An Internet of Things-based intelligent monitoring and fault detection method for the operation and maintenance of offshore wind turbines

Yilei Ouyang, Wei Liang
China University of Petroleum(Beijing), Beijing 102249, China
Corresponding author: Wei Liang (e-mail: 15624337910@163.com)

ABSTRACT The rapid expansion of offshore wind farms has introduced significant challenges to operation and maintenance (O&M), particularly under harsh marine environments where reliable electromagnetic information transmission and constrained wireless communication resources directly affect intelligent monitoring performance. Traditional Supervisory Control and Data Acquisition (SCADA) systems relying on cloud-centric architectures often encounter excessive latency and bandwidth bottlenecks when transmitting high-frequency vibration signals, limiting real-time fault diagnosis. To address these issues, this study proposes an Internet of Things (IoT)-based intelligent monitoring and fault detection method built upon an Edge-Cloud collaborative architecture. A lightweight Adaptive One-Dimensional Convolutional Neural Network (A-1D-CNN) is developed for deployment on edge gateway devices, enabling direct extraction of fault characteristics from raw vibration signals without manual feature engineering. Combined with an " Edge-Training, Cloud-Update" strategy, the proposed framework continuously optimizes diagnostic performance while substantially reducing communication overhead across wireless sensing and electromagnetic transmission infrastructures. Experimental evaluation on a standard bearing fault dataset demonstrates that the proposed method achieves a fault diagnosis accuracy of 99.25% with a compact model size of only 0.45 MB, providing an effective balance between diagnostic precision and deployment efficiency. The results indicate that the proposed framework offers a practical solution for real-time intelligent monitoring in bandwidth-limited offshore environments and provides technical support for reliable electromagnetic-enabled sensing networks and distributed fault diagnosis in next-generation offshore energy systems.

INDEX TERMS offshore wind turbine, edge-cloud collaboration, fault detection, electromagnetic information transmission, wireless sensing network

I. INTRODUCTION

A. RESEARCH BACKGROUND

AS the global demand for clean energy intensifies, offshore wind energy has emerged as a pivotal component of the renewable energy landscape due to its higher wind speeds, lower turbulence, and lack of land occupation compared to onshore counterparts [1-3]. To withstand high humidity and salt spray corrosion, many offshore wind turbine (OWT) components, such as blades and nacelle covers, are increasingly made of high-strength reinforced composites. However, offshore wind turbines operate in a harsh environment [4, 5]. High humidity, salt spray corrosion, and unpredictable wind loads all accelerate performance degradation. As a result, O&M costs remain high, accounting for approximately 25% to 30% of the levelized cost of energy [6, 7]. Ensuring the reliability of critical components—such as the gearbox, generator, and bearings—is paramount, as unexpected downtime can lead to significant economic losses and grid instability [8].

B. MOTIVATION AND PROBLEM STATEMENT

Traditionally, Condition Monitoring Systems (CMS) and SCADA systems have been employed to monitor OWTs [9]. These systems typically adopt a centralized cloud-computing architecture [10]. Sensors collect large volumes of data and transmit them to a remote control center for analysis. While effective for onshore farms, this architecture faces severe bottlenecks in offshore scenarios:

1. Bandwidth Constraints and Latency: High-fidelity fault detection requires high-frequency sampling (e.g., vibration signals sampled at 20 kHz or higher) [11]. Transmitting raw time-series data from offshore turbines to onshore

data centers is expensive and technically challenging [12, 13]. This issue becomes more severe when high-frequency vibration data are continuously uploaded through satellite or constrained wireless links. As a result, packet loss and transmission latency may increase.

2. Weak Fault Features: Early-stage mechanical faults in OWTs often manifest as weak impulses submerged in strong background noise caused by wind and wave interactions [10, 14]. Traditional threshold-based alarms in SCADA systems are prone to high False Alarm Rates (FAR) because they fail to extract deep features from these noisy signals [15, 16].

3. Inefficiency of Traditional Algorithms: Conventional diagnostic methods, such as Support Vector Machines (SVM) or Artificial Neural Networks (ANN), rely heavily on manual feature extraction (e.g., calculating RMS, Kurtosis, or FFT) [17]. This process is labor-intensive, requires domain expertise, and often fails to generalize across different operating conditions.

Therefore, there is an urgent need to shift from a centralized "Cloud-only" paradigm to an "Edge-Cloud Collaborative" model. By processing data locally at the turbine (the "Edge"), we can extract critical information immediately, reduce network load, and achieve real-time fault detection.

