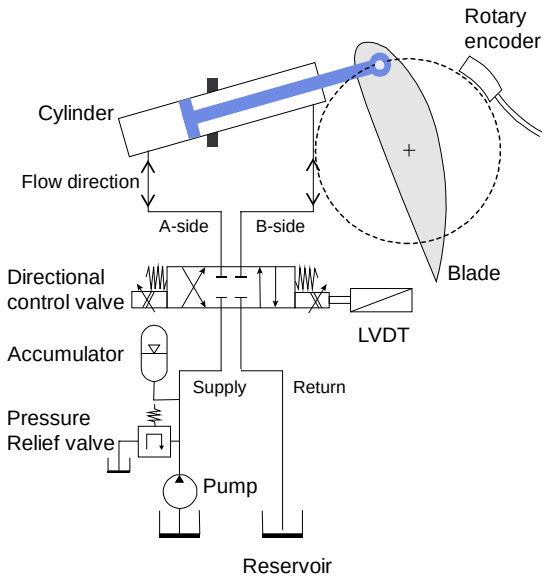


Fig. 2 Schematic view of a hydraulic blade pitch system (Cho *et al.* 2020)

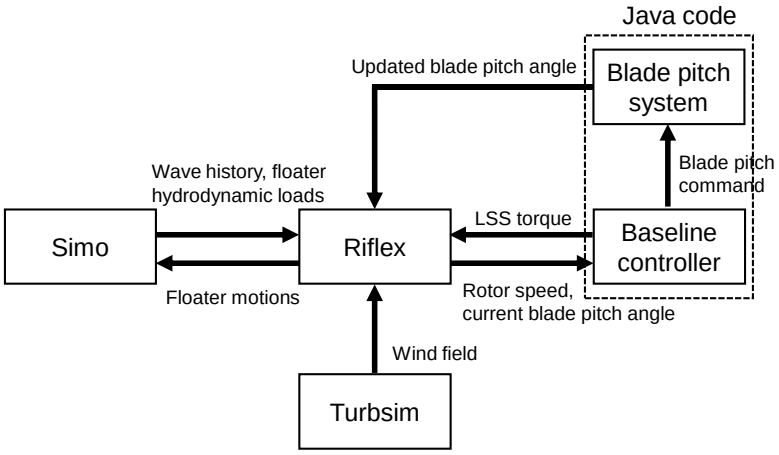


Fig. 3 Data transmission between Simo-Riflex and controller (SINTEF Ocean 2018a, 2018b, Cho *et al.* 2020)

Table 3 Load cases based on winds and waves

Load case	U_w(m/s)	H_s(m)	T_p(s)
1	14	3.58	10.27
2	16	3.97	10.44
3	17	4.17	10.53
4	19	4.58	10.72
5	20	4.8	10.82
6	22	5.23	11.02

then processes this information and generates blade pitch commands. These commands are passed to the Blade pitch system, which adjusts the blade pitch accordingly. The updated blade pitch angle is fed back to Riflex, creating a feedback loop that continually adjusts the turbine's blades and monitors low speed shaft (LLS) torque, ensuring optimal performance based on environmental conditions and platform dynamics.

The components from Simo-Riflex and the baseline controller work together to optimize the operation of the floating wind turbine. The hydraulic pitch control system directly manipulates the blade angle to regulate turbine performance and structural loads. Meanwhile, the baseline controller ensures that the turbine adapts dynamically to its floating environment by considering both aerodynamic (wind) and hydrodynamic (wave and floater motion) forces. The integration of these systems allows the turbine to operate efficiently and safely, even in the unpredictable and highly dynamic conditions of offshore environments.

Simulations of the dynamic responses of a floating wind turbine were conducted under various ocean environmental loads with the corresponding wind and wave conditions. Turbulence modeling for wind loads was performed using TurbSim (Jonkman and Kilcher 2012), based on the Kaimal turbulence model and following the guidelines of IEC 61400-1 (IEC 2005a) and IEC 61400-3 (IEC 2005b). For irregular wave loads, the Joint North Sea Wave Project (JONSWAP) spectrum (Hasselmann *et al.* 1973) was used in relation to wind speed in the Statfjord area of the North Sea (Johannessen *et al.* 2001) to determine the peak period and significant wave height. These environmental conditions were applied to model a floating wind turbine to simulate the real operational environment. Table 3 describes the relationship between the wind speed and wave loads for each load case, where U_w is the wind speed, T_p is the peak period, and H_s represents the significant wave height.

Table 4 Fault description in valves (Cho *et al.* 2020)

Fault number	Fault name	Consequence
1	Excessive friction (VEF)	Response delay
2	Slit lock on spool (VSL)	Blade pitch runaway
3	Wrong voltage (VWV)	Response delay
4	Short Circuit (VSC)	Actuator stuck

3. Fault modes and data acquisition of blade pitch systems

3.1 Fault description of the blade pitch system

In floating wind turbines, the blade pitch system plays a critical role in controlling the angle of the blades relative to the wind, thereby optimizing power generation and ensuring turbine stability. However, faults in the blade pitch system can lead to significant operational issues. Specifically, actuator faults in this system can cause rotor imbalance and misalignment of the blades, which can reduce the turbine's efficiency and lead to mechanical strain.

