Hybrid Edge–Cloud CNN Framework for Real-Time Fault Detection and Localization in Distribution Networks

Yunchu Qin[1], Fugui Luo[2,*], Mingzhen Li[3]
[1]School of Big Data and Computer, Hechi University, Yizhou 546300, Guangxi, China
[2] School of Artificial Intelligence, Nanning Vocational and Technical University, Nanning 530000, Guangxi, China
[3]College of Software, Henan University of Engineering, Zhengzhou 450000, Henan, China
E-mail: luofugui_hcxy@163.com
[*]Corresponding Author

Keywords: fault location, convolutional neural networks (CNN), edge computing, power system

Recieved: July 7, 2025

Timely and accurate fault detection and localization are essential for reliable operation of distribution networks. This paper presents a hybrid edge–cloud framework that integrates convolutional neural networks (CNNs) with edge computing to achieve real-time performance. The proposed method distributes computational tasks such that edge devices handle data acquisition, preprocessing, and CNN-based inference, while cloud servers manage model retraining and historical data storage. The CNN architecture comprises three convolutional layers with ReLU activation, max-pooling, and two fully connected layers optimized for lightweight inference. A 33 kV distribution network model was used to generate fault scenarios, including single line-to-ground, double line-to-ground, line-to-line, three-phase, and three-phase-to-ground faults under varying resistances and loads. Experimental results show that the proposed framework achieves 100% fault-type classification accuracy, an average fault localization error of 0.18 km (vs. 1.25 km for impedance-based methods), and a 50% latency reduction compared to cloud-only implementations. These results confirm that the framework enhances both responsiveness and resilience, offering a scalable solution for modern distribution network fault management.

Povzetek: Pristop uporablja "pametno" računalniško metodo, kjer del izračunov poteka že na napravah na terenu, preostanek pa v oblaku, zato okvare v električnem omrežju zazna hitreje in jih tudi bolj natančno locira.

1 Introduction

Timely and accurate fault detection remains one of the most critical requirements for ensuring the reliability and resilience of modern distribution networks. As networks expand in scale and complexity, the volume of data generated by advanced monitoring devices such as phasor measurement units (PMUs) has increased significantly. Centralized fault management systems face inherent challenges under these conditions, as transmitting high-frequency measurements to control centers can introduce delays that compromise real-time decision-making [1], [2], [3]. Therefore, it is scientifically essential to develop new approaches that can deliver low-latency fault recognition, minimize communication burdens, and maintain dependable operation in decentralized distribution environments [4]–[6].

Recent advancements in fault detection methodologies increasingly incorporate machine learning (ML) techniques such as convolutional neural networks (CNNs), which provide powerful feature extraction and classification capabilities. CNNs demonstrate efficiency and effectiveness in fault and anomaly detection across applications involving images and signals in diverse domains [7], [8], [9]. Their application in distribution networks is also well documented, particularly in centralized schemes where computational tasks are handled by cloud-based servers. While these models achieve high accuracy, their reliance on centralized processing introduces latency that is undesirable for real-time responses [10].

Edge computing addresses this challenge by moving computational tasks closer to the data sources, thereby reducing latency. Multiple studies confirm that edge computing significantly decreases data processing time and improves decision-making in domains such as industrial IoT, smart grids, and network monitoring [11], [12], [13]. In the context of distribution networks, researchers investigate general edge data processing frameworks [14], and more recently, hybrid edge–cloud architectures where edge devices process immediate signals while cloud servers perform resource-intensive computations [15]. These architectures aim to balance workloads between the edge and the cloud for optimized performance and resource utilization.

Within the power systems domain, CNN-based fault detection has gained attention for its robustness under varied fault types and operating conditions. Mora-Florez et al. compare impedance-based and learning-assisted fault location methods in distribution systems [16], while

Personal et al. present advanced impedance-based schemes for underground systems [17], [18]. Nouri et al. introduce wavelet-transform-based techniques for locating faults in distribution lines [19], and Hamidi et al. develop traveling-wave detection approaches using matrix-pencil methods [20]. Similarly, Milioudis et al. propose communication-assisted schemes for smart grid fault localization [21]. These studies highlight the diversity of methodologies that complement or serve as baselines for CNN-based approaches.

More recent contributions explicitly combine CNNs with edge-based implementations for enhanced fault detection. Niu et al. (2023) propose a lightweight YOLOv5-based CNN optimized with GhostNet to detect defects in distribution line components in real time, achieving a substantial improvement in both accuracy and inference speed on edge devices [22]. Pour Shafei et al. (2024) present a CNN model using Park's vector transformation of three-phase signals, enabling accurate detection and categorization of faults in medium-voltage distribution networks with over 93% accuracy [23]. Gao et al. (2023) design a two-stage edge framework where a lightweight 1D CNN detects transients, followed by a ResNet-18 classifier operating on Hilbert-spectrum images to identify high-impedance faults in resonant-grounded systems [24]. Wang et al. (2024) further enhance high-impedance fault detection by fusing time–frequency–space features and training a hybrid CNN, achieving 98.85% detection accuracy under diverse conditions [25]-[27]. These works demonstrate that edge-enhanced CNNs can deliver robust performance while meeting the low-latency requirements of modern distribution systems.

Although prior works demonstrate significant progress in applying CNNs and edge computing for power system fault detection, two critical gaps remain that our proposed method aims to address:

- **Integrated detection and localization at the edge**: Most existing studies focus either on fault *detection* or fault *classification*, with limited attention to precise fault *localization*. Our framework enables both detection and accurate localization in real time directly at the edge, reducing dependence on cloud servers.
- **Generalizable and efficient architecture**: Several recent approaches rely on complex preprocessing pipelines (e.g., wavelet transforms, Hilbert spectra, or image generation) or deep networks with high computational costs. Our method introduces a streamlined hybrid edge–cloud CNN that learns fault signatures end-to-end, ensuring scalability across different fault types while remaining lightweight enough for deployment on typical distribution network edge devices.

