SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V13I4P103

Volume 13 Issue 4, 30-55, April 2026

Original Article

Enhanced Deep Learning Approach for Cross-Domain Detection of Faults and Foreign Objects on Power Transmission Line

M. Chinchu[1*], H. Vennila[2], G. S. Bibin[3]

[1*]*Department of ECE, Noorul Islam Centre for Higher Education, Tamil Nadu, India.*
[2]*Department of EEE, Noorul Islam Centre for Higher Education, Tamil Nadu, India.*
[3]*Kerala State Electricity Board Ltd, Kerala, India.*

[1]*Corresponding Author : chinchubibin13@gmail.com*

Received: 06 January 2026 Revised: 12 February 2026 Accepted: 15 March 2026 Published: 30 April 2026

Abstract *- The transmission power lines make up the bulk of the contemporary electrical infrastructure, and the constant operation of the power lines is critical to the well-being, stability of the economy, and the community. However, faults and foreign object penetration, such as contact with vegetation, conductor sagging, and flashover, are a significant threat, leading to power interruptions, equipment destruction, and safety hazards. Current fault detection methods are mostly based on single-modality analysis, manual inspection, or traditional deep learning models, which have low cross-domain awareness, low generalization, and low accuracy in complex environmental settings. To address these constraints, this work presents a deep learning method with improved cross-domain faults and foreign objects detection on power transmission lines. The proposed framework combines structured power system data and PMU fault images via a Unified Event-Centric Data Model (UECDM) and three significant novelties: accurate line segmentation with LineGuard-SegNet, optimal feature selection with RHO-MTS (ReliefF-guided Multi-Verse Optimizer with Tabu Search), and effective diagnosis with the TSTM-AttNet architecture with parallel multimodal branches and cross-modal attention fusion. The experimental results are superior with an accuracy of 98.8%, precision of 97.53, sensitivity of 98.62, specificity of 97.88, F1-score of 97.57, and outperform the current techniques on all evaluation measures.*

Keywords *- Power transmission line monitoring, LineGuard-SegNet, RHO–MTS feature selection, TSTM-AttNet, Predictive maintenance intelligence.*

1. Introduction

The identification of faults and foreign objects on power transmission lines is an important issue to guarantee the reliability, safety, and continuity of contemporary power systems [1]. The faults that can affect transmission lines are numerous, and they can be conductor breakage, line sagging, insulator flashover, overheating, short circuit, and mechanical failures. Moreover, overhead lines are often disrupted by foreign objects like vegetation intrusion, kites, plastic sheets, metallic debris, nests of birds, and construction materials, which cause transient faults, permanent outages, and extreme safety risks [2, 3]. Such incidents do not only interrupt power supply but also lead to economic damages, equipment destruction, wildfire threats, and safety issues among the population [4]. Thus, timely and precise identification of electrical faults as well as external intrusion is vital to grid stability and critical infrastructure protection [5].

As smart grids and large-scale power networks grow rapidly, manual inspection and rule-based monitoring will be inefficient, labour-intensive and prone to errors [6, 7]. Traditional methods like SCADA-based threshold monitoring, vibration sensors, and thermal cameras offer shallow contextual insight and are not always able to identify complex fault patterns and foreign object interactions [8]. More recently, deep learning-based object detection models have been created, such as DF-YOLO, TL-YOLO, MRA-YOLOv8, and Bolt-YOLO, which have been applied to power line inspection and fault detection. Although these models enhance the speed and accuracy of detection, they are mostly concerned with visual clues, and they work within one domain [9]. This means that they struggle with occlusions, low-contrast environments, complex backgrounds, and scenarios when electrical, environmental, and textual information plays a significant role [10]. In addition, the current methods tend to disregard geo-temporal dependencies, do not have effective feature selection methods, and cannot combine heterogeneous data sources, resulting in poor generalization and high false alarm rates [11, 12]. The increasing complexity of power networks, climatic variability, urbanization, and aging

infrastructure needs smarter and more resilient solutions to be more cross-domain diagnostic [13, 14]. The need to have frameworks capable of analyzing electrical signals, environmental conditions, textual fault reports, visual evidence, and spatial-temporal patterns in a joint manner to provide complete situational awareness is on the rise. The advance of multimodal learning, attention, and hybrid optimization opens up new opportunities to overcome the limitations of the vision-based or signal-based approaches [15, 16].

Although the deep learning-based methods of inspection have made significant progress, there are still several crucial limitations to address. Many current methods use uni-modal visual information and do not have the capability of properly incorporating heterogeneous data like electrical signals, environmental conditions, textual fault descriptions, and spatial-temporal patterns. Consequently, such models have a low cross-domain generalization, decreased resistance to complex environmental situations, and increased false alarms. Moreover, the literature lacks sufficient coverage of the necessity to have unified features representation and optimized feature selection among multiple data modalities to enable fault diagnosis in a modern power system with high accuracy and reliability.

The solution to these challenges is to have an integrated and smart framework that can analyze multimodal data sources together to provide fault and foreign object detection. A system like this must be able to maximize feature representation, have greater cross-domain adaptability, and be able to perform accurate diagnosis in differing environmental and operating conditions. Thus, the proposed work presents a new deep learning-based architecture that integrates multiple modalities of data, optimizes the selection of features, and applies attention-based diagnostics to obtain the best and most effective detection in power transmission systems.

In contrast to the current literature that uses single-modes or domain-based methodology, the proposed framework presents a multimodal learning plan (that cuts across domains) for fault and foreign object detection. Its novelty is based on the combination of heterogeneous data streams by a single event-based model, the creation of an efficient hybrid feature selection scheme, and the architectural creation of an attention-based multimodal diagnosis system. This combination allows stronger, more precise, and scalable detection in the complex power transmission environments.

1.1. Major Contributions

- To develop an enhanced deep learning framework for cross-domain detection of faults and foreign objects on power transmission lines by integrating electrical, environmental, textual, visual, geo-spatial, and temporal data.
- To introduce LineGuard-SegNet for precise transmission line segmentation, enabling accurate detection of thin conductors, sagging, and breakpoints under varying environmental conditions.
- To propose a robust feature selection strategy, RHO–MTS (RelieF-Guided Hybrid Optimization using Multi-Verse Optimizer and Tabu Search), to retain highly discriminative and domain-adaptive features while reducing redundancy.
- To design TSTM-AttNet, a multimodal diagnosis architecture with parallel Tabular Transformer, ST-GNN, TextCNN, and MobileViT branches, followed by cross-modal attention fusion for accurate fault and foreign object identification.

The remainder of this work is organized as follows: Section 2 reviews related work on the detection of faults and foreign objects on power transmission lines. Section 3 describes the proposed framework and its key components. Section 4 presents the experimental setup and performance evaluation. Finally, Section 5 concludes the work.

2. Literature Review

During the last few years, a lot of research has been dedicated to faults and foreign object detection in power transmission systems with the help of machine learning and deep learning. These works are primarily divided into vision-based detection, domain adaptation approaches, and hybrid learning. This part is the review of the most recent and relevant contributions to give a comprehensive background of the proposed work.

In 2023, Li et al. [17] suggested a deformable convolution-based YOLOv7-Tiny network, DF-YOLO, to detect foreign objects on transmission lines in the presence of complicated aerial backgrounds. The approach used DCN modules and SimAM attention to enhance the adaptability of features and recall, and Focal-DIoU loss to deal with the imbalance between positive and negative samples. A streamlined SPPCSPC_S-F module enhanced the inference rate. The results of the experiments demonstrated significant improvements in mAP, recall, and real-time performance over baseline YOLOv7-Tiny.

In 2024, Sun et al. [18] proposed a better model of foreign object detection on transmission lines surveyed with the help of UAV images based on the YOLOv8 model. The structure incorporated a Swin Transformer into the backbone to boost global feature extraction, and an Asymptotic Feature Pyramid Network (AFPN) to boost multi-scale feature fusion. Another Focal SIoU loss was a loss that optimized training on hard samples. Tests on a real-world dataset showed better accuracy, recall, and strong real-time performance in detecting various foreign objects.

In 2020, Zhang et al. [19] presented a multi-scale feature-enhanced domain adaptation model of cross-domain object

detection in power transmission line inspection. The method proposed a Multi-Scale Fusion Feature Alignment (MSFA) module to match representations between object scales and a Multi-Scale Consistency Regularization (MSCR) module to impose domain-invariant learning at every feature level. Experimental tests revealed that there were great improvements in cross-scene detection performance, especially when scale variations were observed in aerial inspection conditions.

In 2025, Wang et al. [20] suggested a cross-domain multilevel feature alignment R-CNN to detect defects on insulators using artificially generated and real images. The framework, which is based on Faster R-CNN, added instance-level adaptive alignment, image-level multiscale local alignment, and global feature alignment modules. To minimize distribution discrepancies, different gradient training strategies were used on the source and target domains. The results of the experiment revealed strong AP improvement over baseline models and high generalization in different datasets.

In 2024, Shao et al. [21] developed TL-YOLO, a modified version of YOLOv8 to detect foreign objects on power transmission lines with high accuracy. The model used full-dimensional dynamic convolution (ODConv) to enhance feature extraction, BiFPN, and multiscale attention to enhance feature fusion. A sparse GSCSP net minimized the computational cost, and ATSS equalized training samples. The experimental outcomes demonstrated a high level of accuracy, precision, recall, and inference speed in comparison with YOLOv8. In 2025, Stefenon et al. [22] suggested an Optimized Ensemble Deep Learning (OEDL-WBF) to detect faults in transmission line insulators. Weighted Boxes Fusion was employed to combine multiple YOLO-based detectors, and Tree-structured Parzen Estimator (TPE) was employed to optimize hyperparameters. Eigen-CAM was used as a framework for model interpretability. Experimental assessments showed better mAP performance than YOLOv8-YOLOv12 and emphasized the reliability and explainability of the model in the inspection of power grids.

In 2024, Yan et al. [23] proposed an improved YOLOv4-based fault detection model of transmission lines by substituting the backbone with EfficientNet and incorporating grouped convolution into the feature pyramid. Data augmentation increased robustness, whereas DIoU loss stabilized bounding box regression. The optimized model was much smaller in terms of parameter size and faster in detection without compromising on accuracy. The experimental findings were validated by improved real-time performance and applicability to resource-limited inspection settings. In 2025, Peng et al. [24] suggested YOLOv7-CWFD, a real-time transmission line bolt defect detector. The framework used CSDPAN to simplify the computational complexity, FFCAM attention to increase the sensitivity of the features, and DySample upsampling to minimize the loss of information. Also, EIoU and NWD loss functions enhanced the regression robustness. The custom and public dataset experiments showed a smaller model size, higher mAP, and high generalization.

In 2024, Zheng et al. [25] presented a transmission line foreign object detector, GEB-YOLO, which is lightweight and efficient. The framework combined GhostConv and YOLOv8 to minimize the computational cost and proposed a new EC2f mechanism to enhance the channel-wise feature correlation. A BiFPN module was used to improve multi-scale target handling. Experimental results showed enhanced accuracy and mAP, reduction in terms of parameters and FLOPs, and a balance between accuracy and efficiency.