Several key components within the blade pitch system can be prone to failure, particularly the hydraulic actuator, which adjusts the blade pitch through the control of valves and oil circulation. Failures related to oil contamination, valve blockages, and sludge buildup are responsible for a significant proportion (37.3%) of these failures (Carroll *et al.* 2016). These contaminants increase friction and impede the movement of the actuator, potentially leading to rotor instability.

In particular, valve faults within the blade pitch actuator can alter the dynamic characteristics of the system, affecting both its transient and steady-state performance under normal or challenging wind conditions. This study focuses on four common types of faults occurring in the control valve of a hydraulic blade pitch system, as outlined in Table 4 (Cho *et al.* 2020).

The causes of these faults often stem from oil

Table 5 Measurement parameters for monitoring system Germanischer Lloyd (2010)

Sensor Measurement	Unit
Power	W
Acceleration (Nacelle)	m/s^2
Displacement (Nacelle)	m
Rotor speed	rad/s
Thrust	N
Blade pitch angle (three blades)	rad

contamination in the hydraulic actuator or improper filtering. For example, the buildup of sludge on the valve spool reduces the clearance between the spool and the valve body, leading to excessive friction (Fault 1). If this sludge hardens over time, it can cause the valve to seize completely (Fault 2). In some cases, damage to the solenoid controlling the valve can result in wrong voltage application (Fault 3), while excessive current flowing through a malfunctioning solenoid can cause a short circuit, rendering the actuator non-functional (Fault 4).

To study the effects of these faults on the overall dynamic behavior of the wind turbine, each fault mode described in Table 4 was simulated independently in the blade pitch system model. These fault models were implemented using Java code, interfacing with the Simo-Riflex simulation software, which is commonly used to analyze offshore wind turbines. Through mathematical modeling (Cho *et al.* 2018, 2020), the distinct characteristics of each fault were analyzed and incorporated into the simulations, providing valuable data for training predictive maintenance systems.

3.2 Methods for data acquisition and processing

Given the complex nature of floating wind turbine systems, it is essential to continuously monitor their performance and anticipate potential failures. A typical floating wind turbine consists of several major components, including the hub, gearbox, bearings, spindle, and generator sets. Each of these components plays a vital role in ensuring the overall efficiency and safety of the turbine.

To monitor these components, wind turbine operators utilize a condition-monitoring system equipped with a variety of sensors. These sensors measure several critical physical parameters, such as rotational speed, power output, blade pitch angle, and nacelle (turbine housing) vibrations. The GL report Germanischer Lloyd (2010) recommends tracking these measurements to ensure the safety and longevity of wind turbines. The key measurement parameters used in this study are summarized in Table 5.

In this study, data were collected from wind turbine simulations that incorporated sensors to monitor the performance of various components. Each time series from these sensors exhibited unique behaviors, depending on the sensor's location and the specific fault simulated. Sudden changes in the time-series data, such as unexpected increases or decreases in parameters like rotor speed or nacelle displacement, were identified as anomalies that could signal potential faults in the system. These anomalies were the focus of the predictive models developed in this study.

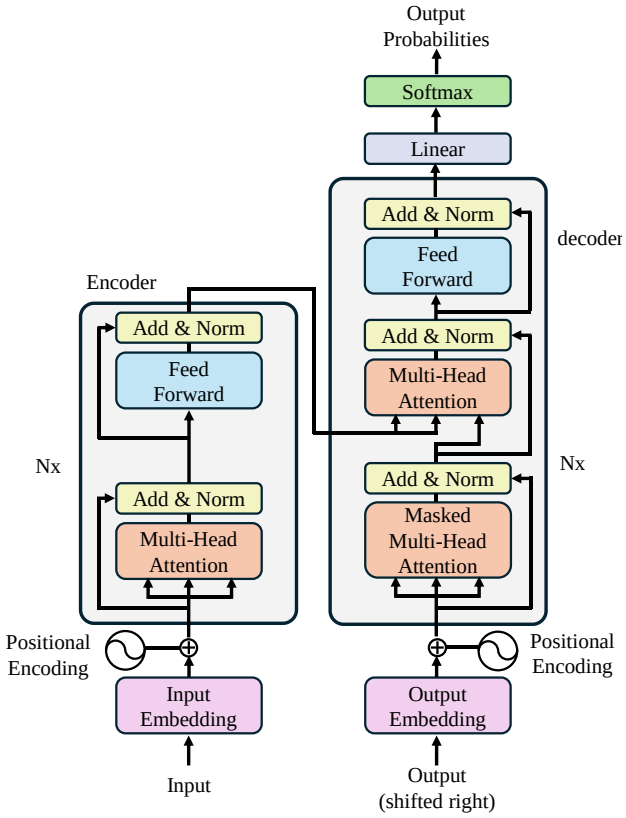


Fig. 4 Transformer model architecture (Vaswani *et al.* 2017)

By analyzing the collected data, the goal is to predict faults before they result in catastrophic failures, thus enabling predictive maintenance and improving the reliability and safety of offshore wind turbine systems.

4. Fault detection with transformer

4.1 Transformer

In the field of deep learning, transformers are model architectures designed to handle sequential data and revolutionize natural language processing (NLP). Unlike traditional RNNs or LSTMs, transformers do not require sequential data processing but instead use a self-attention mechanism (Vaswani *et al.* 2017) that independently weighs the influence of different parts of the input data, thus allowing the model to consider the entire input sequence simultaneously, effectively capturing long-range dependencies.