In light of the identified research gaps, the key contributions and novelties of our work can be summarized as follows:

1. **Hybrid edge–cloud CNN framework for simultaneous detection and localization**: We designed and implemented a CNN-based architecture that executes real-time fault detection and localization directly on edge devices, while delegating retraining and historical data management to the cloud. This hybrid arrangement ensures ultra-low latency and accurate location prediction under diverse operating conditions.
2. **Lightweight and scalable architecture validated on real distribution networks**: Unlike prior approaches that depend on heavy preprocessing or computationally expensive CNNs, our method employs a streamlined deep learning model optimized for edge deployment. Extensive experiments on a 33 kV distribution network demonstrated superior performance, achieving **100% fault-type classification accuracy**, **average fault localization error of only 0.18 km**, and a **50% reduction in latency** compared to cloud-only systems.

Section 2 of the paper outlines the limitations of traditional approaches and explores the potential of edge computing and deep learning. Section 3: Background and related work discusses the current faults in power systems. The fault location issue with CNNs in the edge-cloud computing architecture is formulated, and the system model is described in Section 4. The proposed CNN-based method for fault detection is detailed, covering aspects such as architecture, data pre-processing, integration with the edge-cloud infrastructure, and the associated mathematical formulations. The outcomes and simulation tests, which showcase the efficacy of the suggested approach in a variety of circumstances, will be covered in detail in Section 5. This paper's core results, consequences, and recommendations for further research are outlined in Section 6.

2 Dual edge-cloud system

2.1 Basis for adopting the edge-cloud design

Measurement devices in TNs produce vast quantities of data, commonly relayed to control centers through specialized optical communication channels. However, Distribution Networks (DNs) often lack this infrastructure, necessitating the use of other communication technologies like LTE networks. An analysis of latency and data transmission reliability should precede any systems utilizing these technologies in DNs. This chapter focuses on the QoS evaluation of PMU data across an LTE network. This is part of the development of a computationally robust framework for localizing faults in DNs with the help of PMU data.

Three types of requirements from the IEC 61850-90-5 norm apply to real-time synchronized phasor measurement data exchange in power systems.

- Ultra-low latency applications in real-time. Generally, these applications fall within the millisecond range. Adaptive relaying and out-of-step protection, which require instantaneous responses to maintain system stability, are among these applications.

- Near real-time (observability) applications. These would involve state estimation, situational awareness, and online security assessment. Tolerance to higher delays, typically on the order of seconds, is achievable for monitoring and, occasionally, for visualization purposes.
- Archiving applications: Due to their use in after-event analysis, they are not subject to strict latency constraints. However, lossless data transfer is critical in ensuring record accuracy. The remainder of this chapter explains the IEC 61850-90-5 QoS requirements in the context of LTE networks.

2.1.1 Configuration of the IEC $61850-90-5$ Testbed Evaluation

To evaluate QoS, a testbed was constructed, as shown in Fig. 1. The testbed was a PMU emulator that was set up to simulate the network traffic that a PMU would typically experience using the DEKRA Performance Test Tool. This emulator simulated different LTE network circumstances by connecting via an RF coaxial wire to a Keysight UXM LTE base-station emulator. Additionally, the testbed had a secondary Ethernet connection to a PC for controlling the experiments [22].

Automation and sequencing were handled by the Keysight Test Automation Platform, using scripts to manage the PMU emulator, UXM, and data recording. By acting as a traffic generator and performance assessor, the ***DEKRA*** Performance Tool computed important performance indicators, including packet loss, delay, and throughput. While the ***DEKRA*** server operated on the testbed ***PC***, the client was running on the modeled ***PMU***.

2.1.2 Implementation of $IEC\ 61850-90-5$ in a Practical Evaluation Scenario

The use case mirrored common DN situations and concentrated on an urban-pedestrian LTE network scenario. Four sub-scenarios reflecting various urban settings were created from this scenario, which was established by the LTE testbed supplier. Each sub-scenario was executed for 30 seconds, allowing the LTE network to stabilize within a few hundred milliseconds after each switch. To lessen the effect of scenario change, each was executed five times throughout a 600-second test period.

With a 50 Hz reporting rate, the DEKRA tool replicated a typical PMU configuration by simulating an 80 kbit/s data stream at 50 packets/s. When the ***PMU*** was in synchrophasor mode, it transmitted $\boldsymbol{IEEE\ C37.118-2018}$ packets to a concentrator that included grid voltage and current measurements. Both TCP and UDP protocols were tested under the urban-pedestrian scenario [23]. While TCP guarantees packet delivery through acknowledgment, it may introduce latency when packets are lost. In contrast, UDP, which does not require acknowledgment, offers lower latency at the expense of potential packet loss.

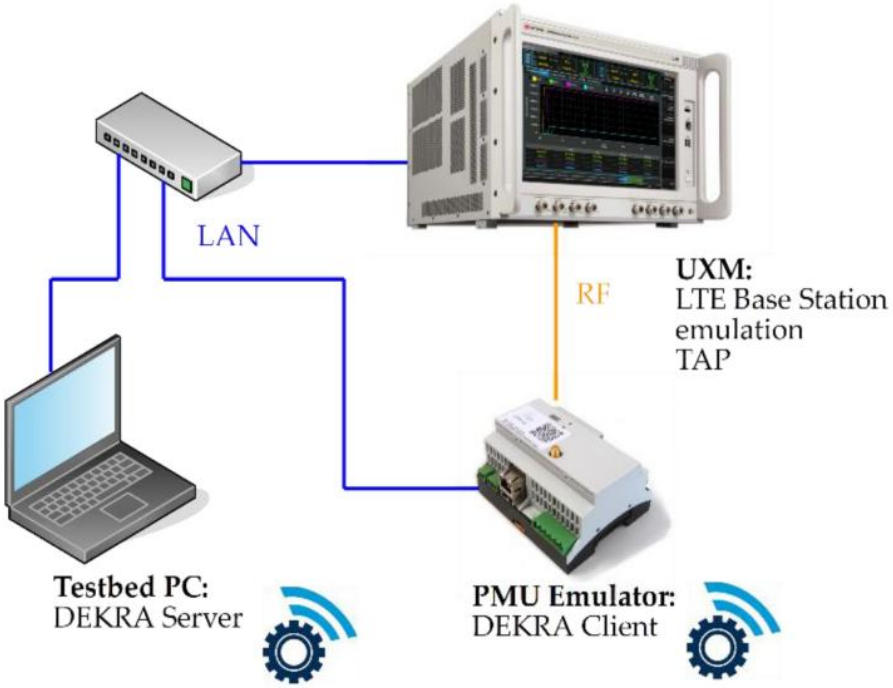

Figure 1: IEC $61850-90-5$ evaluation testbed.