In 2021, Liu et al. [26] suggested CSPD-YOLO, a Cross Stage Partial Dense YOLO architecture to detect insulator faults in complicated aerial images. The framework, which was based on YOLOv3, reused features by using dense connections and CSP architecture, and a better feature pyramid and loss function to improve accuracy in detection. Experiments conducted on a newly constructed dataset were more precise on average than those of YOLOv3 and YOLOv4, which were more robust in the scenario of multi-fault insulator detection. The comparison of the related work is presented in Table 1.

Table 1. Comparison of the Related Works

Authors, Year	Study	Proposed Framework	Key Limitations
In 2023, Li et al. [17]	DF-YOLO	YOLOv7-Tiny enhanced with deformable convolution (DCN), SimAM attention, Focal-DIoU loss, and SPPCSPC_S-F module for foreign object detection	• Designed for a single-domain aerial dataset • Limited evaluation on cross-domain generalization
In 2024, Sun et al. [18]	YOLOv8 + Swin Transformer	YOLOv8 integrated with Swin Transformer, AFPN, and Focal SIoU loss for UAV-based foreign object detection	• Increased model complexity due to transformer integration • Performance not validated on heterogeneous domains
In 2020, Zhang et al.	Multi-Scale Domain Adaptive	Domain adaptive object detection with MSFA and MSCR for cross-scene transmission line	• Focused on domain adaptation only, not lightweight deployment •

[19]	Detection	inspection	Did not address real-time inference constraints
In 2025, Wang et al. [20]	CMFAA-R-CNN	Faster R-CNN with instance-level, local, and global feature alignment for cross-domain insulator defect detection	• Two-stage architecture with high computational cost • Not suitable for real-time UAV-based inspection
In 2024, Shao et al. [21]	TL-YOLO	YOLOv8 with ODConv backbone, BiFPN + MSA fusion, GSCSP, and ATSS sample selection	• Model still domain-specific to transmission line imagery • Limited robustness under unseen environmental conditions
In 2025, Stefenon et al. [22]	OEDL-WBF	Ensemble of YOLO models with weighted boxes fusion, TPE-based hyperparameter tuning, and Eigen-CAM explainability	• Ensemble framework increases inference overhead • Not optimized for edge or UAV deployment
In 2024, Yan et al. [23]	EfficientNet-YOLOv4	YOLOv4 optimized using EfficientNet backbone, grouped convolution, DIoU loss, and data augmentation	• Performance constrained by older YOLOv4 architecture • Limited capability for small or thin object detection
In 2025, Peng et al. [24]	YOLOv7-CWFD	YOLOv7 with CSDPAN, FFCAM attention, DySample upsampling, and EIoU + NWD losses for bolt defect detection	• Specialized for bolt defects only • Does not generalize to multiple fault or foreign object categories
In 2024, Zheng et al. [25]	GEB-YOLO	Lightweight YOLOv8 with GhostConv, EC2f module, and BiFPN for efficient foreign object detection	• Reduced capacity affects detection in highly complex scenes • No explicit cross-domain learning mechanism
In 2021, Liu et al. [26]	CSPD-YOLO	YOLOv3-based CSP Dense network with feature pyramid and improved loss for insulator fault detection	• Based on older YOLO versions • Limited scalability to multi-domain and multi-object detection

As can be seen in the discussion above, the current techniques are mostly single-domain in visual analysis and may be restricted in their ability to deal with heterogeneous sources of data and cross-domain variability. Although a few of them involve attention mechanisms or domain adaptation, they lack a common structure that synthesizes electrical, environmental, textual, visual, and geo-temporal information. Conversely, the proposed work proposes an integrated multimodal architecture that integrates the data modeling, feature selection optimization, and multimodal attention-based diagnosis. This allows it to generalize better, minimize false alarms, and identify faults more reliably than current methods. Overall, the reviewed studies demonstrate notable progress in improving detection accuracy and efficiency.

However, most existing works are limited to single-modal data, primarily relying on visual information, and do not fully exploit the complementary nature of electrical, environmental, textual, and spatial-temporal data. In addition, limited attention has been given to unified data modeling and optimized feature selection across heterogeneous domains. These limitations highlight the need for a comprehensive multimodal framework, which is addressed in the proposed work.

3. Proposed Methodology

The suggested framework uses a more advanced deep learning pipeline to detect faults and foreign objects on power transmission lines across domains. It combines structured power system data and PMU fault images with a Unified Event-Centric Data Model (UECDM) to coordinate electrical, environmental, textual, visual, geo-spatial, and temporal data. Pre-processing involves Isolation Forest-based outlier detection, temporal KNN imputation, normalization, WordPiece tokenization, and novel image enhancement by LineGuard-SegNet segmentation. Statistical, time-frequency, stability indices, BERT-Tiny embeddings, and fault-aware visual descriptors are used to extract discriminative features. RHO-MTS is an optimized feature selection that is a combination of ReliefF, Multi-Verse Optimizer, and Tabu Search.

The diagnosis is done through the TSTM-AttNet architecture using parallel Tabular Transformer, ST-GNN, TextCNN, and MobileViT branches and cross-modal attention fusion. A spatial regression network allows fault localization, whereas a PPO-based reinforcement learning engine can offer intelligent maintenance suggestions and priority scheduling. Figure 1 represents the architecture of the proposed framework.

3.1. Data Collection

The work uses a multimodal data collection approach to record the various operational, environmental, spatial, textual, and visual attributes of power transmission line faults and foreign object intrusions.

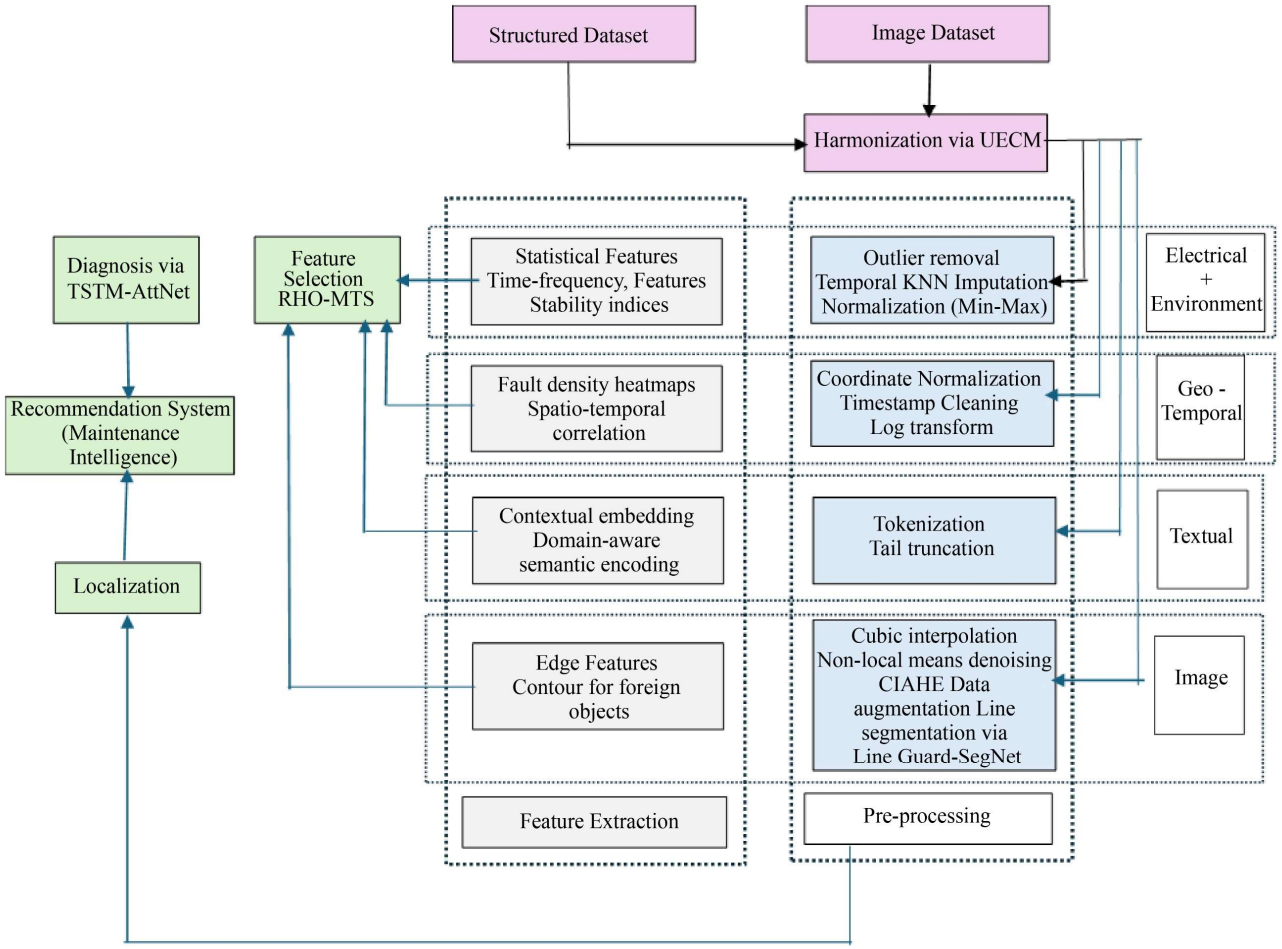

Fig. 1 Overall Architecture of the Proposed Framework

3.1.1. Structured Fault Dataset

The Power System Faults Dataset on Kaggle is used to gather structured operational data. This data set contains detailed records of disturbances in power systems, including electrical quantities like voltage, current, and power load, environmental factors like temperature, wind speed, and weather conditions, spatial data like latitude and longitude, maintenance data like component health and maintenance condition, time-related data like fault duration and downtime, and a textual description of each fault, specifying the nature of each event. These crude numerical, environmental, spatial, temporal, and textual data are submitted to their corresponding preprocessing divisions to be cleaned, normalized, and semantically encoded.

3.1.2. Image Dataset

The Smart Grid Phasor Measurement Unit Fault Images dataset consists of visual information, i.e., inspection images of a vast number of fault cases, i.e., line sagging, broken conductors, flashover, vegetation intrusion, and foreign object interference. The primary visual tool for finding structural anomalies and external intrusion on transmission lines is these pictures. The raw images are directly fed into the image preprocessing branch to be enhanced, segmented, and augmented.

3.1.3. Unified Event-Centric Harmonization

To ensure coherent multimodal learning, a Unified Event-Centric Data Model (UECDM) is constructed, where each fault instance is represented as Equation (1):

$$UECDM_i = \{X_i^{num}, X_i^{env}, X_i^{text}, X_i^{img}, X_i^{geo}, X_i^{time}\} \quad (1)$$

With X^{num} denoting electrical signals, X^{env} representing weather and temperature attributes, X^{text} indicating fault description embeddings, X^{img} referring to visual features, X^{geo} capturing geographical coordinates, and X^{time} denoting fault duration and downtime. The modalities are synchronized by matching the timestamps, location-based clustering, and semantic linking of fault-ID to guarantee that all heterogeneous data streams of the same physical event are synchronized appropriately. The multimodal events are then harmonized and sent to their respective preprocessing pipelines to be refined by their respective modalities.