The self-attention mechanism is defined by the following equations

$$\mathbf{Q} = \mathbf{X}\mathbf{W}_\mathbf{Q} \tag{1a}$$

$$\mathbf{K} = \mathbf{X}\mathbf{W}_\mathbf{K} \tag{1b}$$

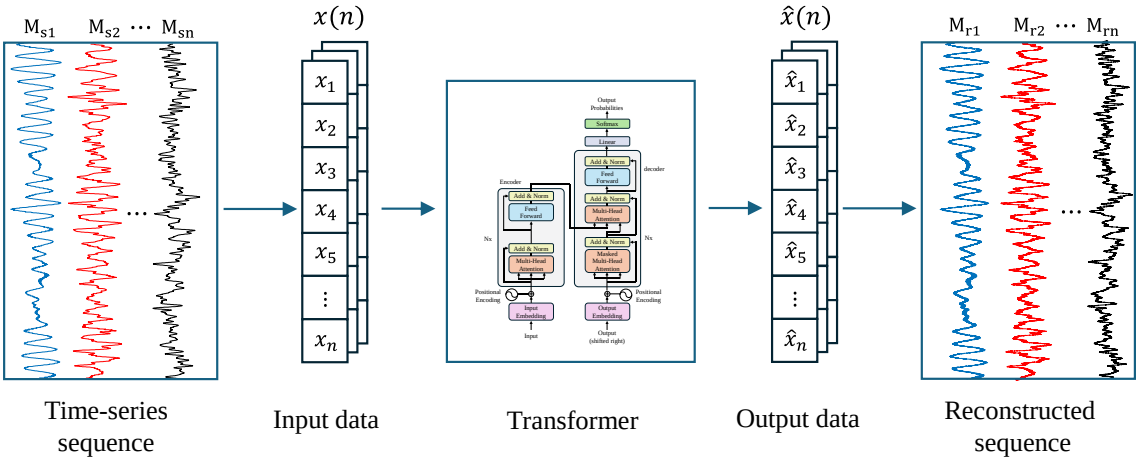


Fig. 5 Structure of the transformer-based fault detection scheme

$$\mathbf{V} = \mathbf{X}\mathbf{W_V} \tag{1c}$$

where **X** is the input sequence matrix, with matrices **Q** (queries), **K** (keys), and **V** (values) obtained by multiplying **X** by the learned weight matrices $\mathbf{W_Q}$, $\mathbf{W_K}$, and $\mathbf{W_V}$, respectively. The attention scores are calculated as follows

$$\text{Attention}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\mathbf{T}}}{\sqrt{\overrightarrow{d_k}}}\right)\mathbf{V} \tag{2}$$

where $\overrightarrow{d_k}$ denotes a key dimension vector. The transformer model stacks multiple layers of these self-attention mechanisms along with feedforward neural networks to transform the input sequence into an output sequence.

Transformers use an encoder-decoder structure wherein the encoder maps an input sequence to a continuous representation, and the decoder uses this representation to generate the output sequence. The encoder consists of a stack of identical layers, each with a multi-head self-attention mechanism and a position-wise, fully connected feedforward network. The decoder is similarly structured but includes an additional layer of multi-head attention that attends to the encoder's output. The multi-head attention mechanism is expressed as

$$\begin{aligned}&\text{MultiHead}(\mathbf{Q},\mathbf{K},\mathbf{V})\\&= \text{Concat}(\text{head}_1,\text{head}_2,\dots,\text{head}_h)\mathbf{W_O}\end{aligned} \tag{3a}$$

$$\text{head}_i = \text{Attention}(\mathbf{Q}\mathbf{W}_{\mathbf{Q}_i},\mathbf{K}\mathbf{W}_{\mathbf{K}_i},\mathbf{V}\mathbf{W}_{\mathbf{V}_i}) \tag{3b}$$

where $\mathbf{W_O}$ denotes the output weight matrix. After the attention heads are computed, their results are concatenated and then multiplied by $\mathbf{W_O}$ to produce the final output of the multi-head attention mechanism. This architecture allows transformers to model complex dependencies and interactions, making them highly effective in tasks (Fig. 4).

The fault-detection model employs the transformer method to learn from time-series data. The transformer was designed to learn the representations of the time-series data by replicating the system input as the output. The transformer-based encoder typically learns a compressed vector representation of the input time series, whereas the transformer decoder reconstructs the time-series data to closely match the input. During the learning process, the transformer-based algorithm learns to reconstruct instances of time-series data accurately. Fig. 5 illustrates the structure of the transformer-based fault detection scheme.

The transformer-based fault detection system learns from sensor data to generate the reconstruction loss, i.e., the difference between the input and reconstructed output data, thus identifying anomalies. A transformer model trained on sensor data from a fault-free system can effectively detect faults in multi-sensor time-series data. During training, if the transformer is exposed only to normal-state data and performs accurate reconstructions, the reconstruction loss is minimized under normal conditions. However, if abnormal data is input into the detection scheme, the reconstructed data will significantly diverge, leading to an increased reconstruction loss. Therefore, when a system experiences a fault, the reconstruction loss exceeds a predefined threshold, and this state is classified as abnormal. Fig. 6 shows the fault-detection system algorithm using the reconstruction loss.