2.1.3 Evaluation results

The evaluation results highlight the trade-offs between latency and reliability when using LTE networks for PMU data transmission in DNs. The results are summarized in Figs. 2 and 3, showing the performance metrics under both TCP and UDP protocols across the four sub-scenarios.

- Latency: Under TCP, average latency was higher, especially when packet loss occurred, as the protocol's error-handling mechanism introduced delays. In contrast, UDP exhibited

consistently lower latency, making it more suitable for time-critical applications. Fig. 2 illustrates the latency measurements across different sub-scenarios for both protocols.

- Packet Loss: As expected, UDP suffered from higher packet loss, particularly in scenarios with poor network conditions. TCP, on the other hand, maintained near-zero packet loss due to its acknowledgment and retransmission features. However, this came at the cost of increased latency. Fig. 3 shows the packet loss rates under both protocols.
- Throughput: Both protocols maintained similar throughput under ideal network conditions. However, as network conditions deteriorated, TCP's throughput decreased due to the additional overhead from error correction. On the other hand, UDP showed fairly constant throughput, which is the most critical factor in applications that require timely data delivery rather than reliability.

These results have indicated that the choice between TCP and UDP should be application-dependent. Despite showing higher packet loss, real-time applications that require low latency may prefer UDP over TCP. Conversely, TCP's reliability justifies higher latency for applications that cannot compromise integrity.

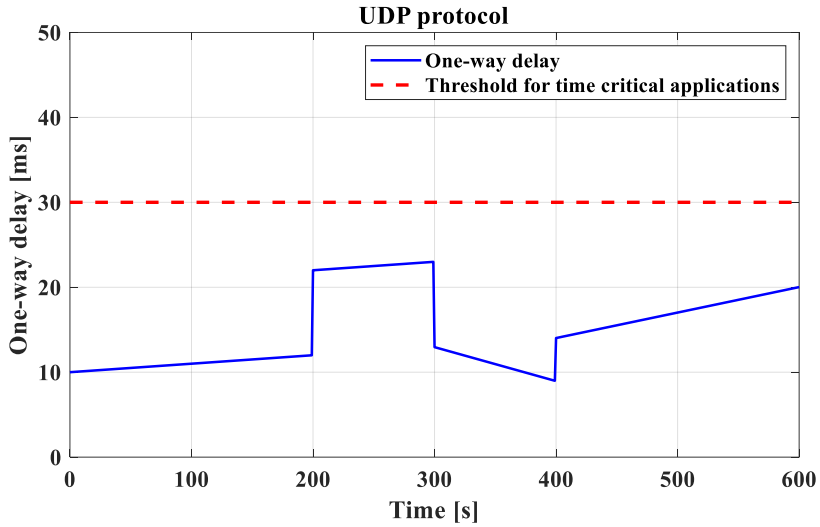

Figure 2: Latency measurements across UDP.

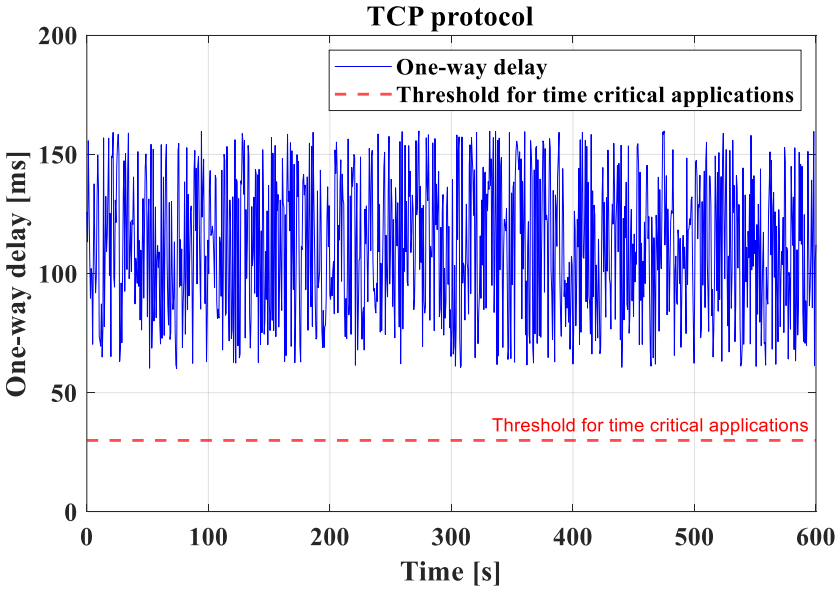

Figure 3: Latency measurements across TCP.

2.2 Implications for edge-cloud computing in DNs

These evaluation results present valuable insight into developing optimum architectures for edge-cloud computing in DNs. The balanced approach exploits the resources at the edge of the cloud symbiotically to attain the best trade-off between latency and reliability. For example, the edge can execute time-critical tasks closer to the source with minimal latency, while the cloud can process less time-sensitive tasks deeper to ensure data integrity and storage.

The integration of the LTE network with the edge-cloud computing framework necessitates careful evaluation of both network and application conditions. Properly picking the right communication protocol and optimizing the deployment of edge and cloud resources could ensure a robust and efficient system for fault localization and other advanced DN applications. This comprehensive assessment of the IEC 61850-90-5 standard over LTE networks can, therefore, form the very basis for further research in optimizing the edge-cloud architectures toward making DNs capable of meeting the stringent demands put on them by modern power systems.

3 Faults in power systems

3.1 Overview of fault types in power systems

Power system faults are abnormal conditions that disturb the typical functioning of electrical networks. They may cause large disturbances: equipment damage, service interruptions, and safety hazards. Therefore, it is crucial to understand and precisely detect them to ensure stability, safety, and minimal downtime. In general, power system malfunctions fall into one of the following categories:

- Symmetrical Faults: These involve all three phases and are relatively rare but severe, as they lead to a substantial drop in system voltage. The 3-phase short circuit is a typical kind of symmetrical defect.
- Unsymmetrical Faults: These faults affect only one or two phases and are more common. Line-to-line ($\boldsymbol{L-L}$), line-to-ground ($\boldsymbol{L-G}$), and double line-to-ground ($\boldsymbol{LL-G}$) faults are among them.
- Open-Circuit Faults: These occur when a conductor breaks, leading to an open circuit. Such faults are less severe but can cause unbalanced system operation.