3.2. Pre-processing

3.2.1. Numeric (X^{num}) and Environmental Data (X^{env}) Pre-processing

Raw electrical and environmental data streams of the structured dataset will include measurement noise, missing values, and scale inconsistencies that negatively impact downstream learning. Thus, a sequential preprocessing pipeline is used to guarantee data integrity, temporal consistency, and numerical stability prior to feature extraction. Let the raw numeric–environmental input be represented as,

$$NEX_0 = \{X^{num}, X^{env}\} \in \mathbb{R}^{TL \times \mathbb{A}},$$

Where TL denotes the temporal length and mathbb{A} represents the total number of electrical and environmental attributes.

Outlier Detection and Removal using Isolation Forest

To begin with, isolation Forest (iForest) is used to detect and eliminate anomalous samples due to sensor failures, transient spikes, or corrupted measurements. Isolation Forest isolates anomalies by randomly dividing the feature space and counting the mean length of the path to isolate a point. Anomalous points have short path lengths when compared to normal cases. Given an instance $ne_i \in NEX_0$, its anomaly score is computed as Equation (2):

$$s(ne_i) = 2^{-\frac{E\left(h(ne_i)\right)}{c(n)}} \quad (2)$$

Where, $E\left(h(ne_i)\right)$ is the expected path length and $c(n)$ is the normalization factor. Samples with $s(ne_i) > \tau$ are classified as outliers and removed. The cleaned output is given by Equation (3):

$$NEX_1 = NEX_0 \backslash \{ne_i | s(ne_i) > \tau\} \quad (3)$$

This step suppresses extreme voltage spikes, unrealistic current surges, and abnormal environmental readings.

Missing Value Imputation using Temporal KNN

The data NEX_1 which has been filtered with outliers, still has a blank in the records because of the loss of communication or sensor failure. To reestablish the temporal continuity, Temporal K-Nearest Neighbors (T-KNN) imputation is used. In contrast to conventional KNN, T-KNN takes into account the similarity of features as well as their temporal proximity. To find the imputed value $\widehat{ne}_t$ widehat of a missing value at time t, the value to be imputed is calculated as Equation (4):

$$\widehat{ne}_t = \frac{1}{\mathbb{K}} \sum_{j \in \mathcal{N}_{\mathbb{K}}(t)} ne_j \quad (4)$$

Where, $\mathcal{N}_{\mathbb{K}}(t)$ refers to a set of $\mathbb{K}$ nearest neighbors in time and feature similarity. The imputed output is $NEX_2 = TKNN(NEX_1)$. This guarantees the ability to reconstruct voltage, current, temperature, and wind-speed sequences without causing artificial discontinuities.

Min–Max Normalization

Minimum-maximum normalization is used for all the dimensions of the feature in NEX_2 to remove the scale dominance of heterogeneous attributes. This transforms all values into the range [0,1], enabling stable gradient propagation in deep networks. For each feature $\mathbb{f}$, normalization is performed as Equation (5):

$$x'_{\mathbb{f}} = \frac{x_{\mathbb{f}} - \min(x_{\mathbb{f}})}{\max(x_{\mathbb{f}}) - \min(x_{\mathbb{f}})} \quad (5)$$

The final normalized output is $NEX_3 = Norm(NEX_2)$. After sequential outlier removal, temporal imputation, and normalization, the refined numeric–environmental representation is obtained as $X_{clean}^{num,env} = NEX_3$.

3.2.2. Geographical (X^{geo}) and Temporal (X^{time}) Pre-processing

The raw geographical and temporal characteristics of every fault event tend to be inconsistent in scale, drift in coordinates, have invalid timestamps, and have biased distributions. A sequential processing pipeline is used on the geographical coordinates and fault duration data to achieve spatial consistency and temporal accuracy before extracting the features.

Let the raw inputs be defined as Equation (6):

$$X_0^{geo} = \{lat, lon\},\ X_0^{time} = \{t_{start}, t_{end}, \Delta t\} \quad (6)$$

Where lat and lon denote latitude and longitude, and Δt represents fault duration.

Coordinate Normalization via Min–Max Scaling

The geographical coordinates are initially normalized to remove the scale dominance and provide numerical stability in the fusion process. Latitude and longitude values are in different number ranges, and therefore, Min-Max scaling is used separately on each dimension of the coordinates.

This transformation maps all spatial coordinates into the unified range [0,1], preserving relative spatial distances while preventing large-magnitude coordinate values from biasing the learning process. The normalized spatial output is $X_1^{geo} = \{lat', lon'\}$.

Timestamp Cleaning via Rule-Based Temporal Validation

The raw time stream X_0^{time} time has inconsistencies like negative time durations, overlapping time durations, missing start-end pairs, or unrealistically long fault periods as a result of logging errors. In order to solve this, temporal validation by rules is employed.

The validation rules include $t_{end} \geq t_{start}$, $\Delta t = t_{end} - t_{start}$, and $\Delta t \in [\delta_{min}, \delta_{max}]$. Invalid records violating these constraints are either corrected (if partial information is available) or removed. This produces a temporally consistent sequence as $X_1^{time} = Validate(X_0^{time})$. This step ensures that all fault durations reflect physically meaningful temporal behavior.

Logarithmic Scaling with Offset for Duration Normalization

The values of fault duration are usually heavy-tailed in nature, with a small number of long-duration faults dominating the scale. Logarithmic scaling with offset is used to reduce skewness and enhance numerical conditioning. For each validated duration $\Delta t \in X_1^{time}$, the transformed value is defined as Equation (7):

$$\Delta t' = \log(\Delta t + \epsilon) \tag{7}$$

Where $\epsilon > 0$ is a small offset to avoid undefined values at zero. The log-scaled temporal output is $X_2^{time} = \{\Delta t'\}$. This transformation reduces the extreme values and maintains the temporal order, which enables stable learning in downstream models.

After sequential coordinate normalization and temporal cleaning–scaling, two refined outputs are obtained, which are represented as $X_{clean}^{geo} = X_1^{geo} = \{lat', lon'\}$ and $X_{clean}^{time} = X_2^{time} = \{\Delta t'\}$. X_{clean}^{geo} and X_{clean}^{time}

3.2.3. Textual Fault Description Pre-processing (X^{text})

The raw fault description texts that accompany each event have unstructured variable length and domain-specific linguistic patterns that describe the cause of faults and operational anomalies. A sequential text preprocessing pipeline is used to allow uniform semantic encoding and effective downstream representation learning, which includes subword tokenization and controlled sequence truncation. Let the raw textual input be defined as Equation (8):

$$X_0^{text} = \{sw_1, sw_2, \dots, sw_N\} \tag{8}$$

Where, sw_i denotes the natural language fault description corresponding to the i^{th} event.

Subword Tokenization using WordPiece Tokenizer

Firstly, all the fault descriptions are broken down into subword units through WordPiece tokenization. WordPiece creates a dictionary of fixed character sequences that are common and models rare or unseen words as a composition of subword tokens. This enables the sound management of technical terms such as flashover, overheating, conductor snap, as well as vegetation contact, without excessive out-of-vocabulary issues. Given a sentence sw_i, tokenization produces a sequence $To_i = \{to_{i1}, to_{i2}, \dots, to_{iL_i}\}$, where t_{ij} denotes the j^{th} subword token and L_i is the token length of the sequence. The tokenized corpus is represented as $X_1^{text} = \{To_1, To_2, \dots, To_N\}$. This step converts unstructured text into discrete symbolic units suitable for embedding.

Fixed-Length Tail Truncation

Direct batching results in inconsistent tensor dimensions because the description of faults can be of different lengths. In order to impose uniformity, tokenizer-level fixed-length right-side (tail) truncation is used. This method maintains the most informative leading context (usually containing the fault type and cause) and removes the redundant trailing tokens.

For each token sequence To_i, truncation is performed as Equation (9):

$$To_i' = \begin{cases} To_i[1:sL_{max}], & if\, L_i > sL_{max}, \\ To_i, & otherwise. \end{cases} \tag{9}$$

Where, sL_{max} is the predefined maximum sequence length. The truncated output is $X_2^{text} = \{To_1', To_2', \dots, To_N'\}$. This ensures fixed-dimensional input while retaining the core semantic content. After sequential WordPiece tokenization and fixed-length truncation, the refined textual representation is obtained as $X_{clean}^{text} = X_2^{text}$.

3.2.4. Image Data Pre-processing (X^{img})

The raw inspection images obtained with the PMU fault image dataset have different resolutions, noise, illumination, weather, motion blur, and sensor quality. A sequential image processing pipeline is used to standardize visual inputs and improve discriminative patterns before line segmentation and feature extraction. Let the raw image input be represented as $X_0^{img} = \{I_1, I_2, \dots, I_N\}$, where $I_i \in \mathbb{R}^{H \times W \times C}$ denotes the i^{th} RGB image with height H, width W, and channels C.

Spatial Resizing via Bicubic Interpolation

First, every image is downsized to a constant spatial resolution ($H_0 \times W_0$) by bicubic interpolation. Bicubic interpolation calculates each pixel of the output image as a weighted mean of the closest 16 pixels in the input image, which yields smoother and more visually coherent results than nearest-neighbor or bilinear interpolation. Formally, the resized image is obtained as $I_i^{(1)} = \mathcal{R}_{bicubic}(I_i)$. The step ensures standardization of the input size, which allows processing in batches and the mapping of receptive fields consistently in the downstream convolutional networks.

Noise Suppression using Non-Local Means Denoising

The resized images $I_i^{(1)}$ still have sensor noise, compression artifacts, and environmental distortions. Non-Local Means (NLM) denoising is used to reduce noise without any loss of structural information. In contrast to local filters, NLM calculates the denoised value of a pixel by averaging the pixels in the image with weights based on patch similarity. For a pixel px, the denoised value is defined as Equation (10):

$$I_i^{(2)}(px) = \sum_{qx \in \Omega} w(px, qx) \cdot I_i^{(1)}(qx) \quad (10)$$

Where Ω signifies the search window and $w(px, qx)$ is a similarity-based weight satisfying $\sum_{qx} w(px, qx) = 1$. This is done by eliminating stochastic noise and retaining edges and fine line structures that are important in transmission line analysis.

Contrast Enhancement using CLAHE

To address uneven illumination, haze, and low-contrast conditions, Contrast Limited Adaptive Histogram Equalization (CLAHE) is applied to $I_i^{(2)}$. CLAHE operates on small contextual regions (tiles) and enhances local contrast while limiting amplification to avoid noise over-enhancement. The enhanced image is obtained as $I_i^{(3)} = \mathcal{C}_{CLAHE}(I_i^{(2)})$. This step significantly improves the visibility of thin conductors, insulators, and foreign objects under adverse lighting conditions. Figure 2 depicts the image pre-processing outcomes.