4.2 Learning procedures

When measuring sensor data, physical quantities such as the acceleration of the nacelle, power, and rotational speed exhibit significant variations in terms of dimensions and magnitude. When these data are combined into a single training dataset, these differences can create difficulties during learning. Therefore, data normalization is essential for facilitating effective learning. The training dataset was normalized using the min-max normalization method based on the minimum and maximum values, as detailed in Eq. (4)

$$x_{i,normalized} = \frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)} \tag{4}$$

where x_i represents the input data of the i-th iteration, $x_{i,normalized}$ is the normalized data, and $\min(x_i)$ and $\max(x_i)$ are the minimum and maximum values of the data, respectively.

The model was trained solely with unlabeled samples

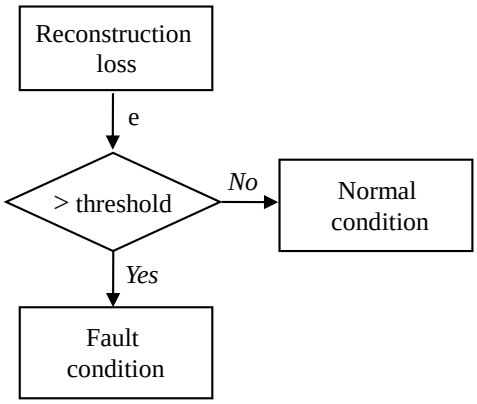


Fig. 6 Fault detection algorithm based on reconstruction loss

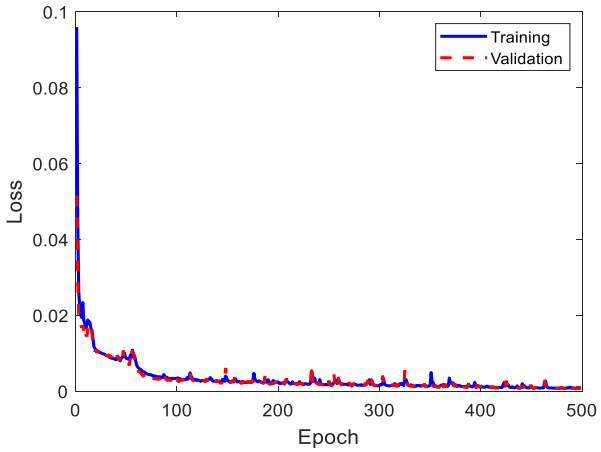


Fig. 7 Convergence of training and validation losses during the learning procedure

Table 6 Model hyperparameters of transformer architecture

Hyperparameter	Value
Batch size	128
Number of epochs	500
Embedding size	128
Number of layers	6
Number of attention heads	8
Optimizer	Adam
Learning rate	0.001
Loss function	MSE

using data measured during the steady-state condition of the system. The dataset was divided into three subsets—training, validation, and test—each containing approximately 108,000 values recorded over 3 h. Specifically, 70% of the initial portion of the dataset was used as training data, with the remaining 20% used as validation data and 10% for test data. The training and validation data, which contained only normal sequences, were used to train and validate the transformer model. The data for faults 1, 2, 3, and 4 were included in the test datasets, with the performance of the transformer model assessed through testing procedures for each fault.

During the training process, the transformer weights are updated with mini-batches, while Adam-based optimization techniques are employed to accelerate the training process (Kingma *et al.* 2014). The loss function used in the training procedure is generally the mean square error (MSE) between the network input and output. The network reconstructs the input data as accurately as possible to minimize the MSE. In the transformer model, the reconstruction loss was calculated to minimize the difference between the input and output. Eq. (5) represents the formula for calculating the reconstruction loss using the mean square error.

$$L(x_i - \hat{x_i}) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x_i})^2 \quad (5)$$

Here, $\hat{x}$ represents the output/predicted data, where the subscript i refers individual elements within a dataset and n denotes the number of samples in the training dataset.

The model was trained on a specified training dataset during the training process, and its performance was evaluated on a separate validation dataset. Fig. 7 shows the training and validation losses of the transformer model, with the blue line representing the training loss and the dotted red line representing the validation loss. Table 6 lists the model hyperparameters of the learning architecture used for the training and validation.

After training the model and setting the threshold for fault detection, the test dataset was fed into the trained model. The training data were also used to calculate the reconstruction loss. The distribution of the reconstruction loss for the training dataset, composed of data from the steady state of the system, is shown in Fig. 8.

Therefore, determining an appropriate alarm threshold is necessary to assess the operational status of wind turbines. In this study, the threshold was set using the kernel density estimation (KDE) method, which involved establishing limits based on a kernel density function. A confidence interval was used as the control limit for the normal operation of the blade pitch system. The real-time predicted threshold limits help determine the operational state of the system. An exceeded threshold limit indicates abnormal operation, prompting a fault warning. Fig. 8 illustrates the probability distribution of the residuals under normal conditions. Using domain knowledge and this distribution, the threshold for identifying data points as faults was set to 0.015.

4.3 Performance evaluation of transformer-based fault detection

We evaluated the effectiveness of the proposed fault-detection method using a test dataset. The results demonstrate the ability of the method to detect faults in data without explicit labels. The test dataset included four types of faults—Fault 1 (VEF), Fault 2 (VSL), Fault 3 (VWV), and Fault 4 (VSC)—divided into four subsets, each containing measurements from a floating wind turbine. These datasets were generated from numerical simulations using Simo-Riflex with faults.