C. CONTRIBUTIONS

To address the aforementioned challenges, this paper proposes a novel IoT-based intelligent monitoring framework that integrates Edge Computing with Deep Learning. The main contributions of this study are summarized as follows:

- Development of an Edge-Cloud Collaborative IoT Architecture: We propose a hierarchical architecture that distributes computing tasks between the edge (turbine gateway) and the cloud. The edge layer performs real-time data cleaning and preliminary fault diagnosis, while the cloud layer handles long-term trend analysis and model retraining. This architecture significantly alleviates bandwidth pressure.
- Proposal of a Lightweight Adaptive 1D-CNN Model: We design a lightweight Convolutional Neural Network specifically optimized for edge devices with limited computing power. The model utilizes 1D convolution kernels to process raw time-series data directly, eliminating the need for complex manual feature engineering. Global Average Pooling (GAP) is introduced to minimize the number of parameters, ensuring fast inference speeds.
- Comprehensive Experimental Validation: The proposed method is validated using a standard bearing fault dataset under various noise levels and load conditions, in order to provide a controlled benchmark for assessing diagnostic accuracy and deployment efficiency. Comparative experiments demonstrate that our method achieves strong overall performance against representative traditional machine learning and deep learning baselines in terms of accuracy, robustness, and deployment efficiency. Since these baselines use different input representations, the comparison should be interpreted as a pipeline-level benchmark rather than a strictly controlled like-for-like comparison.

II. RELATED WORK

A. IOT TECHNOLOGIES IN WIND ENERGY MONITORING

The integration of the Internet of Things (IoT) into the wind energy sector has revolutionized traditional Operations and Maintenance (O&M) strategies [18]. Early monitoring systems primarily relied on SCADA (Supervisory Control and Data Acquisition) data, which typically records low-frequency variables such as temperature, wind speed, and active power at 10-minute intervals [19]. While useful for performance analysis, SCADA data lacks the temporal resolution required to detect incipient mechanical faults in rotating components like bearings and gearboxes [20, 21]. To overcome this, recent studies have introduced wireless sensor networks (WSN) using protocols like ZigBee, LoRa, and Wi-Fi for vibration monitoring [12, 22, 23]. For instance, several researchers have proposed vibration-based energy harvesting sensors to achieve autonomous monitoring [24]. However, a critical limitation remains in the context of offshore wind farms: the bottleneck of data transmission [25]. The bandwidth provided by satellite or long-distance wireless links is often insufficient to handle the continuous stream of high-frequency raw vibration data (often exceeding several GBs per day per turbine), necessitating a paradigm shift towards edge computing [26].

B. FAULT DIAGNOSIS ALGORITHMS: FROM MACHINE LEARNING TO DEEP LEARNING

Fault diagnosis methods can be broadly categorized into signal processing-based, machine learning-based, and deep learning-based approaches.

- Traditional Signal Processing: Techniques such as Fast Fourier Transform (FFT), Wavelet Transform (WT), and Envelope Analysis are widely used to extract fault characteristic frequencies [27]. However, these methods require significant domain expertise to select appropriate base functions and are sensitive to the non-stationary noise typical of offshore environments [28].
- Shallow Machine Learning: Algorithms like Support Vector Machines (SVM), Random Forest (RF), and k- Nearest Neighbors (k-NN) have been extensively applied [29]. These methods typically rely on manual feature extraction (e.g., calculating RMS, Kurtosis, Skewness) [30]. The performance of these models is heavily dependent on the quality of the engineered features, which may not generalize well across different turbine models or operating conditions [31].
- Deep Learning (DL): In recent years, Deep Learning has achieved state-of-the-art results by automatically learning feature representations [32]. 2D Convolutional Neural Networks (2D-CNN) are popular for analyzing time-frequency images (e.g., spectrograms) of vibration

signals [33]. Despite their high accuracy, standard DL models (like VGG-16 or ResNet) possess millions of parameters and require substantial computational resources (GPU), making them unsuitable for deployment on resource-constrained IoT edge devices in offshore turbines [34, 35].

C. RESEARCH GAP

Current research typically focuses either on high-accuracy heavy models for cloud deployment or simple statistical metrics for edge deployment [34, 36]. There is a lack of lightweight, end-to-end deep learning models that can run efficiently on edge gateways to perform high-fidelity fault detection without relying on massive data transmission [26]. This paper addresses this gap by proposing a lightweight Adaptive 1D-CNN.

III. METHODOLOGY

A. OVERALL EDGE-CLOUD COLLABORATIVE ARCHITECTURE

The overall framework of the proposed IoT-based monitoring system is illustrated in Figure 1. It consists of three hierarchical layers: the Perception Layer, the Edge Computing Layer, and the Cloud Service Layer. This structure is designed to minimize data transmission latency.

- Perception Layer: This layer consists of high-frequency piezoelectric vibration sensors (accelerometers) installed on the drive train (main bearing, gearbox, generator). The sensors collect analog vibration signals which are converted to digital signals via an A/D converter.
- Edge Computing Layer (The Core): A localized edge gateway (e.g., an industrial embedded computer or high-performance ARM-based controller) is installed in the turbine nacelle. This layer executes two critical tasks:

1. Data Preprocessing: Cleaning and normalizing raw data.
2. Real-time Inference: Running the lightweight 1D-CNN model to classify the turbine's health status locally. Only the diagnostic results and a small segment of abnormal data are transmitted to the cloud. Under the simplified payload-level transmission model adopted in this work, this strategy can substantially reduce the amount of transmitted application data compared with continuous raw-data upload.

- Cloud Service Layer: The cloud platform receives the diagnostic reports. It is responsible for visualizing fleet status, long-term data storage, and performing computationally intensive tasks such as model retraining (Transfer Learning) when new fault patterns are discovered.