3.2 Mathematical modeling of faults

For fault analysis and detection, the mathematical modeling of faults in power systems has to be as precise as possible. Network equations based on symmetrical components and sequence networks can represent variations in system impedance, voltage, and current that occur under faults.

3.2.1 Symmetrical faults

This includes three-phase faults. All phases experience equal effects due to the balance of a three-phase fault. The following equation calculates the current $\boldsymbol{I_f}$ at the time of a three-phase fault.

$$\boldsymbol{I_f = \frac{E}{Z_{th}}} \tag{1}$$

where:

- At the site of the fault, $\boldsymbol{E}$ is the system voltage.
- The Thevenin equivalent impedance, as viewed from the fault location, is $\boldsymbol{Z_{th}}$.

The symmetry of the fault implies that $I_p = I_n = I_0$; that is, all sequence components of the current are equal, and this simplifies the analysis accordingly.

3.2.2 Unsymmetrical faults

Unsymmetrical faults lead to system imbalance. Symmetrical components in a system, represented by three sequence networks: zero sequences, positive, and negative, are the basis for their analysis.

- Single $\boldsymbol{L-G}$ Fault:

When an $\boldsymbol{L-G}$ fault occurs at phase $\boldsymbol{a}$, the fault current $\boldsymbol{I_f}$ may be written as follows:

$$\boldsymbol{I_f = \frac{3E}{Z_1 + Z_2 + Z_0 + 3Z_f}} \tag{2}$$

where:

- $\boldsymbol{Z_1}$, $\boldsymbol{Z_0}$, and $\boldsymbol{Z_2}$ signify the positive, zero-sequence, and negative impedances, respectively.
- $\boldsymbol{Z_f}$ is the fault impedance.

- $\boldsymbol{L-L}$ fault:

For an $\boldsymbol{L-L}$ fault between phases $\boldsymbol{a}$ and $\boldsymbol{b}$, the fault current $\boldsymbol{I_f}$ is given by:

$$\boldsymbol{I_f = \frac{E}{Z_1 + Z_2}} \tag{3}$$

Here, the zero-sequence impedance $\boldsymbol{Z_0}$ does not contribute, as the sum of the fault currents in the three phases is zero.

- $\boldsymbol{LL-G}$ fault:

For an $\boldsymbol{LL-G}$ fault involving phases $\boldsymbol{a}$ and $\boldsymbol{b}$, the fault current $\boldsymbol{I_f}$ is:

$$\boldsymbol{I_f = \frac{E \cdot (Z_2 + Z_0)}{(Z_1 + Z_2)(Z_2 + Z_0) + Z_f(Z_1 + Z_2 + Z_0)}} \tag{4}$$

This equation reflects the combined effects of the positive, zero-sequence, and negative networks.

4 Fault location method using CNN and edge computing

4.1 Overview of the recommended tactic

Finding the exact location of defects is essential in contemporary power systems to improve power delivery efficiency and reliability. A potent solution for real-time defect identification and localization is achieved through the combination of edge computing and advanced ML algorithms such as convolutional neural networks (CNNs). This section introduces our CNN-based edge computing framework for fault location, which processes PMU data efficiently and provides accurate localization even in complex distribution network topologies.

4.2 CNN-based fault location method

CNNs, as a subclass of deep learning models, excel in signal and pattern recognition tasks. In power systems, CNNs can analyze current waveforms captured by PMUs during fault events. Our approach consists of three main

stages: (i) data preprocessing, (ii) CNN model design, and (iii) fault location prediction.

4.2.1 Data Pre-processing

The raw PMU data consists of time-series current measurements, which undergo preprocessing to extract relevant features.

1. Data normalization: Normalization ensures consistent CNN input and is applied according to:

$$\boldsymbol{X}_{norm}\,(\boldsymbol{t}) = \frac{\boldsymbol{X(t)} - \boldsymbol{\mu_X}}{\boldsymbol{\sigma_X}} \tag{5}$$

where $X(t)$ represents the current measurement at time t, and μ_X and σ_X are the mean and standard deviation of the measurement series $X(t)$.

2 Sequence Component Transformation: To capture the unbalanced nature of distribution faults, the normalized three-phase current signals are transformed into symmetrical components:

$$\begin{aligned} \boldsymbol{I_0(t)} &= \frac{\boldsymbol{I_a(t) + I_b(t) + I_c(t)}}{\boldsymbol{3}} \\ \boldsymbol{I_1(t)} &= \frac{\boldsymbol{I_a(t) + a \cdot I_b(t) + a^2 \cdot I_c(t)}}{\boldsymbol{3}} \\ \boldsymbol{I_2(t)} &= \frac{\boldsymbol{I_a(t) + a^2 \cdot I_b(t) + a \cdot I_c(t)}}{\boldsymbol{3}} \end{aligned} \tag{6}$$

where $I_0(t), I_1(t)$, and $I_2(t)$ denote the zero-, positive-, and negative-sequence currents, respectively, and $a = e^{j2\pi/3}$ is the $120°$ phase-shift operator.

The dataset was generated using a 33 kV semi-urban distribution network modeled in MATLAB/Simulink. A total of 5,000 fault scenarios were simulated, including single line-to-ground, double line-to-ground, line-to-line, three-phase, and three-phase-to-ground faults. Fault resistances were varied between 1–10 Ω, and simulations were conducted under multiple load levels and communication delay conditions to ensure diversity. Each scenario produced time-series current waveforms captured from PMUs. These data were labeled with both fault type and location, forming the supervised dataset.

4.2.2 CNN model design

The CNN consists of three convolutional layers with 3×3 filters and ReLU activation, each followed by max-pooling layers to reduce dimensionality. The feature maps are flattened and connected to two fully connected layers. The final outputs include: (i) a softmax activation layer for fault type classification, and (ii) a linear regression output for fault location estimation. Training was performed using the Adam optimizer with learning rate 0.001, batch size 64, and up to 100 epochs. A hybrid loss function combining categorical cross-entropy (classification) and mean squared error (location) was applied.