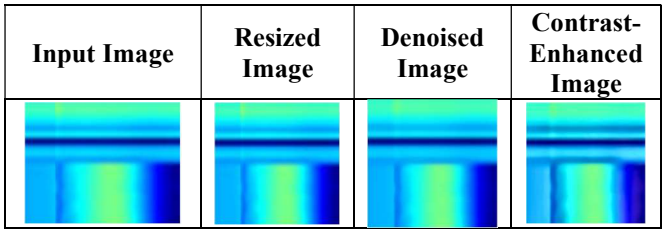

Fig. 2 Image Pre-processing Outcome

Domain-Driven Data Augmentation

To improve cross-domain generalization and robustness against environmental variability, controlled data augmentation is applied to $I_i^{(3)}$. The augmentation operators include:

- Rotation: $\theta \in [-\alpha, +\alpha]$ to simulate camera tilt
- Gaussian Blur: to emulate motion blur and defocus
- Fog Simulation: to replicate haze and atmospheric scattering
- Brightness Adjustment: to model illumination changes

Each augmented sample is generated as $I_i^{(4)} = \mathcal{AU}(I_i^{(3)})$, where $\mathcal{AU}$ denotes a composite augmentation operator. This step expands the data distribution and reduces domain bias caused by weather and sensor conditions. The data augmentation is graphically represented in Figure 3.

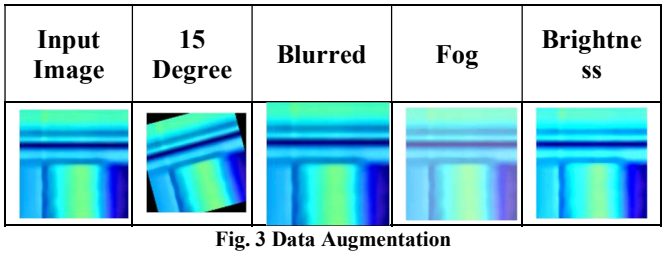

Fig. 3 Data Augmentation

After sequential resizing, denoising, contrast enhancement, and augmentation, the refined visual representation is obtained as $X_{clean}^{img} = \{I_1^{(4)}, I_2^{(4)}, \dots, I_N^{(4)}\}$. These enhanced images are then forwarded as input to the Transmission Line Segmentation Module for structural isolation prior to visual feature extraction.

3.2.5. Transmission Line segmentation using LineGuard-SegNet

LineGuard-SegNet is a structure-sensitive segmentation model that is lightweight and is designed to detect transmission lines, foreign objects, and faults in cross-domain conditions with high accuracy. It defeats thin-line geometry, background clutter, scale variation, and domain shift through a combination of Learnable Line-aware *Extraction* Edge Module (LLEEM), Domain-Aware Coordinate Refinement (DA-CFRM), and Cross-Scale Line-Guided Fusion (CS-LGF) in a dual-path GhostConv backbone. The network generates accurate pixel-based segmentation and fault localization masks to be used in downstream diagnosis. Figure 4 represents the general architecture of the proposed LineGuard-SegNet.

Learnable Line-Aware Edge Extraction Module (LLEEM)

The refined visual representations are then received as $X_{clean}^{img} = \{I_1^{(4)}, I_2^{(4)}, \dots, I_N^{(4)}\}$ after sequential resizing, denoising, contrast enhancement, and augmentation, and sent to the Transmission Line Segmentation module. The initial step is the Learnable Line-Aware Edge Extraction Module (LLEEM), which explicitly models thin conductor geometry, sagging patterns, and breakpoints in a domain-adaptive and fault-sensitive way.

Power transmission lines are very thin, high aspect ratio, and highly directional, whereas faults like sagging, breakage, and flashover cause localized directional discontinuities. Classical edge operators with fixed kernels are insufficient due to:

- lack of adaptability to domain shifts (weather, illumination, background),
- poor sensitivity to sub-pixel thin structures,
- high noise susceptibility under fog, rain, and glare.

Thus, LLEEM uses learnable, directional, depthwise Ghost convolutional filters instead of fixed edge filters, which change their morphology to data distribution and fault morphology during training.

Directional Depthwise GhostConv Filtering

Given an input image $I^{(4)} \in \mathbb{R}^{H \times W \times 3}$, LLEEM applies a bank of direction-specific depthwise GhostConv kernels oriented at $\Theta = \{0°, 45°, 90°, 135°\}$ to capture horizontal, diagonal, vertical, and anti-diagonal line structures. For each direction $\theta \in \Theta$, the response map is computed as Equation (11):

$$R_\theta = GhostConv_\theta^{dw}(I^{(4)}) \quad (11)$$

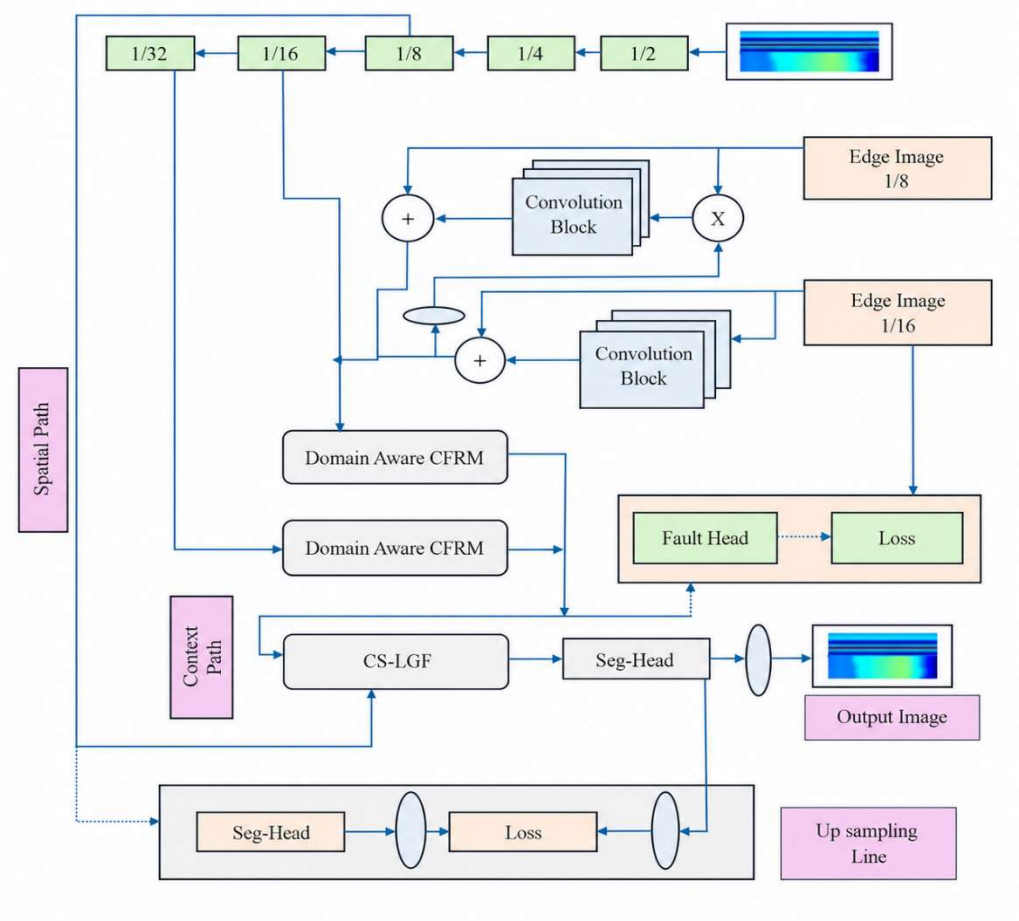

Fig. 4 General Architecture of the proposed LineGuard-SegNet

In which, $GhostConv_{\theta}^{dw}$ refers to depthwise separable Ghost convolution with directional kernel initialization, depthwise operation is sensitive to channelwise structure, and Ghost feature generation is less redundant and more expressive. This produces a set of multi-directional line response maps as $\mathcal{RM} = \{R_{0^\circ}, R_{45^\circ}, R_{90^\circ}, R_{135^\circ}\}$.

Multi-Directional Line Response Aggregation

The response maps are fused by channel-wise concatenation and pointwise convolution to consolidate directional evidence and increase thin-structure continuity, which is defined as Equation (12):

$$R_f = \sigma s(\mathbb{W}_{1\times1} * R_{cat} + b) \quad (12)$$

Where, $R_{cat} = Concat(R_{0^\circ}, R_{45^\circ}, R_{90^\circ}, R_{135^\circ})$, $\mathbb{W}_{1\times1}$ signifies a learnable pointwise convolution kernel, $\sigma s(\cdot)$ represents the sigmoid activation, $R_f \in \mathbb{R}^{H\times W}$ represents the fused directional line activation.

With this formulation, the network can learn anisotropic line patterns and is sensitive to sagging curvature and sharp discontinuities due to breaks.

Edge Confidence Map Formulation

Instead of producing a hard binary edge mask, LLEEM outputs a continuous Edge Confidence Map that encodes the probability of line presence and structural integrity as $E_c = R_f$, $E_c \in \mathbb{R}^{H\times W}$.

Each pixel value $E_c(i,j) \in [0,1]$ represents the confidence that location (i,j) belongs to a conductor edge or fault-induced boundary. This soft representation maintains uncertainty information, which is essential in the case of occlusion, vegetation overlap, and low-contrast conditions.

LLEEM is domain-adaptive in the sense that all directional filters are entirely learnable and trained end-to-end, enabling the module to adapt to weather conditions (fog, rain, haze), changes in illumination, and camera noise and motion blur.

Therefore, LLEEM directly captures line continuity, orientation, and structural integrity, and is therefore very well adapted to transmission line inspection in cross-domain conditions. The output of this step is the edge confidence representation $E_c \in \mathbb{R}^{H\times W}$ that is sent to the dual-path feature encoding in geometry semantic learning, and Cross-Scale

Line-Guided Feature Fusion (CS-LGF) in guiding the attention weighting and fusion dynamics.

This makes all the further segmentation and localization steps edge-conscious, structure-conscious, and fault-conscious.

Dual-Path Feature Encoding with Lightweight GhostConv Backbone

Following edge-aware enhancement via LLEEM, the refined visual representation X_{clean}^{img} and its corresponding edge confidence map E_c are forwarded to a Dual-Path Feature Encoding module built on a lightweight GhostConv backbone. This design explicitly decouples fine-grained geometric preservation from high-level semantic context modeling, enabling accurate segmentation of thin conductors while capturing complex fault and background semantics.

Transmission line inspection imposes two conflicting requirements:

1. Preservation of ultra-thin structures (conductors, broken strands, sagging curves) demands high-resolution, low-stride feature extraction.
2. Modeling of long-range contextual dependencies (vegetation intrusion, tower proximity, flashover regions) requires deep receptive fields and semantic abstraction.