Several performance metrics were used to evaluate the performance of the proposed transformer model. These

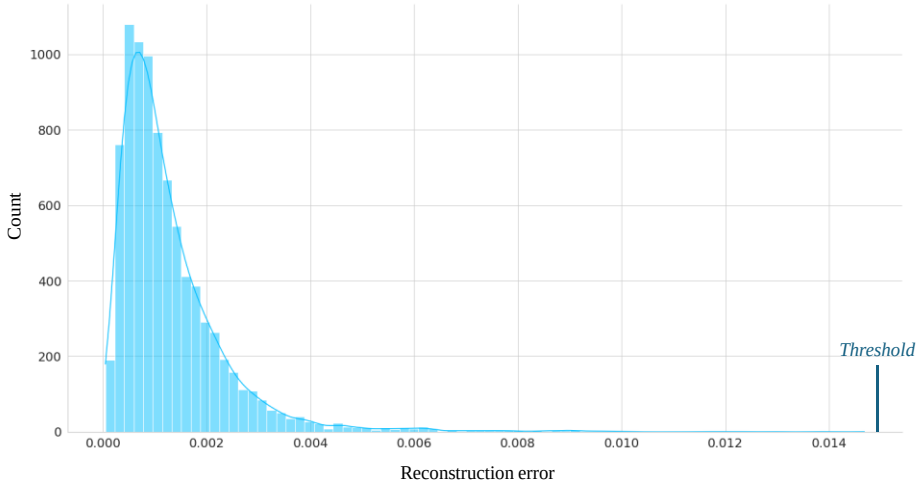


Fig. 8 Distribution of the reconstruction loss of the training data

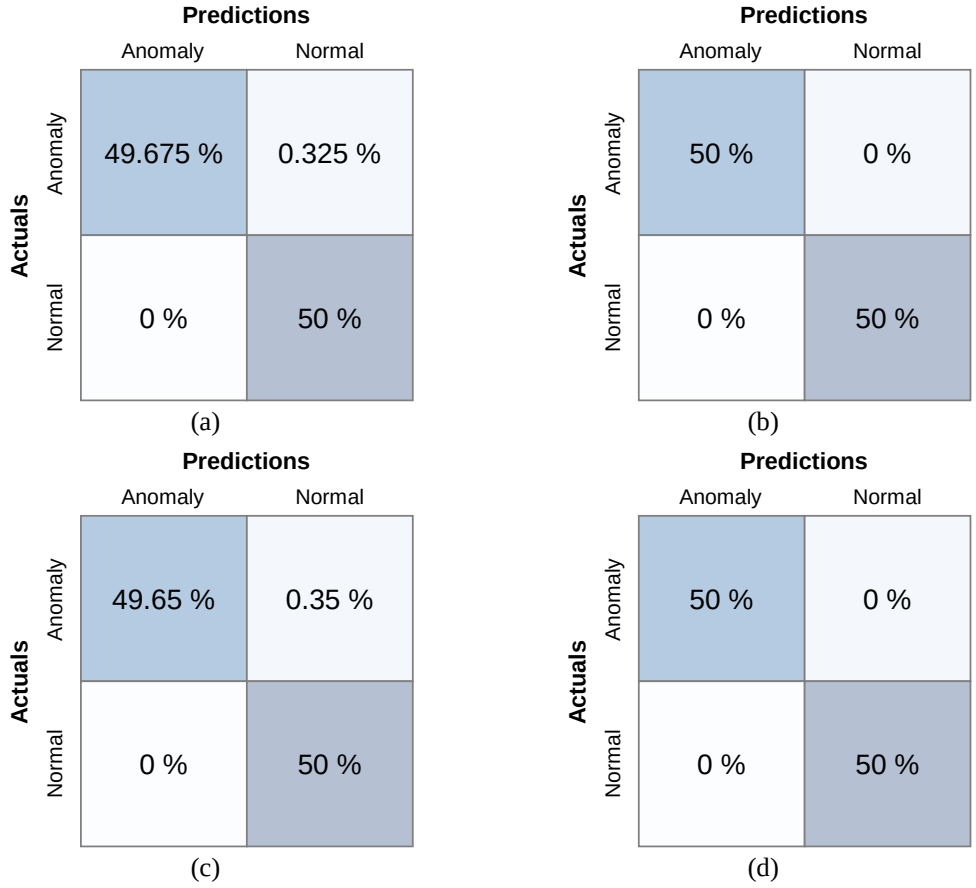


Fig. 9 Confusion matrix of the test data: (a) Fault 1, (b) Fault 2, (c) Fault 3, and (d) Fault 4

metrics include the accuracy, precision, recall, and F1 scores, as represented by the following formulas

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

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

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

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{9}$$

where true positive (TP) denotes the number of correctly identified fault events, true negative (TN) indicates the number of correctly identified normal events, false Positive (FP) represents the number of normal events incorrectly diagnosed as anomalies, false negative (FN) denotes the number of actual anomalies incorrectly identified as normal events.