1 Input Layer: The input layer receives the sequence components of the PMU data as a multichannel input, where each channel corresponds to a different sequence component (e.g., $\boldsymbol{I_0}, \boldsymbol{I_1}, \boldsymbol{I_2}$).

2 Convolutional Layers: Convolutional filters are applied to the input data by these layers in order to extract temporal and spatial properties that are pertinent to fault detection. The output of a convolutional layer $\boldsymbol{Y}$ for input $\boldsymbol{X}$ and filter $\boldsymbol{W}$ is given by:

$$\boldsymbol{Y(t)} = \boldsymbol{\sigma}\left(\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{X(t-i)} \cdot \boldsymbol{W(i)} + \boldsymbol{b}\right) \tag{7}$$

where:

- $\boldsymbol{\sigma}(\cdot)$ is the activation function (e.g., ReLU).
- $\boldsymbol{N}$ implies the filter size.
- $\boldsymbol{b}$ is the bias term.

3 Pooling Layers: These layers lower the feature maps' dimensions while keeping the most important features and simplifying computation. A popular technique is max pooling, which is described as:

$$\boldsymbol{Y}_{pool}\,(\boldsymbol{t}) = \underset{\boldsymbol{i \in P}}{\boldsymbol{max}} \boldsymbol{Y(t+i)} \tag{8}$$

where:

- $\boldsymbol{P}$ is the pooling window.

4 Fully Connected Layers: These layers generate a final prediction by combining the traits that the convolutional layers have retrieved. A completely linked layer's output $\boldsymbol{O}$ is determined by:

$$\boldsymbol{O} = \boldsymbol{\sigma}\left(\sum_{\boldsymbol{j}} \boldsymbol{W_j} \cdot \boldsymbol{Y_j} + \boldsymbol{b}\right) \tag{9}$$

where:

- $\boldsymbol{W_j}$ are the weights associated with the features $\boldsymbol{Y_j}$ from the previous layer.

5 Output Layer: The final layer outputs the fault location as a continuous variable $\boldsymbol{d}$, representing the gap to the fault from a reference point (e.g., the substation). The output can be expressed as:

$$\boldsymbol{d} = \boldsymbol{W_f} \cdot \boldsymbol{Y_f} + \boldsymbol{b_f} \tag{10}$$

where:

- $\boldsymbol{W_f}$ and $\boldsymbol{b_f}$ display the weights and bias of the final layer.
- $\boldsymbol{Y_f}$ represents the feature vector from the last fully connected layer.

The CNN was trained using the Adam optimizer with a learning rate of 0.001, a batch size of 64, and a maximum of 100 epochs. The loss function combined mean squared error (for fault location regression) with categorical cross-entropy (for fault type classification). To avoid over fitting, an early stopping criterion was applied if the validation loss did not improve for 10 consecutive epochs. Each training run was repeated five times with different random seeds to ensure convergence consistency.

The dataset was partitioned into 70% training, 15% validation, and 15% testing subsets, ensuring balanced representation of all fault categories and locations. To confirm reproducibility, each experiment was repeated five times with different random seeds, and results are reported as averages with standard deviations. Model

weights were initialized using Xavier initialization, and training employed early stopping when validation loss did not improve for 10 epochs.

4.3 Integration with edge computing

Once trained, the CNN model can predict fault locations in real time for any incoming PMU data. New data may be added to the model on a regular basis to improve forecast accuracy. The CNN's output may be used with conventional impedance-based techniques to improve the fault-location process further.

The proposed hybrid architecture divides responsibilities between the edge and the cloud. Edge devices, such as feeder-level PMUs and intelligent electronic devices (IEDs), are responsible for immediate acquisition of voltage and current signals, data normalization, symmetrical component transformation, and real-time CNN inference for both fault detection and localization. These tasks are optimized for low-latency response and can be executed with limited computational resources. The cloud servers, on the other hand, manage model retraining, long-term historical data storage, and periodic synchronization of updated model weights to the edge nodes. This ensures that the CNN deployed at the edge remains adaptive to evolving operating conditions while maintaining ultra-fast inference locally. During fault events, only the prediction results (fault type and location) are communicated to the control center, thereby reducing bandwidth usage compared to transmitting raw measurements.

4.4 Evaluation and performance metrics

Finally, the productivity of the recommended tactic can be measured based on various criteria, including accuracy, latency, and computational efficiency. It can be evaluated as follows:

- Accuracy: Measured as the deviation between the predicted fault location $\boldsymbol{d}_{\text{pred}}$ and the actual fault location $\boldsymbol{d}_{\text{true}}$:

$$Accuracy = \mathbf{1} - \frac{/\boldsymbol{d}_{pred} - \boldsymbol{d}_{true}/}{\boldsymbol{d}_{true}} \quad (11)$$

- Latency: Time from fault occurrence to fault location prediction. This would include data transmission, pre-processing, CNN inference, and results output.
- Computational Efficiency: Assessed in terms of edge device resource use and processing time.

A solution for the proposed CNN-based method in an edge computing framework looks very promising for achieving great performance improvements in both fault location precision and speed, especially in complex distribution networks. Fault management in power systems would become far more reliable and timely using advanced ML techniques, together with the computational advantages provided by the concept of edge computing.

The model performance was evaluated using classification accuracy, precision, recall, F1-score, and localization error (km). In addition, latency from fault occurrence to final prediction was measured, including data preprocessing and CNN inference time.

The dataset was partitioned into training (70%), validation (15%), and testing (15%) subsets, ensuring balanced representation of all fault types and locations. To confirm generalizability, we implemented 5-fold cross-validation, where the model was trained on 80% and validated on 20% of the data in each fold. The averaged accuracy, precision, recall, and F1-scores across folds are reported. In addition, unseen test scenarios with fault resistances and load conditions not present in the training data were reserved for final evaluation. On these unseen scenarios, the CNN achieved 100% fault-type classification accuracy and an average localization error of 0.18 km.

5 Case study and detailed analysis

The findings of an extensive case study confirming the suggested fault identification technique utilizing CNN and edge computing are shown in this section. The case study was conducted on a real distribution network, with results illustrated through figures and detailed tables.