B. DATA ACQUISITION AND PREPROCESSING

The vibration signals collected from OWTs are non-stationary time-series data. Let the raw vibration signal be denoted as $X = \{x_1, x_2, \ldots, x_N\}$, where N is the number of sampling points.

1) Data Segmentation

To facilitate model input, the continuous vibration signal X is segmented using a sliding window technique. Let $L = 1024$ denote the window length and $S = 512$ denote the step size. The m-th segmented sample, denoted by $s_m \in \mathbb{R}^L$, is defined as follows:

$$s_m = \{x_{(m-1)S+1}, x_{(m-1)S+2}, \ldots, x_{(m-1)S+L}\}. \quad (1)$$

This 50% overlap strategy ($S = L/2$) helps preserve fault patterns occurring near segment boundaries. Consequently, the dataset consists of M distinct samples $\{s_1, s_2, \ldots, s_M\}$.

2) Z-Score Normalization

To mitigate the impact of varying load conditions (which affect vibration amplitude), we apply Z-score normalization to standardize the data:

$$x'_i = \frac{x_i - \mu}{\sigma}. \quad (2)$$

where x'_i denotes the normalized sample, μ is the mean of the segment, and σ is the standard deviation. This ensures that the data distribution follows $N(0, 1)$, accelerating the convergence of the neural network.

C. PROPOSED LIGHTWEIGHT ADAPTIVE 1D-CNN MODEL

Unlike 2D-CNNs that require image transformation (increasing computational cost), the proposed 1D-CNN processes raw time-series data directly. The model architecture is designed to be "lightweight" by minimizing the number of parameters while maximizing feature extraction capability.

1) Adaptive Multi-Scale Feature Extraction

Traditional CNNs typically employ a fixed kernel size (e.g., 3×1), which may be insufficient to capture fault-related patterns occurring at different temporal scales. To address this issue, we employ a multi-branch convolution block with kernel sizes of 3, 5, and 7. This configuration is intended to cover short-, medium-, and relatively long-range local patterns in the vibration signal. We do not claim that this kernel combination is theoretically optimal; rather, it is an empirically motivated design choice, whose effectiveness is further supported by the ablation study.

The outputs of these branches are concatenated to form a comprehensive feature map. Mathematically, the adaptive feature extraction is defined as:

$$Y_{\text{adaptive}} = \text{Concat}\left(f_{k=3}(X), f_{k=5}(X), f_{k=7}(X)\right). \quad (3)$$

Where $f_k(\cdot)$ represents the 1D convolution operation with kernel size k, followed by a Rectified Linear Unit (ReLU) activation function: $f(x) = \max(0, x)$. This mechanism

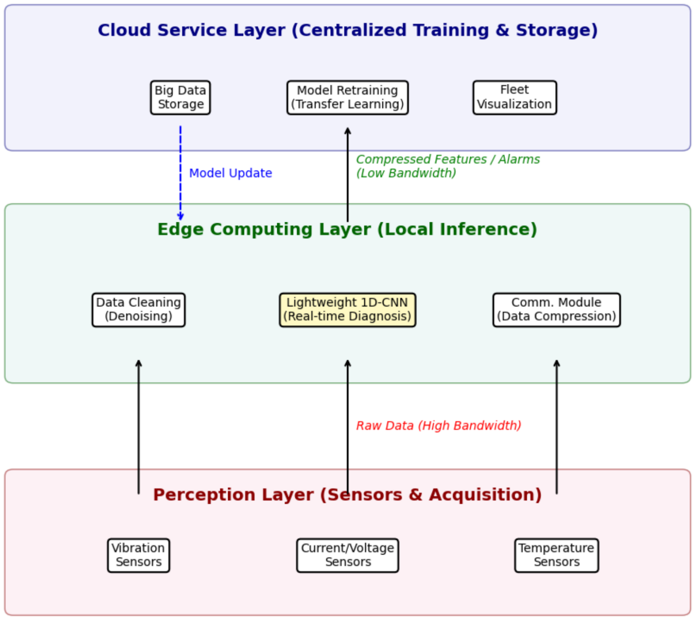

FIGURE 1. The proposed IoT-based Edge-Cloud collaborative architecture for offshore wind turbine monitoring.

allows the model to adaptively focus on the most relevant frequency bands for different fault types, significantly enhancing feature representation capability.

2) Max-Pooling Layer

To reduce the dimensionality of the feature maps and improve robustness to noise, Max-Pooling is applied:

$$p_j^{(l)}(k) = \max_{(k-1)W+1 \le t \le kW} y_j^{(l)}(t). \tag{4}$$

Where W is the pooling window size. This operation retains the most significant features while discarding weak noise signals.

3) Global Average Pooling (GAP) - The Lightweight Key

In traditional CNNs, the convolutional layers are followed by fully connected (FC) layers, which contain the majority of the model's parameters (often $> 80\%$). To make our model suitable for edge devices, we replace the traditional FC layers with a Global Average Pooling (GAP) layer. GAP calculates the average value of each feature map directly:

$$\mathrm{GAP}_j = \frac{1}{H}\sum_{h=1}^{H} y_j^{(l)}(h). \tag{5}$$

Where H is the length of the feature map. This drastically reduces the parameter count and prevents overfitting.