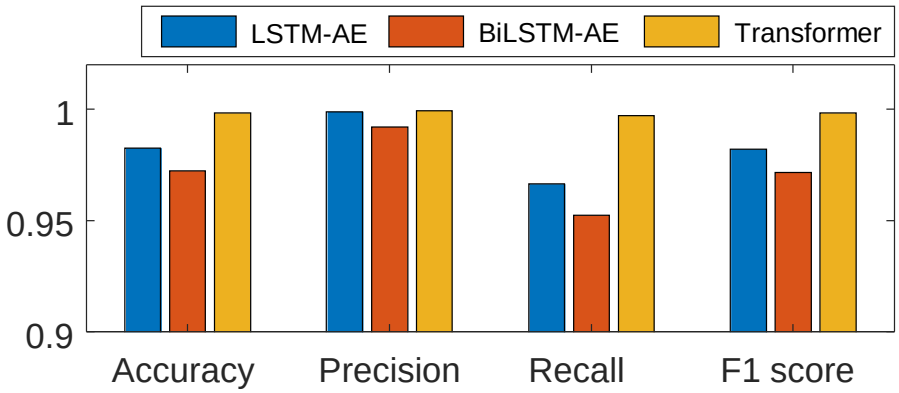


Fig. 10 Comparison of model performance metrics for fault detection

Table 7 Performance of transformer-based fault detection scheme

Fault number	Accuracy	Precision	Recall	F1 Score
1	0.997	1.000	0.994	0.99
2	1.000	1.000	1.000	1.000
3	0.996	1.000	0.993	0.996
4	1.000	1.000	1.000	1.000
Mean	0.998	1.000	0.997	0.998

The confusion matrix for the test dataset is shown in Fig. 9. The diagonal components, TP and TN, are significantly higher than the FP and FN values for each fault. These values influenced the overall accuracy, precision, recall, and F1 scores.

We calculated the performance metrics for the proposed model using Eqs. (6) and (9), respectively. The performance comparison results of the transformer and its metrics are summarized in Table 7. Fig. 10 presents a bar chart comparing the performance metrics, Accuracy, Precision, Recall, and F1 score of the three different fault-detection models: LSTM-AE, bidirectional LSTM-AE, and transformer. Each bar group represents the mean value of the metrics for Faults 1-4, with different colors indicating the three models. The proposed model based on the transformer consistently outperformed the other two models across all metrics, indicating its superior performance in fault detection tasks.

5. Conclusions

In this study, we developed a transformer-based fault detection system and validated its performance using datasets generated from wind turbine simulation data. The system effectively analyzes time-series data by utilizing a transformer architecture, which constructs an encoder and decoder trained on normal-state data. The transformer model reconstructs the sequence data within a defined space, enabling rapid fault detection in wind turbines using only normal operational data for training, with the fault detection threshold determined by the distribution of the reconstruction loss observed during training.

The results demonstrate that the proposed model is highly effective, achieving high accuracy in detecting four specific fault conditions in the blade pitch system of wind turbines. The model outperformed the LSTM-AE and Bi-LSTM-AE models across key performance metrics, showing superior ability in identifying faults and providing valuable system health information.

However, it is important to note that faults may occur in various components of wind turbines beyond just the blade pitch system, including the gearbox, generator, and nacelle, or even between these components. While our model focuses on detecting faults within the blade pitch system, the fault detection capabilities for other turbine components have not been comprehensively tested in this study. This limitation should be acknowledged, as the model's ability to detect faults across all turbine subsystems has yet to be fully explored.

Future work will involve expanding the fault detection capabilities to cover a wider range of turbine components, ensuring that potential faults in or between any critical subsystems can be accurately identified. Additionally, we aim to refine the model to further improve fault detection accuracy and operational efficiency.

In conclusion, this study contributes to the development of advanced fault detection methods for wind turbines, providing a foundation for models that can monitor turbine conditions with minimal or no human supervision.

Acknowledgments

We thank Professor Torgeir Moan and Zhen Gao at Norwegian University of Science and Technology (NTNU) for the valuable discussion and comments. This research was performed by the MIT-NTNU-Statoil Wind Turbine Program funded by Equinor (formerly Statoil). This research was also supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. 2022R1A6A1A03056784 and 2022R1C1C2006328).

References

Bach-Andersen, M., Rømer-Odgaard, B. and Winther, O. (2018), "Deep learning for automated drivetrain fault detection", *Wind Energy*, **21**(1), 29-41. https://doi.org/10.1002/we.2142.

Bakdi, A., Kouadri, A. and Mekhilef, S. (2019), "A data-driven

algorithm for online detection of component and system faults in modern wind turbines at different operating zones", *Renew. Sustain. Energy Rev.*, **103**, 546-555. https://doi.org/10.1016/j.rser.2019.01.013.

Carroll, J., McDonald, A. and McMillan, D. (2016), "Failure rate, repair time and unscheduled O&M cost analysis of offshore wind turbines", *Wind Energy*, **19**, 1107-1119. https://doi.org/10.1002/we.1887.

Chang, M., Kim, J.K. and Lee, J. (2019), "Hierarchical neural network for damage detection using modal parameters", *Struct. Eng. Mech.*, **70**(4), 457-466. https://doi.org/10.12989/sem.2019.70.4.457.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. and Bengio, Y. (2014), "Learning phrase representations using RNN encoder-decoder for statistical machine translation", arXiv preprint arXiv:1406.1078. https://doi.org/10.48550/arXiv.1406.1078.

Cho, S. (2024), "Anomaly detection in blade pitch systems of floating wind turbines using LSTM-Autoencoder", *J. Aerosp. Syst. Eng.*, **18**(4), 43-52. https://doi.org/10.20910/JASE.2024.18.4.43.

Cho, S., Bachynski, E., Nejad, A.R., Gao, Z. and Moan, T. (2020), "Numerical modeling of hydraulic blade pitch actuator in a spar-type floating wind turbine considering fault conditions and their effects on global dynamic responses", *Wind Energy*, **23**(2), 370-390. https://doi.org/10.1002/we.2438.