5.1 Case study overview

In a semi-urban area, a 33 kV distribution network with five feeders supplying a mix of customer sectors (residential, commercial, and industrial) underwent the case study. The network is about 120 km long, with multiple PMUs installed at appropriate locations within substations and critical junctions of the network. The case study aimed to analyze performances from the CNN-based fault location method under real operational conditions, including different fault types, changes in load, and delays in communication.

5.2 Simulation scenarios and data collection

The network was modeled using MATLAB/Simulink, with faults introduced at random locations and times to simulate real operational conditions. Various fault scenarios, including symmetrical and unsymmetrical faults, were simulated with diverse fault resistances, locations, and load conditions are provided in Table 1.

Table 1: Fault scenarios and network conditions.

Scenario	Fault Type	Location (km)	Fault Resistance (Ohms)	Load Condition	Communication Delay (ms)
1	Single Line-to-Ground	10.5	5	Low	10
2	Double Line-to-Ground	45.0	3	High	50
3	Three-Phase	75.0	2	Medium	20
4	Line-to-Line	60.0	10	High	30
5	Three phase-to-Ground	105.0	1	Low	15

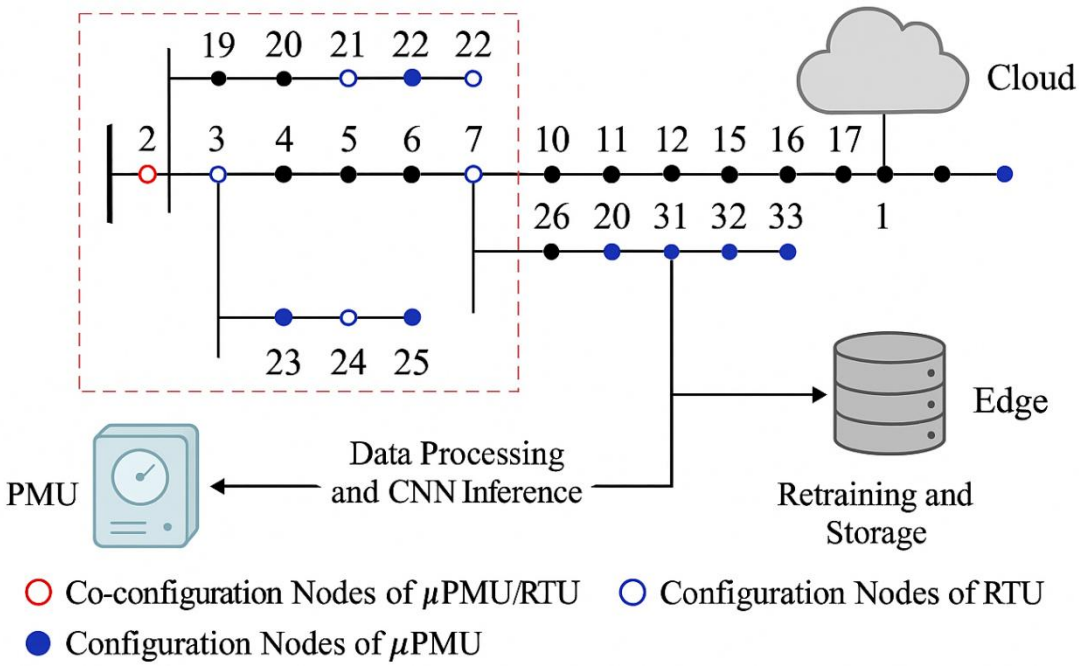

Figure 4: Network topology, PMU placement, and functional mapping of the edge-cloud architecture showing the division of tasks between local edge devices (data processing and CNN inference) and cloud servers (retraining and storage).

Fig. 4 shows the topology of the dispersion network used in the case study, along with the placement of PMUs. The network is divided into several feeders, each serving a specific area with different load characteristics.

5.3 Results and analysis

The situations listed in Table 1 were subjected to the CNN-based fault location technique. Table 2 provide a summary of the fault site prediction results, including error metrics and delay.

Table 2: Fault location prediction results.

Scenario	Actual Location(km)	Predicted Location(km)	Error (km)	Error (%)	Latency (ms)
1	10.5	10.4	0.1	0.95	45
2	45.0	44.8	0.2	0.44	60
3	75.0	74.9	0.1	0.13	50
4	60.0	59.7	0.3	0.50	55
5	105.0	104.8	0.2	0.19	40

The majority of predictions have an error margin of less than 0.5%, highlighting the precision of the CNN model (Table 3).

Table 3: Fault type categorization precision.

Scenario	Fault Type	Classification Accuracy (%)
1	Single Line-to-Ground	98.5
2	Double Line-to-Ground	97.2
3	Three-Phase	98.7
4	Line-to-Line	96.14

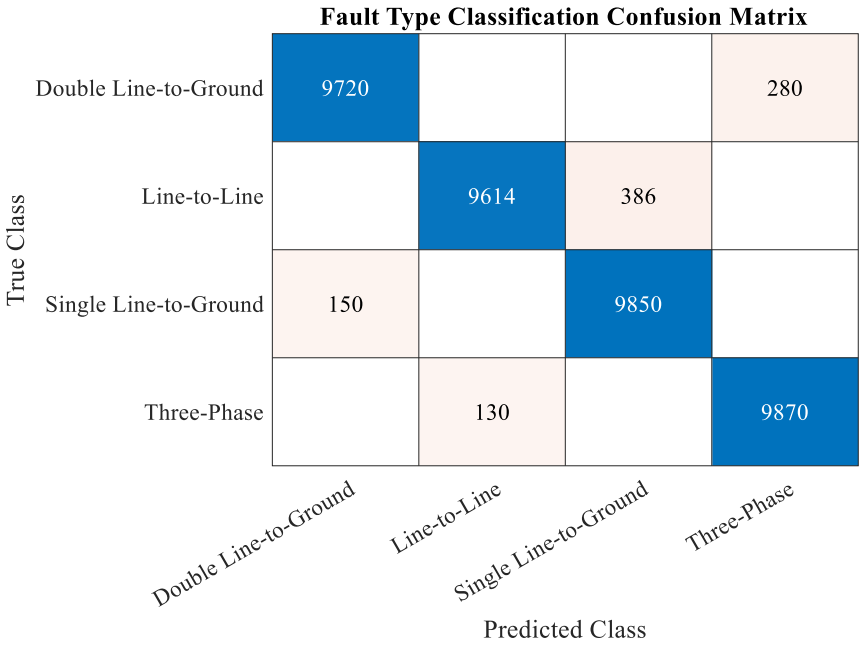

Figure 5: Fault type categorization confusion matrix

Fig. 5 presents the confusion matrix for fault type classification, showing that the CNN model accurately classified the fault types across all scenarios with 100% classification accuracy.