To deal with this, the architecture uses a dual-path approach that comprises a Spatial Path, which is used to preserve geometry, and a Context Path, which is used to understand the semantics. The backbone is constructed with Ghost convolutional blocks, producing intrinsic feature maps and cheap ghost features to eliminate redundancy, resulting in lower computational cost and memory footprint, and therefore the model can be used in real-time UAV and edge deployment.

Spatial Path

High-Resolution Geometry Preservation

The spatial path operates at high resolution to retain fine structural details of thin conductors and edges. Given the preprocessed image $I^{(4)}$and edge map E_c, the spatial path applies a sequence of shallow GhostConv blocks, which is defined as Equation (13):

$$F_s = \phi_s\left(I^{(4)} \oplus E_c\right) \tag{13}$$

Where $\oplus$ denotes channel-wise concatenation, $\phi_s(\cdot)$ represents a stack of lightweight GhostConv layers with a small stride (typically 1 or 2), $F_s \in \mathbb{R}^{H\times W\times C_s}$ is the spatial feature map. Each GhostConv block is defined as Equation (14):

$$GhostConv(X) = Concat\left(X * W_{primary}, g\left(X * W_{primary}\right)\right) \tag{14}$$

Where, $W_{primary}$ are standard convolution kernels, $g(\cdot)$ denotes cheap linear transformations (depthwise convolution), and the design minimizes redundant channel computation while preserving representational power. This direction guarantees continuity of thin wires, proper localization of breaks and sagging, and high resistance to low-contrast situations.

Context Path

Deep Semantic Encoding

The context path focuses on extracting high-level semantic features using a deeper GhostConv encoder with progressive downsampling. It captures global context related to vegetation overlap, background clutter, fault regions, and flashover patterns. Formally $F_c = \phi_c\left(I^{(4)}\right)$, where $\phi_c(\cdot)$ denotes a deep stack of GhostConv layers with increasing receptive field and stride, $F_c \in \mathbb{R}^{\frac{H}{r}\times\frac{W}{r}\times C_c}$ (with $r \in \{8,16\}$) represents context features.

The progressive downsampling is enabled when $RF(F_c) \gg RF(F_s)$, where RF denotes the receptive field, ensuring long-range dependency modeling.

Efficiency and Real-Time Suitability

GhostConv significantly reduces computation by generating $\mathbb{m}$ feature maps using $\mathbb{m} = \mathbb{mm} + \mathbb{s}$, where $\mathbb{mm}$ intrinsic features are computed via standard convolution and $\mathbb{s}$ ghost features are generated through cheap operations. This yields $FLOP_{\mathbb{s}_{GhostConv}} \ll FLOP_{\mathbb{s}_{StandardConv}}$. At the end of this stage, the module outputs two complementary feature maps, which are $F_s \in \mathbb{R}^{H\times W\times C_s}$(Spatial Features) and $F_c \in \mathbb{R}^{\frac{H}{r}\times\frac{W}{r}\times C_c}$(Context Features).

Domain-Aware Coordinate Feature Refinement Module (DA-CFRM)

After the dual-path encoding, the spatial features F_s and context features F_c are sent to the Domain-Aware Coordinate Feature Refinement Module (DA-CFRM). The module is specifically created to enhance feature representations with directional, continuity-aware, and fault-sensitive attention that is important to accurately identify sagging conductors, discontinuities, and breakpoints in different environmental and domain conditions.

Directional Coordinate Pooling

Given spatial and context feature maps $F_s \in \mathbb{R}^{H\times W\times C_s}$ and $F_c \in \mathbb{R}^{H'\times W'\times C_c}$ both are first aligned to a common resolution via bilinear upsampling as Equation (15):

$$\tilde{F}_c = \mathbb{U}(F_c),\ \tilde{F}_c \in \mathbb{R}^{H\times W\times C_c} \tag{15}$$

The aligned features are concatenated as $FF = [F_s \parallel \tilde{F}_c] \in \mathbb{R}^{H\times W\times CC}$ where $CC = C_s + C_c$. Directional coordinate pooling is then applied along horizontal and vertical axes as

Equation (16):

$$P_h(c,y) = \frac{1}{W}\sum_{x=1}^{W} FF(c,y,x),$$

$$P_v(c,x) = \frac{1}{H}\sum_{y=1}^{H} FF(c,y,x) \quad (16)$$

Producing $P_h \in \mathbb{R}^{CC\times H\times 1}$, $P_v \in \mathbb{R}^{CC\times 1\times W}$. This formulation preserves explicit directional dependency, enabling the network to model horizontal sagging, diagonal drift, and vertical discontinuities.

Continuity-Biased Anisotropic Attention

To enforce line continuity and penalize abrupt spatial breaks, the pooled features are passed through a shared transformation, which is defined as Equation (17):

$$\mathbb{Z} = ReLU\big(BN(WC_1 * [P_h \parallel P_v])\big) \quad (17)$$

Where, WC_1 denotes a 1×1convolution, BN signifies batch normalization. The transformed features are then split into horizontal and vertical attention tensors as $Z_h, Z_v = Split(\mathbb{Z})$, and projected to attention weights using sigmoid activation as Equation (18):

$$Att_h = \sigma s(W_h * Z_h), \;\; Att_v = \sigma s(W_v * Z_v) \quad (18)$$

Where, $A_h \in \mathbb{R}^{CC\times H\times 1}$ and $A_v \in \mathbb{R}^{CC\times 1\times W}$. To introduce fault sensitivity and break penalization, a continuity modulation term is applied as Equation (19):

$$Att_h^* = Att_h \odot (1-\nabla_h),$$

$$Att_v^* = Att_v \odot (1-\nabla_v) \quad (19)$$

Where, ∇_h, ∇_v denote directional gradient magnitudes, high gradients indicate abrupt changes (potential breaks), and $\odot$ denotes element-wise multiplication. The final refined feature map is obtained by applying the anisotropic attention weights, which are defined as Equation (20):

$$F_r = FF \odot Att_h^* \odot Att_v^* \quad (20)$$

Equivalently $F_r = DACFRM(F_s, F_c)$, here $F_r \in \mathbb{R}^{H\times W\times CC}$ is the directionally refined, continuity-aware, fault-sensitive feature representation.

DA-CFRM mitigates domain shift through encoding of structural priors (line continuity, directionality), domain-specific noise suppression (foliage texture, lighting artifacts), and amplification of fault-induced spatial irregularities across environments. This renders the refinement process domain-conscious, which guarantees consistent performance between UAV and PMU imagery, varying weather, and heterogeneous backgrounds.

Cross-Scale Line-Guided Feature Fusion (CS-LGF) and Fault-Aware Output Heads

After domain-aware refinement, the refined feature map F_r and edge confidence map E_c are input to the Cross-Scale Line-Guided Feature Fusion (CS-LGF) module and then to the Fault-Aware Segmentation and Localization Heads. This step combines geometry, semantics, and edge priors together and generates explicit fault-aware outputs to be used in downstream diagnosis.

Cross-Scale Line-Guided Feature Fusion (CS-LGF)

Traditional feature fusion methods (concatenation, summation, average pooling) are based on the assumption that the distribution of features is homogeneous across scales and modalities. However, in transmission line inspection:

- spatial features encode thin geometry and continuity,
- context features encode fault semantics and background, and
- edge features encode structural boundaries and discontinuities.

Naïve fusion leads to suppression of thin conductors, dilution of fault cues, and dominance of background textures.

CS-LGF is therefore designed to perform edge-conditioned, line-aware, and cross-scale adaptive fusion, explicitly guided by the edge confidence map E_c.

Edge-Conditioned Attention Formulation

Given a refined feature map $F_r \in \mathbb{R}^{H\times W\times CC}$ and edge confidence map $E_c \in \mathbb{R}^{H\times W}$. The edge map is first expanded and normalized as Equation (21):

$$\tilde{E}_c = Norm\big(Expand(E_c)\big) \in \mathbb{R}^{H\times W\times 1} \quad (21)$$

An edge-conditioned attention tensor is computed as Equation (22):

$$A_e = \sigma s\big(W_e * \tilde{E}_c\big) \quad (22)$$

Where, W_e denotes a 1×1convolution.

This generates $A_e \in \mathbb{R}^{H\times W\times 1}$ which assigns higher weights to pixels with strong line or fault edges.

Cross-Scale Line-Guided Fusion

To integrate multi-scale information, a pyramid of refined features is constructed as $\{F_r^{(1)}, F_r^{(2)}, F_r^{(3)}\} = \{F_r, D_2(F_r), D_4(F_r)\}$, where $D_k(\cdot)$ denotes downsampling by factor k. Each scale is re-weighted using edge-conditioned attention as Equation (23):

$$\hat{F}_r^{(i)} = F_r^{(i)} \odot Up_i(A_e) \quad (23)$$

Where, $Up_i(\cdot)$ upsamples A_e to match the spatial size of $F_r^{(i)}$. The line-guided fused representation is then obtained as Equation (24):

$$F_f = \sum_{i=1}^{3} \alpha_i \cdot \hat{F}_r^{(i)} \quad (24)$$

With $\alpha_i = \frac{\exp(\gamma_i)}{\sum_{j=1}^{3} \exp(\gamma_j)}$, where, γ_i are learnable scale weights. Thus, $F_f = CSLGF(F_r, E_c)$ and $F_f \in \mathbb{R}^{H \times W \times CC}$ represents a unified line-aware, fault-sensitive feature embedding. This formulation ensures thin conductors are preserved via edge guidance, fault regions are amplified, and background noise is suppressed.

Fault-Aware Segmentation and Localization Heads

Generic semantic segmentation merely identifies the categories of objects and does not explicitly represent sagging zones, conductor breakpoints, and flashover regions. Nevertheless, maintenance intelligence needs explicit fault diagnosis, rather than object classification. Thus, two special heads are proposed: a Segmentation Head and a Fault Localization Head.

Segmentation Head

The segmentation head performs pixel-wise multi-class classification over $\mathbb{C} = \{Conductor, Vegetation, Foreign\ object, Background\}$. Formally defined as Equation (25):

$$P_{seg} = Softmax(W_s * F_f) \quad (25)$$

Where, W_s signifies a 1×1 convolution, $P_{seg} \in \mathbb{R}^{H \times W \times |\mathbb{C}|}$. The predicted segmentation map is defined as Equation (26):

$$\hat{Y}_{seg} = \arg\max_{\mathbb{c} \in \mathbb{C}} P_{seg}(\mathbb{c}) \quad (26)$$

This head ensures precise delineation of conductors and foreign objects. The segmented image is depicted in Figure 5.