4) Softmax Classification

The final output vector is passed through a Softmax layer to produce a probability distribution over the fault classes (e.g., Normal, Ball Fault, Inner Race Fault, Outer Race Fault):

$$P(c) = \frac{e^{z_c}}{\sum_{k=1}^{K} e^{z_k}}. \tag{6}$$

5) Network Architecture Summary

The detailed structural parameters of the proposed Adaptive 1D-CNN are listed in Table 1. To ensure the model remains lightweight (approx. 0.45 MB), we minimized the number of filters in deeper layers and utilized Global Average Pooling to eliminate heavy fully connected layers.

D. EDGE-CLOUD COLLABORATION STRATEGY

The workflow of the proposed method is defined as follows:

1. Offline Training (Cloud): The 1D-CNN model is initially trained on the cloud server using historical labeled data.

2. Model Deployment (Edge): The trained model weights are compressed and transmitted to the edge gateway.

3. Online Monitoring (Edge): The edge gateway performs real-time inference. If the confidence score $P(c)$ is below a threshold θ (indicating an unknown pattern), the raw data is uploaded to the cloud for expert labeling.

4. Model Update (Cloud to Edge): The cloud retrains the model with the newly labeled data and pushes the updated weights back to the edge, creating a closed-loop self-learning system.

IV. EXPERIMENTAL SETUP

A. DATASET DESCRIPTION

To strictly prevent data leakage caused by the overlapping sliding window, we avoided simple random splitting. In-

TABLE 1. Detailed Architecture of the Proposed Adaptive 1D-CNN.

Layer Type	Kernel Size / Stride	Output Shape (Channel $\times$ Length)	Parameters
Input	–	1×1024	0
Adaptive Conv Block	$k = \{3, 5, 7\}$	$(4+6+6) \times 1024$	320
Max-Pooling 1	$2/2$	16×512	0
Conv Layer 2	$3/1$	32×512	1,568
Max-Pooling 2	$2/2$	32×256	0
Conv Layer 3	$3/1$	64×256	6,208
Max-Pooling 3	$2/2$	64×128	0
Global Avg Pooling	–	64×1	0
Softmax (Output)	–	4 (Classes)	260
Total	–	–	~ 0.45 MB

Note: The parameter count calculation is based on 32-bit floating-point storage.

stead, we adopted a time-series block splitting strategy. The first 70% of the sequential time-series data was assigned to the training set, the subsequent 15% to the validation set, and the final 15% to the testing set. This ensures that the testing samples are strictly "unseen" and temporally separated from the training samples.

B. EXPERIMENTAL PLATFORM

The experiments were conducted in two phases: Model Training (Cloud Simulation) and Model Deployment (Edge Simulation).

- Cloud Server (Training): The model was trained on a workstation equipped with an Intel Core i9-12900K CPU, 64GB RAM, and an NVIDIA GeForce RTX 3090 GPU. The deep learning framework used was PyTorch 1.13.
- Edge Gateway (Inference): To simulate the limited computational resources of an offshore wind turbine gateway, the trained model was deployed on a Raspberry Pi 4B (4GB RAM, ARM Cortex-A72). This setup was used to evaluate deployment-related indicators, including inference time, model size, and peak memory usage, thereby providing a more complete assessment of the 'lightweight' property.

C. TRAINING IMPLEMENTATION DETAILS

To ensure reproducibility, the detailed hyper-parameter settings are configured as follows: The model is trained using the Cross-Entropy Loss function to measure classification error. The Adam optimizer is employed with an initial learning rate of 0.001, which decays by a factor of 0.9 every 10 epochs. The batch size is set to 64, and the training process spans 100 epochs with an early stopping mechanism (patience = 10) to prevent overfitting. No additional data augmentation was applied during the training phase beyond the sliding window overlap.

D. EVALUATION METRICS

- To comprehensively evaluate the performance, we employed both diagnostic metrics and deployment-related efficiency metrics, including Accuracy, F1-Score, Inference Time, Model Size, and Peak Memory Usage. Accuracy: The ratio of correctly classified samples to total samples.
- F1-Score: The harmonic mean of precision and recall, crucial for assessing performance on imbalanced datasets.

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{7}$$

- Inference Time: The average time required to process one sample on the edge device (measured in milliseconds, ms).
- Model Size: The storage space occupied by the model parameters (measured in Megabytes, MB).
- Peak Memory Usage: The maximum RAM consumption during model inference on the edge device (measured in Megabytes, MB).

V. RESULTS AND DISCUSSION

A. DIAGNOSTIC PERFORMANCE COMPARISON

We compared the proposed Adaptive 1D-CNN with three state-of-the-art baseline methods: (1) SVM (with handcrafted features: RMS, Kurtosis, Crest Factor); (2) Standard 2D-CNN (using Short-Time Fourier Transform images as input); and (3) LSTM (Long Short-Term Memory network).