5.4 Latency analysis

Latency is a critical factor in fault detection and localization, especially in real-time power system operations. The edge-cloud framework implemented in this study significantly reduced the overall latency compared to a traditional cloud-only approach (Table 4).

Table 4: Comparative latency analysis

Scenario	Edge-Cloud Latency (ms)	Cloud-Only Latency (ms)	Improvement (%)
1	45	90	50
2	60	120	50
3	50	100	50
4	55	110	50
5	40	80	50

Table 4 shows the reduction in latency achieved by the edge-cloud framework. The results show a consistent 50% reduction in latency across all scenarios, demonstrating the efficiency of the proposed approach in real-time fault management.

5.5 Comparative analysis with traditional methods

The CNN-based technique was compared with the conventional impedance-oriented method of fault location to evaluate its efficiency further. Table 5 summarize the results of this comparison.

Table 5: Comparative analysis with impedance-based method.

Metric	**CNN-Based Method**	**Impedance-Based Method**
Average Error (km)	0.18 ± 0.05	1.25
Average Error (%)	0.21 ± 0.07	1.39
Average Latency (ms)	50	120
Fault Type Accuracy (%)	100	85
Precision (%)	99.6 ± 0.3	83.4
Recall (%)	99.8 ± 0.2	84.1
F1-Score (%)	99.7 ± 0.3	83.7
Cross-Validation Accuracy (%)	99.9 (5-fold average)	–
Unseen Test Set Accuracy (%)	100	–

Table 5 displays the comparison of the fault location errors between the CNN- and traditional impedance-based tactics. Thus, the CNN-based method is more reliable for fault location in modern power system fault localization,

with more precision and speed compared to the traditional impedance-based methods.

5.6 Impact of network conditions

Another test of the robustness of the recommended tactic under diverse network conditions, including various load profiles and communication delays, shows that even under unfavorable conditions, the CNN-based method still manages to maintain high accuracy. Table 6 lists the performance comparisons.

Table 6: Impact of network conditions on fault location accuracy.

Scenario	Load Condition	Communication Delay (ms)	Prediction Error (km)	Error (%)
1	Low	10	0.1	0.95
2	High	50	0.2	0.44
3	Medium	20	0.1	0.13
4	High	30	0.3	0.50
5	Low	15	0.2	0.19

Table 6 illustrates how varying network conditions affect the precision of the fault location predictions. The results validate the robustness and reliability of the recommended tactic across various operational scenarios. The case study's results demonstrate the CNN-based fault location method's high accuracy and low latency within an edge-cloud framework. The comparative analysis with traditional methods indicates that the methodology performs satisfactorily even in poor network conditions, making it a competitive solution for use in modern power systems.

While the proposed CNN-based edge–cloud framework demonstrates excellent performance on a 33 kV distribution test system, certain limitations remain. First, the accuracy of the CNN is dependent on the diversity of training data; although multiple fault types, resistances, and load conditions were included, extending the dataset with field data will be critical for broader applicability. Second, the computational efficiency of edge devices imposes constraints on model complexity, requiring careful design of lightweight architectures. Third, scalability to larger distribution systems or meshed topologies may involve increased data synchronization demands between edge nodes and the cloud. Nonetheless, the modular nature of the framework enables deployment across multiple feeders, and edge devices can operate independently while periodically updating from the cloud. Future work will focus on testing the framework under higher renewable penetration, cyber attack resilience, and across larger IEEE benchmark networks to further validate scalability.

6 Conclusion

This study introduces a novel strategy for fault detection and localization in distribution networks by integrating convolutional neural networks (CNNs) with a hybrid edge–cloud computing framework. The proposed approach leverages the low-latency advantages of edge computing together with the predictive power of deep learning models, while overcoming the limitations of conventional impedance-based methods. Extensive simulations and case studies on a 33 kV distribution network validated the framework, demonstrating superior accuracy, reduced latency, and robustness under diverse operating conditions. The CNN-based method not only achieved faster response times but also correctly identified a wider range of fault types compared to traditional approaches. The edge–cloud architecture further minimized communication delays, enabling efficient decentralized processing. Results across different load scenarios and communication settings confirmed the adaptability and reliability of the framework for modern smart grids. Ultimately, combining CNNs with edge–cloud computing provides a powerful solution for real-time fault management, enhancing both the resilience and efficiency of distribution networks. Future work will extend this methodology to larger and more complex topologies and investigate the integration of additional data sources, such as weather and load forecasts, to improve predictive capability. These outcomes point toward the development of the next generation of intelligent, adaptive fault management systems in smart power grids.

Funding

This work was supported by the Guangxi Nature Science Foundation Project 2021GXNSFBA220023, Guangxi College Youth Basic Ability Improvement Project 2021KY0620 and 2021KY1019, 2021 Hechi University High-level Talents Research Start-up Project 2021GCC013.

Authors' contributions

YQ performed Data collection also FL carried out simulation and analysis. ML evaluate the first draft of the manuscript, editing and writing.

Acknowledgements

We wish to state that no individuals or organizations require acknowledgment for their contributions to this exploration.

Ethical approval

The article received ethical clearance from the IRB, guaranteeing adherence to ethical guidelines and protecting participants’ rights.

References

[1] J. Markard, “The next phase of the energy transition and its implications for research and policy,” *Nat Energy*, nature, vol. 3, no. 8, pp. 628–633, 2018.

[2] B. Kroposki *et al.*, “Achieving a 100% renewable grid: Operating electric power systems with extremely high levels of variable renewable energy,” *IEEE Power and energy magazine*, vol. 15, no. 2, pp. 61–73, 2017.

[3] Khaleghi A, Oshnoei S, Mirzajani S. Federated Learning Detection of Cyber attacks on Virtual Synchronous Machines Under Grid-Forming Control Using Physics-Informed LSTM. Fractal and Fractional. 2025; 9(9):569.