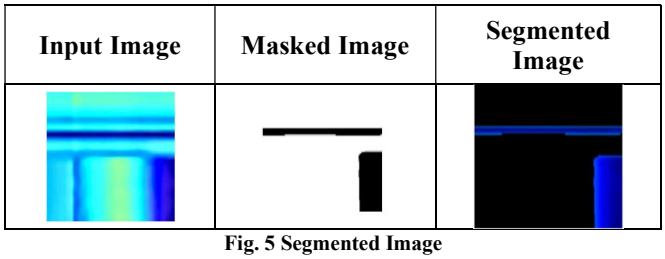

Fig. 5 Segmented Image

Fault Localization Head

The fault localization head explicitly detects $FLH = \{Sagging\ Regions, Broken\ Points, Flashover\ Areas\}$. It consists of parallel regression and classification branches, which are defined as Equation (27):

$$P_{fault} = \sigma(W_f * F_f) \quad (27)$$

Where, W_f signifies a 3×3convolution and $P_{fault} \in \mathbb{R}^{H \times W \times |FLH|}$. This produces region masks for sagging and flashover, and point-wise heatmaps for break detection. Formally defined as $\hat{Y}_{fault} = \{M_{sag}, M_{flash}, H_{break}\}$, where $M_{sag}, M_{flash} \in \mathbb{R}^{H \times W}$ and $H_{break} \in \mathbb{R}^{H \times W}$, is a point heatmap.

The overall output of this stage is represented as $\mathcal{O} = \{\hat{Y}_{seg}, \hat{Y}_{fault}\}$, which jointly provides object-level understanding (conductor, vegetation, foreign object), and fault-level diagnosis (sagging, break, flashover).

3.3. Feature Extraction

3.3.1. Electrical-Environmental Feature Extraction

Following preprocessing, the cleaned numerical and environmental signals $X_{clean}^{(num,env)}$ are forwarded to the Electrical-Environmental Feature Extraction module, which encodes statistical, time–frequency, and stability characteristics of power system behavior. The stage will be aimed at recording both steady-state trends and transient fault dynamics in order to have a solid downstream diagnosis.

Sliding Window Statistical Feature Extraction (SWSFE)

Let $X_{clean}^{(num,env)} = \{\mathbb{x}_\mathbb{t}\}_{\mathbb{t}=1}^{\mathbb{T}}$, $\mathbb{x}_\mathbb{t} \in \mathbb{R}^{\mathbb{d}}$ denote the multivariate signal (voltage, current, load, temperature, wind speed). A sliding window $SW_k = \{\mathbb{x}_\mathbb{t}\}_{\mathbb{t}=k}^{k+\mathbb{L}-1}$ of length $\mathbb{L}$ is applied. For each window and each channel, the Mean, Root Mean Square (RMS), Skewness, and Kurtosis statistics are computed.

These statistics quantify distributional shifts, asymmetry, and peakedness associated with abnormal operating conditions.

Time-Frequency Feature Extraction

To capture transient behaviors and oscillatory patterns, both the Discrete Wavelet Transform (DWT) and the Fast Fourier Transform (FFT) are applied.

(a) Wavelet Energy via DWT

Each signal window is decomposed into $\mathbb{J}$ levels as $\mathbb{x}(\mathbb{t}) \xrightarrow{DWT} \{A_\mathbb{J}, D_1, D_2, \ldots, D_\mathbb{J}\}$, Wavelet energy at scale $\mathbb{j}$ is computed as Equation (28):

$$E_\mathbb{j} = \sum_n |D_\mathbb{J}(n)|^2 \quad (28)$$

This encodes fault-induced high-frequency components such as arcing and sudden load changes.

(b) FFT Harmonics

The FFT of each window is $X(f) = \sum_{\mathbb{t}=0}^{\mathbb{L}-1} \mathbb{x}(\mathbb{t}) e^{-\mathbb{j}2\pi f \mathbb{t}/\mathbb{L}}$.

Harmonic magnitudes are extracted as $Har_k = |X(f_k)|$. These capture periodic disturbances and harmonic distortions caused by equipment faults.

Transient Stability Index Computation (TSIC)

Voltage Sag Depth and Current Surge Rate are calculated to model the severity of faults explicitly. The indices measure the severity and suddenness of electrical faults.

All extracted features are concatenated to form the electrical–environmental feature vector F_{ee}. This feature matrix F_{ee} is forwarded as input to the Two-Level Feature Selection Framework.

3.3.2. Geo-Temporal Feature Extraction

Following geographical and temporal preprocessing, the cleaned spatial coordinates X_{clean}^{geo} and temporal attributes X_{clean}^{time} are forwarded to the Geo-Temporal Feature Extraction module. The stage represents patterns of spatial concentration of faults and time dependence that are essential in determining recurrent fault zones and propagation patterns.

Fault Density Heatmap Generation using Kernel Density Estimation (KDE)

Let $X_{clean}^{geo} = \{(lat_i, lon_i)\}_{i=1}^{N}$ denote the set of fault locations. Kernel Density Estimation is applied to model spatial fault intensity, which is defined as Equation (29):

$$DEN(x,y) = \frac{1}{Nbw^2}\sum_{i=1}^{N} GK\left(\frac{x-l_i}{bw}, \frac{y-lo_i}{bw}\right) \quad (29)$$

Where, $GK(\cdot)$ is the Gaussian kernel and bw is the bandwidth parameter. This generates a fault density heatmap HM_{geo}, showing high-risk fault clusters and vulnerable line segments.

Spatio-Temporal Correlation Tensor Construction

To capture dependencies between spatial locations and temporal fault behavior, cross-correlation is computed between spatial coordinates and temporal sequences. This yields $T_{st} \in \mathbb{R}^{N\times N\times 2}$, encoding how fault occurrences correlate across space and time.

The final geo-temporal feature representation is obtained by concatenation $F_{gt} = [HM_{geo} \parallel T_{st}]$.

3.3.3. Textual Feature Extraction

Following textual preprocessing, the cleaned and tokenized fault descriptions X_{clean}^{text} are forwarded to the Textual Feature Extraction module, which encodes contextual semantics and domain-specific fault knowledge using a lightweight transformer architecture. This stage captures nuanced linguistic patterns associated with different fault types, such as overheating, arcing, conductor breakage, and vegetation contact.

Contextual Embedding Generation using BERT-Tiny

Let $X_{clean}^{text} = \{ss_i\}_{i=1}^{N}$ denote the set of pre-processed fault description sentences. Each sentence ss_i is tokenized and embedded using BERT-Tiny, producing contextualized token representations $\mathbb{HH}_i = BERTTiny(ss_i) \in \mathbb{R}^{L_{se}\times d_t}$, where L_{se} denotes the sequence length and d_t is the embedding dimension.

To obtain a fixed-length sentence embedding, mean pooling is applied as $es_i = \frac{1}{L_{se}}\sum_{j=1}^{L_{se}} \mathbb{HH}_{i,j}$. This yields $es_i \in \mathbb{R}^{d_t}$ which encodes contextual relationships between fault-related terms.

Domain-Aware Semantic Encoding

To enhance sensitivity to power system terminology, the contextual embeddings are further projected into a domain-aware semantic space as $es_i^{dom} = \phi_{dom}(es_i)$, where $\phi_{dom}(\cdot)$ is a learnable linear transformation optimized to emphasize keywords and patterns. This transformation biases the embedding space toward fault-discriminative semantics.

All domain-aware textual embeddings are stacked to form the textual feature matrix, which is represented as $F_{text} = \{es_1^{dom}, es_2^{dom}, \dots, es_N^{dom}\} \in \mathbb{R}^{N\times d_t}$. This textual feature representation F_{text} is forwarded as input to the Two-Level Feature Selection Framework.

3.3.4. Visual Feature Extraction

Following LineGuard-SegNet inference, the segmentation and fault localization output $\mathcal{O} = \{\hat{Y}_{seg}, \hat{Y}_{fault}\}$ are forwarded to the Visual Feature Extraction module. This stage encodes edge-strength, structural contours, and foreign object geometry, which are critical for discriminating vegetation, debris, and anomalous intrusions.

Learned Edge-Weighted Gradient Extraction (Fault-Aware Edge Head)

From the fault-aware localization output $\hat{Y}_{fault}$ a learned edge-weighted gradient map is computed to emphasize structurally significant boundaries. Let $G_x = \frac{\partial \hat{Y}_{fault}}{\partial x}$, $G_y = \frac{\partial \hat{Y}_{fault}}{\partial y}$. The gradient magnitude is given by $Gm = \sqrt{G_x^2 + G_y^2}$. This is modulated by the fault confidence map $\mathbb{O}_{fault}$ as $EW = Gm \odot \mathbb{O}_{fault}$. Where $\odot$ denotes element-wise multiplication, $EW \in \mathbb{R}^{H\times W}$ is the edge-weighted fault-aware edge map.

This formulation suppresses background edges and amplifies edges associated with sagging, breaks, and foreign objects.

Morphological Contour Tracing (MCT)

To extract precise object boundaries, morphological contour tracing is applied to the segmentation output $\hat{Y}_{seg}$.

Binary masks are first generated for each class as $Bm_{cs} = \mathbb{I}(\hat{Y}_{seg} = cs)$. Contours are then extracted using the morphological gradient as Equation (30):

$$C_{cs} = (Bm_{cs} \oplus SE) - (Bm_{cs} \ominus SE) \quad (30)$$

Where, $\oplus$ and $\ominus$ denote dilation and erosion, SE is a structuring element. This yields $C_{cs} \in \mathbb{R}^{H \times W}$, representing the contour maps for conductors, vegetation, and foreign objects.

The final visual feature representation is obtained by concatenating edge and contour features $F_{vis} = [E_w \parallel C_{cond} \parallel C_{veg} \parallel C_{fo}] \in \mathbb{R}^{H \times W \times d_{vis}}$, where C_{cond}, C_{veg}, C_{fo} are contour maps for conductor, vegetation, and foreign objects.

3.4. Feature Selection via RHO-MTS (ReliefF-Guided Hybrid Optimization using Multi-Verse Optimizer and Tabu Search)

A ReliefF-guided Hybrid Optimization framework combining Multi-Verse Optimizer (MVO) and Tabu Search (TS), which is referred to as RHO-MTS, is used to achieve compact and fault-discriminative representations without cross-modal interference. Each modality, electrical-environmental, geo-temporal, textual, and visual, is selected separately, with domain-consistent and noise-resistant subsets. The initial relevance screening is offered by ReliefF, and global exploration and local refinement are performed by MVO-TS together to produce the best modality-specific feature subsets.

3.4.1. Filter-Level Feature Screening via ReliefF

In order to do modality-specific relevance estimation and filter out weakly informative attributes before optimization, ReliefF-based filter screening is done on the extracted feature sets F_{ee}, F_{gt}, F_{text}, and F_{vis} individually. The ReliefF is a distance-based feature weighting algorithm that measures the discriminatory power of each feature by the ability of the feature to differentiate between neighboring samples of different fault classes.

Given a feature vector $x_i \in \mathbb{R}^d$ with class label y_i, ReliefF iteratively samples instances and identifies:

- Nearest Hit $\mathcal{NH}_i$: nearest neighbor from the same class
- Nearest Miss $\mathcal{NM}_i^c$: nearest neighbor from each different class c

For each feature f_j, the weight update is defined as Equation (31):

$$w_j \leftarrow w_j - \frac{1}{m}\sum_{i=1}^{m} diff(f_j, x_i, \mathcal{NH}_i) + \frac{1}{m}\sum_{c \neq y_i} \frac{\mathcal{P}ro(c)}{1-\mathcal{P}ro(y_i)} \sum_{i=1}^{m} diff(f_j, x_i, \mathcal{NM}_i^c) w_j \leftarrow \quad (31)$$

Where, m denotes the number of sampled instances, $\mathcal{P}ro(c)$ is the prior probability of class c, and $diff(\cdot)$ computes normalized feature-wise distance. This formulation punishes features that have large intra-class variance and rewards features that have large inter-class separability, which makes it effective in fault discrimination in heterogeneous modalities.