Cho, S., Choi, M., Gao, Z. and Moan, T. (2021), "Fault detection and diagnosis of a blade pitch system in a floating wind turbine based on Kalman filters and artificial neural networks", *Renew. Energy*, **169**, 1-13. https://doi.org/10.1016/j.renene.2020.12.116.

Cho, S., Gao, Z. and Moan, T. (2018), "Model-based fault detection, fault isolation and fault tolerant control of a blade pitch system in floating wind turbines", *Renew. Energy*, **120**, 306-321. https://doi.org/10.1016/j.renene.2017.12.102.

Cho, S., Park, J. and Choi, M. (2021), "Fault classification of a blade pitch system in a floating wind turbine based on a recurrent neural network", *J. Ocean Eng. Technol.*, **35**(4), 287-295. https://doi.org/10.26748/KSOE.2021.018.

Choi, J.H. (2014), "A review on prognostics and health management and its applications", *J. Aerosp. Syst. Eng.*, **8**(4), 7-17. https://doi.org/10.20910/JASE.2014.8.4.007.

Cui, Y., Bangalore, P. and Tjernberg, L.B. (2021), "A fault detection framework using recurrent neural networks for condition monitoring of wind turbines", *Wind Energy*, **24**(11), 1249-1262. https://doi.org/10.1002/we.2628.

de Andrade Melani, A.H., de Carvalho Michalski, M.A., da Silva, R.F. and de Souza, G.F.M. (2021), "A framework to automate fault detection and diagnosis based on moving window principal component analysis and Bayesian network", *Reliab. Eng. Syst. Saf.*, **215**, 107837. https://doi.org/10.1016/j.ress.2021.107837.

Dervilis, N., Choi, M., Taylor, S.G., Barthorpe, R.J., Park, G., Farrar, C.R. and Worden, K. (2014), "On damage diagnosis for a wind turbine blade using pattern recognition", *J. Sound Vib.*, **333**(6), 1833-1850. https://doi.org/10.1016/j.jsv.2013.11.015.

Dhibi, K., Mansouri, M., Bouzrara, K., Nounou, H. and Nounou, M. (2022), "Enhanced recurrent neural network for fault diagnosis of uncertain wind energy conversion systems", *Proceedings of International Conference on Control, Decision and Information Technologies (CoDIT)*, **1**, 1330-1335. https://doi.org/10.1109/CoDIT55151.2022.9804119.

Dinwoodie, I., McMillan, D., Revie, M., Lazakis, I. and Dalgic, Y. (2013), "Development of a combined operational and strategic decision support model for offshore wind", *Energy Procedia*, **35**, 157-166. https://doi.org/10.1016/j.egypro.2013.07.169 .

Finotti, R.P., Gentile, C., Barbosa, F. and Cury, A. (2022), "Structural novelty detection based on sparse autoencoders and control charts", *Struct. Eng. Mech.*, **81**(5), 647-664. https://doi.org/10.12989/sem.2022.81.5.647.

Gayo, J.B. (2011), "Reliability-focused research on optimizing wind energy system design, operation and maintenance: Tools, proof of concepts, guidelines & methodologies for a new generation", Final Publishable Summary of Results of Project ReliaWind.

Hameed, Z., Hong, Y.S., Cho, Y.M., Ahn, S.H. and Song, C.K. (2009), "Condition monitoring and fault detection of wind turbines and related algorithms: A review", *Renew. Sustain. Energy Rev.*, **13**, 1-39. https://doi.org/10.1016/j.rser.2007.05.008.

Harrou, F., Kini, K.R., Madakyaru, M. and Sun, Y. (2023), "Uncovering sensor faults in wind turbines: An improved multivariate statistical approach for condition monitoring using SCADA data", *Sustain. Energy Grids Netw.*, **35**, 101126. https://doi.org/10.1016/j.segan.2023.101126.

Hasselmann, K., Barnett, T.P., Bouws, E., Carlson, H., Cartwright, D.E., Enke, K., ... & Walden, H. (1973), "Measurements of wind-wave growth and swell decay during the Joint North Sea Wave Project (JONSWAP)", Ergaenzungsheft zur Deutschen Hydrographischen Zeitschrift, Reihe A.

Hochreiter, S. and Schmidhuber, J. (1997), "Long short-term memory", *Neur. Comput.*, **9**(8), 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735.

International Electrotechnical Commission (2005), IEC 61400-1: Wind Turbines-Part 1: Design Requirements.

International Electrotechnical Commission (2005), IEC 61400-3: Wind Turbines-Part 3: Design Requirements for Offshore Wind Turbines.

Isermann, R. (2006), *Fault-diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*, Springer Science & Business Media.

Jiang, G., Xie, P., He, H. and Yan, J. (2017), "Wind turbine fault detection using a denoising autoencoder with temporal information", *IEEE/ASME Trans. Mechatron.*, **23**(1), 89-100. https://doi.org/10.1109/TMECH.2017.2759301.

Jiang, G.Q., He, H.B., Yan, J. and Xie, P. (2018), "Multiscale convolutional neural networks for fault diagnosis of wind turbine gearbox", *IEEE Trans. Ind. Electron.*, **66**, 3196-3207. https://doi.org/10.1109/TIE.2018.2844805.