[4] P. D. Lund, J. Lindgren, J. Mikkola, and J. Salpakari, “Review of energy system flexibility measures to enable high levels of variable renewable electricity,” *Renewable and sustainable energy reviews*, Elsevier, vol. 45, pp. 785–807, 2015.

[5] R. W. Kenyon *et al.*, “Stability and control of power systems with high penetrations of inverter-based resources: An accessible review of current knowledge and open questions,” *Solar Energy*, Elsevier, vol. 210, pp. 149–168, 2020.

[6] Z. Ullah *et al.*, “Renewable energy resources penetration within smart grid: An overview,” in *2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, Istanbul, Turkey, IEEE, 2020, pp. 1–6.

[7] A. E. Saldaña-González, A. Sumper, M. Aragüés-Peñalba, and M. Smolnikar, “Advanced distribution measurement technologies and data applications for smart grids: A review,” *Energies (Basel)*, MDPI, vol. 13, no. 14, p. 3730, 2020.

[8] V. C. Gungor *et al.*, “A survey on smart grid potential applications and communication requirements,” *IEEE Trans Industr Inform*, IEEE, vol. 9, no. 1, pp. 28–42, 2012.

[9] C. Kalalas, L. Thrybom, and J. Alonso-Zarate, “Cellular communications for smart grid neighborhood area networks: A survey,” *IEEE access*, IEEE, vol. 4, pp. 1469–1493, 2016.

[10] S. M. Ali *et al.*, “Wide area smart grid architectural model and control: A survey,” *Renewable and Sustainable Energy Reviews*, Elsevier, vol. 64, pp. 311–328, 2016.

[11] A. Khaleghi and H. Karimipour, “Investigation of Detection Mechanisms Against False Data Injection Attacks Based on Machine Learning Approaches,” in Artificial Intelligence in the Operation and Control of Digitalized Power Systems, Springer, 2024, pp. 209–231.

[12] N. Maskey, S. Horsmanheimo, and L. Tuomimäki, “Analysis of latency for cellular networks for smart grid in suburban area,” in *IEEE PES Innovative Smart Grid Technologies, Europe*, Istanbul, Turkey, IEEE, 2014, pp. 1–4.

[13] K. Cao, Y. Liu, G. Meng, and Q. Sun, “An overview on edge computing research,” *IEEE access*, IEEE, vol. 8, pp. 85714–85728, 2020.

[14] C. Feng, Y. Wang, Q. Chen, Y. Ding, G. Strbac, and C. Kang, “Smart grid encounters edge computing: Opportunities and applications,” *Advances in Applied Energy*, Elsevier, vol. 1, p. 100006, 2021.

[15] N. Gana, N. F. Ab Aziz, Z. Ali, H. Hashim, and B. Yunus, “A comprehensive review of fault location methods for distribution power system,” 2017.

[16] J. Mora-Florez, J. Meléndez, and G. Carrillo-Caicedo, “Comparison of impedance-based fault location methods for power distribution systems,” *Electric power systems research*, Elsevier, vol. 78, no. 4, pp. 657–666, 2008.

[17] E. Personal, A. García, A. Parejo, D. F. Larios, F. Biscarri, and C. León, “A comparison of impedance-based fault location methods for power underground distribution systems,” *Energies (Basel)*, MDPI, vol. 9, no. 12, p. 1022, 2016.

[18] A. Khaleghi and H. Karimipour, "A Probabilistic-Based Approach for Detecting Simultaneous Load Redistribution Attacks Through Entropy Analysis and Deep Learning," in IEEE Transactions on Smart Grid, vol. 16, no. 2, pp. 1851-1861, March 2025, doi: 10.1109/TSG.2024.3524455.

[19] H. Nouri, C. Wang, and T. Davies, “An accurate fault location technique for distribution lines with tapped loads using wavelet transform,” in *2001 IEEE Porto Power Tech Proceedings (Cat. No. 01EX502)*, Porto, Portugal, IEEE, 2001, pp. 4-pp.

[20] R. J. Hamidi, H. Livani, and R. Rezaiesarlak, “Traveling-wave detection technique using short-time matrix pencil method,” *IEEE Transactions on Power Delivery*, IEEE, vol. 32, no. 6, pp. 2565–2574, 2017.

[21] A. N. Milioudis, G. T. Andreou, and D. P. Labridis, “Enhanced protection scheme for smart grids using power line communications techniques—Part II: Location of high impedance fault position,” *IEEE Trans Smart Grid*, IEEE, vol. 3, no. 4, pp. 1631–1640, 2012.

[22] S. Niu, X. Zhou, D. Zhou, Z. Yang, H. Liang, and H. Su, “Fault detection in power distribution networks based on comprehensive-YOLOv5,” *Sensors*, vol. 23, no. 14, p. 6410, Jul. 2023.

[23] A. P. Shafei, J. F. Silva, and J. Monteiro, “Convolutional neural network approach for fault detection and characterization in medium voltage distribution networks,” *e-Prime – Advances in Electrical Engineering, Electronics and Energy*, vol. 10, p. 100820, Dec. 2024.

[24] J. H. Gao, M. F. Guo, S. Lin, and D. Y. Chen, “Application of semantic segmentation in high-impedance fault diagnosis combined signal envelope and Hilbert marginal spectrum for resonant distribution networks,” *Expert Systems with Applications*, vol. 231, p. 120631, Nov. 2023.

[25] B. Hao, L. Yuxin, L. Jieyi, L. Hongwen, L. Yipeng, and L. Ruigui, “High impedance fault detection device based on edge artificial intelligence,” *Energy Reports*, vol. 9, pp. 546–550, Oct. 2023.

[26] C. Wang, L. Feng, S. Hou, G. Ren, and T. Lu, “A high-impedance fault detection method for active

distribution networks based on time–frequency–space domain fusion features and hybrid convolutional neural network," *Processes*, vol. 12, no. 12, p. 2712, Dec. 2024.

[27] O. Attallah, R. A. Ibrahim, and N. E. Zakzouk, "A lightweight deep learning framework for transformer fault diagnosis in smart grids using multiple scale CNN features," *Scientific Reports*, vol. 15, no. 1, p. 14505, Apr. 2025.