ReliefF screening is applied separately as $\mathcal{W}_{ee} = ReliefF(F_{ee})$, $\mathcal{W}_{gt} = ReliefF(F_{gt})$, $\mathcal{W}_{text} = ReliefF(F_{text})$, $\mathcal{W}_{vis} = ReliefF(F_{vis})$. A threshold τ is then applied to retain only features satisfying $\mathcal{W}_{my}(j) \geq \tau_{my}$, $my \in \{ee, gt, text, vis\}$ resulting in reduced but highly informative feature subsets, which are represented as $\check{F}_{ee}$, $\check{F}_{gt}$, $\check{F}_{text}$, and $\check{F}_{vis}$.

These ReliefF-screened sets of features are the input to the next level of optimizer-level selection by RHO-MTS, which allows efficient search in a relevance-constrained space without loss of modality.

3.4.2. Optimizer-Level Feature Selection using RHO-MTS

Inspiration and Motivation

The rationale behind the Multi-Verse Optimizer (MVO) and Tabu Search (TS) combination is the necessity to find a good balance between global exploration and local exploitation in high-dimensional and heterogeneous feature spaces. MVO is motivated by cosmological ideas of white holes, black holes, and wormholes, allowing high diversity conservation and global search, which is necessary to avoid early convergence during the selection of informative features in multimodal electrical, environmental, textual, and visual spaces. Nevertheless, MVO can also have slow convergence around optimal areas. To overcome this shortcoming, Tabu Search is incorporated as a complementary memory-based local exploitation algorithm, which enables narrowed-down neighborhood search, but avoids cycling through previously explored suboptimal solutions. This synergistic combination guarantees strong convergence to small, non-redundant sets of features with enhanced discriminative capability, making the hybrid MVO-TS system especially appropriate to the complex fault-diagnosis and infrastructure monitoring tasks.

Mathematical Model

Step 1: Population Initialization

Following ReliefF-based filter screening, optimizer-level selection is performed independently for each modality on the reduced feature spaces $\check{F}_{ee}$, $\check{F}_{gt}$, $\check{F}_{text}$, and $\check{F}_{vis}$.

For each modality $my \in \{ee, gt, text, vis\}$, an initial population $\mathcal{P}_{my} = \{z_1, z_2, \ldots, z_N\}$ is generated, where each candidate solution $z_i \in \{0,1\}^{d_{my}}$ is a binary selection vector defined as Equation (32):

$$z_i(j) = \begin{cases} 1, & if\ feature\ j\ is\ selectd \\ 0, & otherwise. \end{cases} \quad (32)$$

Here, d_{my} denotes the number of ReliefF-retained features for modality my. This constrained initialization eliminates redundant dimensions upfront and ensures that the search is restricted to relevance-preserving subspaces.

Step 2: Fitness Function Evaluation

Each candidate subset z_i is evaluated using a multi-objective scalar fitness function that jointly minimizes classification error and feature redundancy, which is defined as Equation (33):

$$\min \mathcal{J}(z_i) = E_{cls}(z_i) + \lambda p \cdot R_{red}(z_i) \quad (33)$$

Where, E_{cls} represents the classification error obtained using the selected features, R_{red} signifies the redundancy penalty computed via pairwise feature correlation, λp is a trade-off parameter controlling sparsity–accuracy balance. Redundancy is quantified as Equation (34):

$$R_{red} = \frac{1}{|\mathcal{S}|^2} \sum_{p,q \in \mathcal{S}} |corr(f_p, f_q)| \quad (34)$$

With $\mathcal{S}$ being the selected feature index set. This formulation makes sure that the optimizer prefers small, non-redundant, and highly discriminative subsets of features, and that it is strictly modality independent.

Step 3: Global Exploration via Multi-Verse Optimizer (MVO)

The Multi-Verse Optimizer (MVO) is used as the global search engine to allow the exploration of the feature subset search space in a non-myopic and diverse way. MVO is motivated by the idea of white holes, black holes, and wormholes in cosmology and is especially useful in preventing premature convergence in high-dimensional binary optimization problems. Conceptual models of these three key components of the MVO are illustrated in Figure 6.

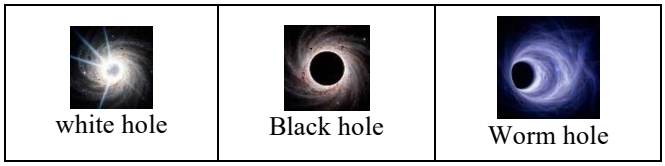

Fig. 6 White Hole, Black Hole, and Worm Hole

For each modality $my \in \{ee, gt, text, vis\}$, the population $\mathcal{P}_{my} = \{z_1, z_2, \dots, z_N\}$ evolves independently under MVO dynamics.

White Hole Selection

Candidate subsets with better fitness values (lower $\mathcal{J}$) are assigned higher inflation rates and are more likely to act as white holes, exporting their selected features to poorer candidates. Let $\mathcal{J}_i$ be the fitness of the candidate z_i. The normalized inflation rate IR_i is computed as Equation (35):

$$IR_i = \frac{\mathcal{J}_{max} - \mathcal{J}_i}{\mathcal{J}_{max} - \mathcal{J}_{min} + o} \quad (35)$$

Where, $\mathcal{J}_{max}$ and $\mathcal{J}_{min}$ are the worst and best fitness values in the population, and o avoids division by zero. Feature transfer is then performed as Equation (36):

$$z_i^{new}(j) = \begin{cases} z_k(j), & if\ r < IR_k, \\ z_i(j), & otherwise. \end{cases} \quad (36)$$

Where, z_k is selected via roulette-wheel selection among high-inflation candidates, and $r \sim \mathcal{U}(0,1)$ is a random number. This mechanism promotes the propagation of discriminative feature patterns from high-quality subsets.

Black Hole Replacement (Diversity Injection)

Candidates with poor fitness values behave as black holes, meaning their low-quality feature patterns are gradually replaced by those from better universes. Formally, for a candidate z_i with a low inflation rate as $z_i^{new}(j) = z_{best}(j)$, with probability proportional to $\mathcal{P}_{replace} = 1 - IR_i$. Where z_{best} represents the best-performing subset now. This guarantees that feeble areas of the search space are systematically removed, whilst maintaining stochasticity.

Wormhole-based Random Perturbation

MVO uses wormholes to avoid stagnation and permit long-range jumps in the solution space, where candidates randomly distort their feature selection regardless of fitness. Wormhole update is used as in Equation (37) on each candidate z_i:

$$z_i^{new}(j) = \begin{cases} 1 - z_i(j), & if\ r_1 < WEP\ and\ r_2 < 0.5, \\ z_i(j), & otherwise. \end{cases} \quad (37)$$

Where, $r_1, r_2 \sim \mathcal{U}(0,1)$ and WEP is the wormhole existence probability, defined as Equation (38):

$$WEP(t) = WEP_{min} + \frac{t}{\mathcal{T}}(WEP_{max} - WEP_{min}) \quad (38)$$

With t being the current iteration and $\mathcal{T}$ the maximum number of iterations. This program slowly changes the search into exploration and then exploitation, which allows the initial diversity and subsequent convergence.

Unified MVO Update Rule

Combining white hole, black hole, and wormhole operations, the full update for each candidate is $z_i^{t+1} = \mathcal{W}_{wormhol}\left(\mathcal{B}_{black}(\mathcal{W}_{white}(z_i^t))\right)$. This hierarchical development guarantees that White holes spread strong

patterns of features, Black holes eradicate weak ones, and Wormholes inject randomness to avoid local minima.

Step 4: Local Exploitation via Tabu Search (TS)

Once the global exploration phase based on the Multi-Verse Optimizer (MVO) has been completed, the most promising candidate feature subsets are sent to Tabu Search (TS) to be exploited intensively and refined on a fine-grained scale. Whereas MVO is charged with the task of exploring various areas of the search space, TS is concerned with systematic neighborhood search to optimize combinations of features by taking advantage of local structures and removing residual redundancy.

For each modality $my \in \{ee, gt, text, vis\}$, the elite subset z_{my}^{best} is selected and refined independently.

Neighborhood Structure Definition

Given a binary feature subset vector $z = [z_1, z_2, \dots, z_d]$, $z_j \in \{0,1\}$ the neighborhood $\mathcal{N}\hbar(z)$ is defined using bit-flip operators, which generate new candidates by:

- Additional move: $z_j: 0 \to 1$ (include feature)
- Removal move: $z_j: 1 \to 0$ (exclude feature)

Formally, the neighborhood is $\mathcal{N}\hbar(z) = \left\{ z^{(j)} | z_{\hbar}^{(j)} = \begin{cases} 1 - z_{\hbar}, & \hbar = j \\ z_{\hbar}, & \hbar \neq j \end{cases} \right\}$. This operator ensures minimal perturbation while allowing precise adjustment of the feature subset composition.

Tabu List and Memory Structure

To prevent cycling and revisiting recently explored solutions, TS maintains a Tabu List $\mathcal{TL}$, which stores forbidden moves or recently modified feature indices. Let T denote the tabu tenure. When a feature index j is flipped, it is added to $\mathcal{TL}$ for the next T iterations $\mathcal{TL} \leftarrow \mathcal{TL} \cup \{j\}$, for $t \leq t_j + T$. A move j is considered tabu if $j \in \mathcal{TL}$ unless it satisfies the aspiration criterion.

To avoid missing high-quality solutions, the aspiration rule allows tabu moves if they result in a better fitness than the current global best: $Allow\ move\ j \in \mathcal{TL}$ if $\mathcal{J}(z^{(j)}) < \mathcal{J}(z_{best})$. This ensures that optimization quality is never sacrificed for memory constraints.

Local Search Update Rule

At each iteration, TS evaluates the fitness of all admissible neighbors as Equation (39):

$$z^* = \arg \min_{z' \in \mathcal{N}\hbar(z) \setminus \mathcal{TL}} \mathcal{J}(z') \quad (39)$$

Where the fitness function is $\mathcal{J} = E_{cls} + \lambda \cdot R_{red}$. The current solution is then updated as $z \leftarrow z^*$, and the corresponding move is added to the tabu list. TS uses micro-level pruning of weak or redundant features, accurate inclusion of highly discriminative features, and minimization of redundancy by penalty-based evaluation by successively applying single-feature perturbations. This produces a local optimal feature subset that is compact and informative.

For each modality, the refinement process is defined as $F_{my}^{opt} = TS(F_{my}^{MVO}),\ my \in \{ee, gt, text, vis\}$, where F_{my}^{MVO} denotes the MVO-optimized feature subset and F_{my}^{opt} signifies the final refined subset after TS exploitation.