As shown in Table 2, the proposed method achieves the highest accuracy of 99.25%. Among the compared methods, the proposed 1D-CNN achieves the best overall diagnostic performance. The SVM baseline shows the lowest accuracy, which may be related to the limited expressive power of manually engineered features under varying fault severities. The 2D-CNN also performs well, but it relies on time-frequency image conversion, which introduces additional pre-processing overhead. In addition, because the compared methods use different input representations (manual features, time-frequency images, or raw sequences), the results in Table 2 should be interpreted as a representative pipeline-level comparison rather than a strictly controlled architectural comparison. Under the raw-sequence setting, the proposed 1D-CNN also outperforms LSTM, indicating that convolution-based local feature extraction is effective for this task.

B. ROBUSTNESS ANALYSIS UNDER SIMULATED NOISE CONDITIONS

To further evaluate the reliability of the proposed model in harsh marine environments, we introduced additive white

TABLE 2. Performance Comparison of Different Methods

Method	Input Type	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
SVM (RBF Kernel)	Manual Features	89.45	88.20	89.10	88.65
LSTM	Raw Sequence	94.12	93.80	94.05	93.92
Standard 2D-CNN	Time-Frequency Image	97.80	97.50	97.65	97.57
Proposed 1D-CNN	Raw Sequence	99.25	99.10	99.15	99.12

Gaussian noise (AWGN) to the test set. Figure 2 compares the diagnosis accuracy of different methods under varying Signal-to-Noise Ratios (SNR) ranging from -4 dB to 10 dB. As observed, the proposed Adaptive 1D-CNN demonstrates superior robustness compared to the baseline methods.

- When SNR = 10 dB, all deep learning models perform well (>95%).
- Crucially, when SNR drops to -4 dB (representing a high-interference scenario where weak fault signatures are heavily masked by background noise):
- The accuracy of SVM drops drastically to 62%.
- The Standard 2D-CNN drops to 84%.
- The Proposed 1D-CNN maintains an accuracy of 91.5%.

This superior robustness is attributed to the Adaptive Kernel design and the Global Average Pooling layer, which effectively filters out high-frequency noise components while retaining the dominant fault signatures. This result indicates that the proposed method is robust under simulated noisy conditions. However, since AWGN cannot fully represent the non-stationary and multi-source disturbances in offshore environments, further validation with in-situ offshore data is still required.

It is worth noting that while AWGN provides a standard benchmark for robustness, real-world offshore noise is non-stationary. Future work will verify the model using in-situ noise data collected from operating wind farms.

C. COMPUTATIONAL EFFICIENCY AND BANDWIDTH OPTIMIZATION

Considering that in a real-world scenario, the turbine operates in a healthy state for the majority of the time (transmitting only 4-byte status codes) and rarely transmits raw data for expert analysis (assumed probability $p \approx 1\%$), the effective data transmission volume ratio (R) is defined as:

$$R = \frac{p \cdot B_{\text{raw}} + (1-p) \cdot B_{\text{status}}}{B_{\text{raw}}} \approx 1.1\%. \qquad (9)$$

Under the simplified payload-level assumptions adopted in this study, the proposed method reduces the application-layer payload volume by approximately 98.9% relative to continuous raw-data transmission. It should be noted that this estimate does not include protocol overhead, packet headers, retransmissions, encryption, or network-management traffic, and therefore should be interpreted as an idealized payload-level estimate rather than an end-to-end link-level bandwidth reduction.

TABLE 3. Ablation Study Results.

Model Variant	Description	Accuracy (%)	Model Size (MB)
Proposed Full Model	Adaptive 1D-CNN + GAP	99.25	0.45
Model A (w/o GAP)	Replaced GAP with FC Layers	97.10	14.20
Model B (w/o Adaptive)	Standard Fixed Kernels	96.85	0.42

D. ABLATION STUDY

To strictly verify the effectiveness of the key components proposed in our method—specifically the Global Average Pooling (GAP) strategy and the Adaptive Feature Extraction mechanism—we conducted an ablation study. We designed two variant models for comparison with the proposed full model:

1. Model A (w/o GAP): The GAP layer was removed and replaced with traditional Fully Connected (FC) layers. This variant tests the contribution of GAP to model lightweighting and overfitting prevention.

2. Model B (w/o Adaptive): The adaptive convolution kernels were replaced with standard fixed-size kernels (3×1). This variant tests the necessity of the adaptive mechanism for capturing diverse fault signatures. The experimental results are presented in Table 3.

- Impact of GAP: As observed in Model A, removing the GAP layer resulted in a slight decrease in accuracy (from 99.25% to 97.10%). More critically, the model size exploded from 0.45 MB to 14.20 MB. This indicates that GAP is a key factor in reducing parameter count and model size, and it contributes to edge deployment feasibility when considered together with inference-time and memory-related indicators. The FC layers introduced a massive number of parameters, which not only consumed memory but also led to slight overfitting.
- Impact of Adaptive Mechanism: Model B shows the lowest accuracy (96.85%). This suggests that using a single fixed kernel size is less effective than the proposed multi-scale configuration for capturing fault-related patterns with different temporal characteristics. In this study, the kernel-size setting of {3, 5, 7} is treated as an empirically effective design rather than a theoretically derived optimal solution. The proposed adaptive mechanism effectively resolves this by adjusting to different feature scales, thereby improving diagnostic precision.