Johannessen, K., Meling, T. and Haver, S. (2001), "Joint distribution for wind and waves in the northern North Sea", *Int. J. Offshore Polar Eng.*, **12**(1), 1-8.

Jonkman, J. (2010), "Definition of the Floating System for Phase IV of OC3", Technical Report, NREL/TP-500-47535, USA.

Jonkman, J. and Kilcher, L. (2012), TurbSim User's Guide, Technical Report, NREL, USA.

Jonkman, J., Butterfield, S., Musial, W. and Scott, G. (2009), "Definition of a 5-MW reference wind turbine for offshore system development", Technical Report, NREL/TP-500-38060, USA.

Kingma, D.P. and Jimmy, B. (2014), "Adam: A method for stochastic optimization", arXiv, 1412.6980v9. https://doi.org/10.48550/arXiv.1412.6980/

Kusiak, A. and Verma, A. (2011), "A data-driven approach for monitoring blade pitch faults in wind turbines", *IEEE Trans. Sustain. Energy*, **2**, 87-96. https://doi.org/10.1109/TSTE.2010.2066585.

Kusiak, A. and Verma, A. (2012), "Analyzing bearing faults in wind turbines: A data-mining approach", *Renew. Energy*, **48**, 110-116. https://doi.org/10.1016/j.renene.2012.04.020.

Lei, J., Liu, C. and Jiang, D. (2019), "Fault diagnosis of wind turbine based on Long Short-term memory networks", *Renew. Energy*, **133**, 422-432. https://doi.org/10.1016/j.renene.2018.10.031.

Liu, J., Yang, G., Li, X., Wang, Q., He, Y. and Yang, X. (2023), "Wind turbine anomaly detection based on SCADA: A deep autoencoder enhanced by fault instances", *ISA Trans.*, **139**, 586-605. https://doi.org/10.1016/j.isatra.2023.03.045.

Germanischer Lloyd (2010), Guideline for the Certification of

Wind Turbines, GL Report.
Lu, B., Li, Y., Wu, X. and Yang, Z. (2009), "A review of recent advances in wind turbine condition monitoring and fault diagnosis", *2009 IEEE Power Electronics and Machines in Wind Applications*, 1-7.
Malhotra, P., Vig, L., Shroff, G. and Agarwal, P. (2015), "Long short term memory networks for anomaly detection in time series", *Proceedings of European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*, 89-94.
Márquez, F.P.G., Tobias, A.M., Pérez, J.M.P. and Papaelias, M. (2012), "Condition monitoring of wind turbines: Techniques and methods", *Renew. Energy*, **46**, 169-178. https://doi.org/10.1016/j.renene.2012.03.003.
Nguyen, H.D., Bui Tien, T., De Roeck, G. and Abdel Wahab, M. (2021), "Damage detection in structures using modal curvatures gapped smoothing method and deep learning", *Struct. Eng. Mech.*, **77**(1), 47-56. https://doi.org/10.12989/sem.2021.77.1.047.
Nguyen, H.D., Tran, K.P., Thomassey, S. and Hamad, M. (2021), "Forecasting and anomaly detection approaches using LSTM and LSTM autoencoder techniques with the applications in supply chain management", *Int. J. Inf. Manage.*, **57**, 102282. https://doi.org/10.1016/j.ijinfomgt.2020.102282.
Rahimilarki, R., Gao, Z., Jin, N. and Zhang, A. (2022), "Convolutional neural network fault classification based on time-series analysis for benchmark wind turbine machine", *Renew. Energy*, **185**, 916-931. https://doi.org/10.1016/j.renene.2021.12.056.
Schlag, I., Irie, K. and Schmidhuber, J. (2021), "Linear transformers are secretly fast weight programmers", *Proceedings of International Conference on Machine Learning*, PMLR, 9355-9366.
Shin, S., Hyun, C., Cho, S. and Lee, P.S. (2023), "Multi-sensor data-based anomaly detection and diagnosis of a pumped storage hydropower plant", *Struct. Eng. Mech.*, **88**(6), 569-581. https://doi.org/10.12989/sem.2023.88.6.569.
SINTEF Ocean (2018), RIFLEX 4.15.0 User Guide.
SINTEF Ocean (2018), SIMO 4.15.0 User Guide.
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., ... and Polosukhin, I. (2017), "Attention is all you need", *Advances in Neural Information Processing Systems*, https://doi.org/10.48550/arXiv.1706.03762.
Wang, L., Zhang, Z., Long, H. and Xu, J. (2016), "Wind turbine gearbox failure identification with deep neural networks", *IEEE Trans. Ind. Inform.*, **13**(3), 1360-1368. https://doi.org/10.1109/TII.2016.2607179.
Wu, H., Triebe, M.J. and Sutherland, J.W. (2023), "A transformer-based approach for novel fault detection and fault classification/diagnosis in manufacturing: A rotary system application", *J. Manuf. Syst.*, **67**, 439-452. https://doi.org/10.1016/j.jmsy.2023.02.018.
Zaher, A.S., McArthur, S.D., Infield, D.G. and Patel, Y. (2009), "Online wind turbine fault detection through automated SCADA data analysis", *Wind Energy*, **12**(6), 574-593. https://doi.org/10.1002/we.319.
Zhou, K.L., Fu, C. and Yang, S.L. (2016), "Big data driven smart energy management: From big data to big insights", *Renew. Sustain. Energy Rev.*, **56**, 215-225. https://doi.org/10.1016/j.rser.2015.11.050.