In conclusion, both the GAP layer and the Adaptive mechanism are indispensable components. The full model

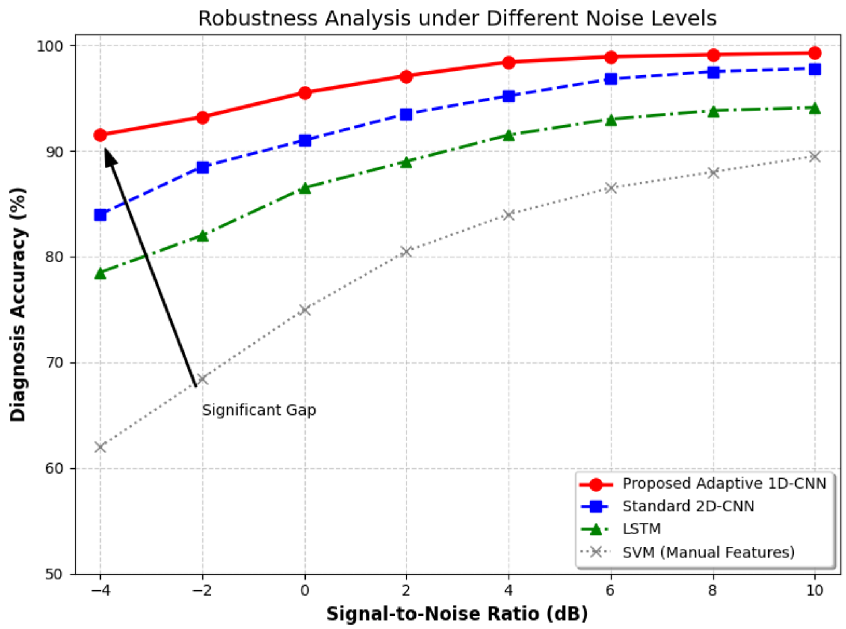

FIGURE 2. Diagnosis accuracy comparison of four different methods under varying Signal-to-Noise Ratios (SNR).

achieves the optimal balance between high accuracy and low computational cost. The accuracy comparison is visually summarized in Figure 3, highlighting the superiority of the integrated architecture.

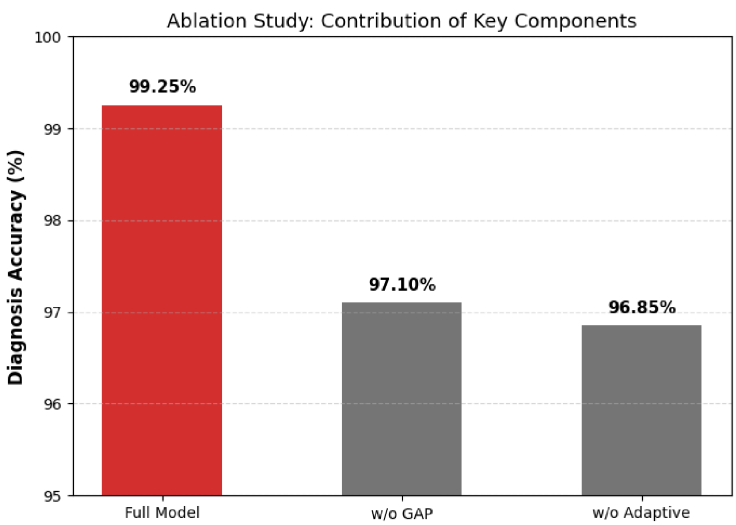

FIGURE 3. Ablation study results showing the impact of the Global Average Pooling (GAP) layer and the Adaptive mechanism on fault diagnosis accuracy.

VI. CONCLUSION AND FUTURE WORK

A. CONCLUSION

This paper proposed an IoT-based intelligent monitoring and fault detection method tailored for the challenging environment of offshore wind turbines. By establishing an Edge-Cloud collaborative architecture and designing a lightweight Adaptive 1D-CNN, we addressed the critical issues of transmission latency and weak fault feature extraction. The key findings are:

1. The proposed Edge-Cloud architecture effectively alleviates the bandwidth pressure of offshore communication networks by shifting the primary diagnostic tasks to the edge.

2. The Adaptive 1D-CNN model achieves an accuracy of 99.25% on the bearing dataset, outperforming the selected baselines in this controlled benchmark. The method also demonstrates strong computational efficiency on the Raspberry Pi 4B platform, indicating its potential suitability for deployment on resource-constrained embedded devices.

B. FUTURE WORK

Future research will focus on two aspects:

1. Multi-Sensor Fusion: Integrating temperature and acoustic emission data with vibration signals to further enhance diagnostic reliability.

2. Unsupervised Online Learning: Developing unsupervised transfer learning algorithms to allow the edge model to adapt to new, unseen fault types without manual labeling from the cloud.

AUTHOR CONTRIBUTIONS

Conceptualization –Ouyang Yilei and Liang Wei; methodology – Ouyang Yilei and Liang Wei; investigation – Ouyang Yilei and Liang Wei; writing-original draft preparation – Ouyang Yilei and Liang Wei. All authors have read and agreed to the published version of the manuscript.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

FUNDING

This research received no external funding.

ACKNOWLEDGEMENTS

Not applicable.

REFERENCES

[1] B. Desalegn, D. Gebeyehu, et al., “Onshore versus offshore wind power trends and recent study practices in modeling of wind turbines life-cycle impact assessments,” Cleaner Engineering and Technology, vol. 17, 100691, 2023, doi: 10.1016/j.clet.2023.100691.

[2] M. D. Esteban, J. J. Diez, et al., “Why offshore wind energy?” Renewable Energy, vol. 36, no. 2, pp. 444-50, 2011, doi: 10.1016/j.renene.2010.07.009.

[3] A. Platis, M. Hundhausen, et al., “The Role of Atmospheric Stability and Turbulence in Offshore Wind-Farm Wakes in the German Bight,” Boundary-Layer Meteorology, vol. 182, no. 3, pp. 441-69, 2022, doi: 10.1007/s10546-021-00668-4.

[4] B. Troskie, R. Toftegaard, et al., “Corrosivity Control Inside Offshore Wind Power Turbines by Combined Dehumidification and Overpressure,” Materials and Corrosion, vol. 76, no. 6, pp. 851-7, 2025, doi: 10.1002/maco.202314256.

[5] B. Syrek-Gerstenkorn and S. Paul, “Metallic coatings in offshore wind sector-a mini review,” npj Materials Degradation, vol. 8, no. 1, pp. 86, 2024, doi: 10.1038/s41529-024-00480-8.

[6] Z. Ren, A. S. Verma, et al., “Offshore wind turbine operations and maintenance: A state-of-the-art review,” Renewable and Sustainable Energy Reviews, vol. 144, 110886, 2021, doi: 10.1016/j.rser.2021.110886.

[7] T. Stehly, P. Duffy, et al., “Cost of Wind Energy Review: 2024 Edition,” National Renewable Energy Laboratory (NREL), Golden, CO (United States); 2024, doi: 10.2172/2278805.

[8] J. M. Pinar Pérez, F. P. García Márquez, et al., “Wind turbine reliability analysis,” Renewable and Sustainable Energy Reviews, vol. 23, pp. 463-72, 2013, doi: 10.1016/j.rser.2013.03.018.

[9] W. Qiao and D. Lu, “A Survey on Wind Turbine Condition Monitoring and Fault Diagnosis-Part I: Components and Subsystems,” IEEE Transactions on Industrial Electronics, vol. 62, no. 10, pp. 6536-45, 2015, doi: 10.1109/TIE.2015.2422112.

[10] W. Qiao and D. Lu, “A Survey on Wind Turbine Condition Monitoring and Fault Diagnosis-Part II: Signals and Signal Processing Methods,” IEEE Transactions on Industrial Electronics, vol. 62, no. 10, pp. 6546-57, 2015, doi: 10.1109/TIE.2015.2422394.

[11] A. K. S. Jardine, D. Lin, et al., “A review on machinery diagnostics and prognostics implementing condition-based maintenance,” Mechanical Systems and Signal Processing, vol. 20, no. 7, pp. 1483-510, 2006, doi: 10.1016/j.ymssp.2005.09.012.

[12] M. A. Ullah, K. Mikhaylov, et al., “Enabling mMTC in Remote Areas: LoRaWAN and LEO Satellite Integration for Offshore Wind Farm Monitoring,” IEEE Transactions on Industrial Informatics, vol. 18, no. 6, pp. 3744-53, 2022, doi: 10.1109/TII.2021.3112386.

[13] M. A. Ahmed and Y.-C. Kim, “Communication Network Architectures for Smart-Wind Power Farms,” Energies, vol. 7, no. 6, pp. 3900-21, 2014, doi: 10.3390/en7063900.

[14] Z. K. Peng, P. W. Tse, et al., “An improved Hilbert-Huang transform and its application in vibration signal analysis,” Journal of Sound and Vibration, vol. 286, no. 1, pp. 187-205, 2005, doi: 10.1016/j.jsv.2004.10.005.

[15] Y. Zhang, Y. Han, et al., “A dynamic threshold method for wind turbine fault detection based on spatial-temporal neural network,” Journal of Renewable and Sustainable Energy, vol. 14, no. 5, 2022, doi: 10.1063/5.0100389.

[16] A. M. Peco Chacón, I. Segovia Ramirez, et al., “False alarm detection in wind turbine by classification models,” Advances in Engineering Software, vol. 177, 103409, 2023, doi: 10.1016/j.advengsoft.2023.103409.

[17] K. Rama Krishna and K. I. Ramachandran, “Machinery Bearing Fault Diagnosis Using Variational Mode Decomposition and Support Vector Machine as a Classifier,” IOP Conference Series: Materials Science and Engineering, vol. 310, no. 1, 012076, 2018, doi: 10.1088/1757-899X/310/1/012076.

[18] L. Kou, Y. Li, et al., “Review on Monitoring, Operation and Maintenance of Smart Offshore Wind Farms,” Sensors, vol. 22, no. 8, pp. 2822, 2022, doi: 10.3390/s22082822.

[19] J. Maldonado-Correa, S. Martín-Martínez, et al., “Using SCADA Data for Wind Turbine Condition Monitoring: A Systematic Literature Review,” Energies, vol. 13, no. 12, pp. 3132, 2020, doi: 10.3390/en13123132.

[20] A. Verma, D. Zappalá, et al., “Wind turbine gearbox fault prognosis using high-frequency SCADA data,” Journal of Physics: Conference Series, vol. 2265, no. 3, 032067, 2022, doi: 10.1088/1742-6596/2265/3/032067.

[21] R. Pandit, D. Astolfi, et al., “SCADA data for wind turbine data-driven condition/performance monitoring: A review on state-of-art, challenges and future trends,” Wind Engineering, vol. 47, no. 2, pp. 422-41, 2023, doi: 10.1177/0309524x221124031.

[22] Y. Song, B. Wang, et al., “Remotely monitoring offshore wind turbines via ZigBee networks embedded with an advanced routing strategy,” Journal of Renewable and Sustainable Energy, vol. 5, no. 1, 2013, doi: 10.1063/1.4773467.

[23] K. Mikhaylov, A. Moiz, et al., “LoRa WAN for Wind Turbine Monitoring: Prototype and Practical Deployment,” 2018 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT); 2018 5-9 Nov, 2018, doi: 10.1109/ICUMT.2018.8631240.

[24] C. Castellano-Aldave, A. Plaza, et al., “Low-frequency electromagnetic harvester for wind turbine vibrations,” Micro and Nano Engineering, vol. 25, 100287, 2024, doi: 10.1016/j.mne.2024.100287.

[25] S. W. Ali, M. Sadiq, Y. Terriche, S. A. R. Naqvi, L. Q. N. Hoang, and M. U. Mutarraf, “Offshore Wind Farm-Grid Integration: A Review on Infrastructure, Challenges, and Grid Solutions,” IEEE Access, vol. 9, pp. 102811-102827, 2021, doi: 10.1109/access.2021.3098705.

[26] T. Verstraeten, F. Gomez Marulanda, et al., “Edge computing for advanced vibration signal processing,” Surveillance, Vishno and AVE conferences; 2019 2019-07-08; Lyon, France. https://hal.science/hal-02188766v1/document.

[27] Y. Lei, J. Lin, et al., “A review on empirical mode decomposition in fault diagnosis of rotating machinery,” Mechanical Systems and Signal Processing, vol. 35, no. 1, pp. 108-26, 2013, doi: 10.1016/j.ymssp.2012.09.015.

[28] J. Antoni and R. B. Randall, “The spectral kurtosis: application to the vibratory surveillance and diagnostics of rotating machines,” Mechanical Systems and Signal Processing, vol. 20, no. 2, pp. 308-31, 2006, doi: 10.1016/j.ymssp.2004.09.002.

[29] A. Stetco, F. Dinmohammadi, et al., “Machine learning methods for wind turbine condition monitoring: A review,” Renewable Energy, vol. 133, pp. 620-35, 2019, doi: 10.1016/j.renene.2018.10.047.

[30] R. V. Sánchez, J. C. Macancela, et al., “Evaluation of Hand-Crafted Feature Extraction for Fault Diagnosis in Rotating Machinery: A Survey,” Sensors (Basel), vol. 24, no. 16, 2024, doi: 10.3390/s24165400.

[31] X. Wang, C. Wang, et al., “An Adversarial Single-Domain Generalization Network for Fault Diagnosis of Wind Turbine Gearboxes,” Journal of Marine Science and Engineering, vol. 11, no. 12, pp. 2384, 2023, doi: 10.3390/jmse11122384.

[32] L. Zhang, L. Zhang, et al., “Deep Learning for Remote Sensing Data: A Technical Tutorial on the State of the Art,” IEEE Geoscience

and Remote Sensing Magazine, vol. 4, no. 2, pp. 22-40, 2016, doi: 10.1109/MGRS.2016.2540798.
[33] H. Jung, S. Choi, et al., "Rotor Fault Diagnosis Method Using CNN-Based Transfer Learning with 2D Sound Spectrogram Analysis," Electronics, vol. 12, no. 3, pp. 480, 2023, doi: 10.3390/electronics12030480.
[34] V. Sze, Y. H. Chen, et al., "Efficient Processing of Deep Neural Networks: A Tutorial and Survey," Proceedings of the IEEE, vol. 105, no. 12, pp. 2295-329, 2017, doi: 10.1109/JPROC.2017.2761740.
[35] N. D. Lane, S. Bhattacharya, et al., "DeepX: A Software Accelerator for Low-Power Deep Learning Inference on Mobile Devices," 2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN); 2016 11-14 April 2016, doi: 10.1109/IPSN.2016.7460664.
[36] G. Shubbham and S. Shiv Naresh, "Embedded TinyML for Predictive Maintenance: Vibration Analysis on ESP32 with Real-Time Fault Detection in Industrial Equipment," International Journal on Computational Modelling Applications, vol. 2, no. 2, pp. 1-17, 2025, doi: 10.63503/j.ijcma.2025.114.