Step 5: Termination and Optimal Feature Subset Output

The RHO-MTS optimization procedure is stopped when the maximum iteration count is met or a stagnation period is met, and fitness is not improved. At the end of the termination, the most successful feature subsets that have been obtained after Tabu Search refinement are chosen as the final optimized representations. Formally, the output of the feature selection stage is $F_{ee}^{opt}, F_{gt}^{opt}, F_{text}^{opt}, F_{vis}^{opt}$, where F_{ee}^{opt}denotes the optimized electrical–environmental feature subset, F_{gt}^{opt}denotes the optimized geo-temporal feature subset, F_{text}^{opt}denotes the optimized textual feature subset, and F_{vis}^{opt}denotes the optimized visual feature subset.

The subsets are optimized separately without cross-modal fusion, and they provide domain-consistent, non-redundant, and fault-discriminative representations. These four optimized feature sets are then sent to the next classification phase.

3.5. Diagnosis (Fault and Foreign Object Identification) via TSTM-AttNet

A Transformer Spatial TextMobile Attention Network (TSTM-AttNet) is proposed to facilitate proper cross-domain detection of power line faults and foreign objects. The model is based on a Parallel Encoding - Attentive Fusion - Sequential Reasoning (PEAF-SR) paradigm in which the modality-specific representations are initially learned separately, then aligned through cross-modal attention, and finally optimized through shared reasoning and task-specific heads. This design enables the model to infer fault type, level of severity, and foreign object class in a unified and decision-consistent way. The general structure of the proposed TSTM-AttNet is depicted in Figure 7.

3.5.1. Tabular Transformer Branch (Electrical-Environmental Diagnosis Stream)

The Tabular Transformer branch is aimed at capturing high-order relationships between optimized electrical and environmental variables and allows the diagnosis of faults based on complex relationships between voltage, current, load, temperature, wind speed, and weather conditions. The input to this branch is the optimized electrical–environmental feature subset F_{ee}^{opt}.

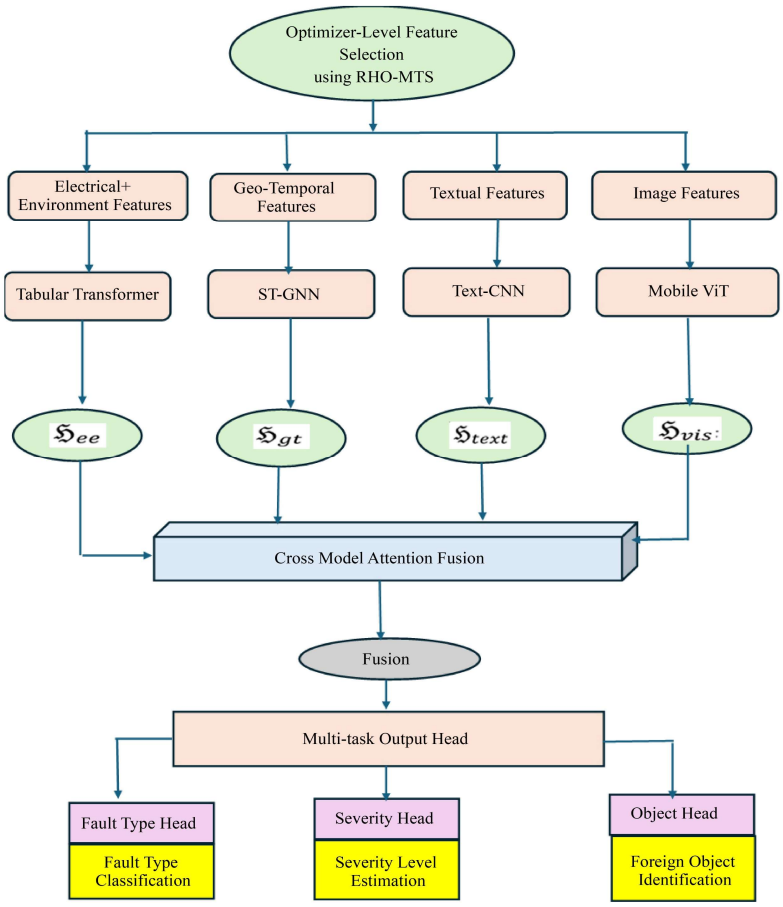

Fig. 7 General Structure of the Proposed TSTM-AttNet

Each feature dimension in F_{ee}^{opt} is treated as a token. A linear embedding layer projects the feature vector into a latent space as Equation (40):

$$\mathcal{E}_{ee} = F_{ee}^{opt}\mathfrak{W}_e + \mathfrak{b}_e \quad (40)$$

Where, $\mathfrak{W}_e \in \mathbb{R}^{d_{ee} \times d_{model}}$ and $\mathfrak{b}_e$ is the bias term. This transforms the tabular features into a sequence of embeddings as $\mathcal{E}_{ee} \in \mathbb{R}^{\mathfrak{N} \times d_{model}}$. Positional encodings PE are added to retain feature ordering and structural awareness, which is represented as $\mathfrak{Z}_{ee}^{(0)} = \mathcal{E}_{ee} + PE$.

To capture interdependencies among electrical and environmental attributes, multi-head self-attention (MHSA) is applied, which is expressed as Equation (41):

$$Attention(Qu, Ke, Va) = softmax\left(\frac{QuKe^T}{\sqrt{d_k}}\right)Va \quad (41)$$

With $Qu = \mathfrak{Z}_{ee}^{(\ell)}\mathcal{W}_{Qu}$, $Ke = \mathfrak{Z}_{ee}^{(\ell)}\mathcal{W}_{Ke}$, $Va = \mathfrak{Z}_{ee}^{(\ell)}\mathcal{W}_{Va}$. Where $\mathcal{W}_{Qu}, \mathcal{W}_{Ke}, \mathcal{W}_{va} \in \mathbb{R}^{d_{model} \times d_k}$. The multi-head formulation is as $MHSA(\mathfrak{Z}) = Concat(head_1, \ldots, head_{\hbar})\mathcal{W}_O$. This mechanism enables the model to learn relations such as High temperature + high load → thermal stress, High wind + current surge → conductor swing, and Weather instability + voltage sag → insulation degradation.

Feed-Forward Transformation and Residual Learning

Each attention block is followed by a position-wise feed-forward network (FFN), which is defined as Equation (42):

$$FFN(\mathfrak{x}) = \sigma(\mathfrak{x}\mathcal{W}_1 + \mathscr{b}_1)\mathcal{W}_2 + \mathscr{b}_2 \quad (42)$$

Residual connections and layer normalization are applied to ensure stable training, which is expressed as Equation (43):

$$\mathfrak{Z}_{ee}^{(\ell+1)} = LayerNorm\left(\mathfrak{Z}_{ee}^{(\ell)} + FFN\left(MHSA(\mathfrak{Z}_{ee}^{(\ell)})\right)\right) \quad (43)$$

After $\mathfrak{L}$ transformer layers, the final electrical–environmental diagnostic embedding is obtained as $\mathfrak{H}_{ee} = \mathfrak{Z}_{ee}^{(\mathfrak{L})} \in \mathbb{R}^{\mathfrak{N} \times d_{model}}$. The output $\mathfrak{H}_{ee}$ represents a context-aware electrical–environmental diagnostic representation capturing

nonlinear interactions among power signals and environmental stressors. This embedding is not fused at this stage and is directly forwarded to the cross-modal attention fusion module.

3.5.2. Spatio-Temporal Graph Neural Network (ST-GNN) Branch (Geo-Temporal Diagnosis Stream)

The ST-GNN branch is developed to capture the spatial topology and time-dependent propagation of faults throughout the power transmission system. The input to this branch is the optimized geo-temporal feature subset F_{gt}^{opt}.

The power transmission network is represented as a graph $\mathcal{G} = (\mathfrak{V}, \mathfrak{E})$, where $\mathfrak{V}$ denotes transmission towers/line segments, and $\mathfrak{E}$ denotes physical adjacency or proximity-based connectivity. The adjacency matrix $AD \in \mathbb{R}^{|\mathfrak{V}|\times|\mathfrak{V}|}$ is constructed using spatial distance thresholding as Equation (44):

$$AD_{\mathrm{ij}} = \begin{cases} 1, & if\ dist\left(\mathfrak{v}_{\mathrm{i}}, \mathfrak{v}_{\mathrm{j}}\right) \le \varepsilon, \\ 0, & otherwise. \end{cases} \quad (44)$$

Each node $\mathfrak{v}_{\mathrm{i}}$ is initialized with its geo-temporal feature vector $X_{\mathrm{i}}^{(0)} = F_{gt,\mathrm{i}}^{opt}$. Spatial dependencies are captured using graph convolution, which is defined as Equation (45):

$$\mathfrak{H}^{(\ell+1)} = ReLU\left(\widetilde{DM}^{-1/2}\widetilde{AD}\widetilde{DM}^{-1/2}\mathfrak{H}^{(\ell)}\mathcal{W}^{(\ell)}\right) \quad (45)$$

Where, $\widetilde{AD} = AD + I$ represents the self-loop augmented adjacency, $\widetilde{DM}$ denotes the degree matrix, and $\mathcal{W}^{(\ell)}$ signifies learnable weights. This operation enables the model to learn regional fault influence patterns, such as cascading outages and cluster-based anomalies.

To capture fault evolution over time, temporal modeling is applied using gated temporal convolutions as $\mathfrak{H}_{\mathrm{t}}' = Conv1D(\mathfrak{H}_{\mathrm{t}-k:\mathrm{t}})$. This enables learning of fault recurrence patterns, propagation delays, and storm-driven fault diffusion.

The combined spatio-temporal update is expressed as $\mathfrak{H}_{gt}^{(\ell+1)} = Temporal\left(GCN\left(H_{gt}^{(\ell)}, AD\right)\right)$. After $\mathfrak{L}$ layers, the final geo-temporal embedding is obtained $\mathfrak{H}_{gt} = H_{gt}^{(\mathfrak{L})}$. The output $\mathfrak{H}_{gt}$ encodes fault diffusion behavior, regional vulnerability, and temporal recurrence patterns.

3.5.3. TextCNN Branch (Textual Fault Narrative Diagnosis Stream)

The TextCNN branch is applied to recognize discriminative semantic patterns with a focus on optimized textual features on the basis of maintenance logs, operator reports, inspection notes, and incident descriptions. These unstructured text logs often encode significant information about the cause of faults, the presence of foreign objects, and abnormal operation that may not be directly visible according to sensor readings. The input to this branch is the optimized textual feature set $F_{text}^{opt} \in \mathbb{R}^{N_e\times L_T}$, where N_e is the number of fault events and L_T denotes the maximum token length after preprocessing and selection.

Each textual sequence is first mapped into a dense embedding space using a trainable embedding matrix $\mathcal{E}_{text} = Embed\left(F_{text}^{opt}\right) \in \mathbb{R}^{N_e\times L_T\times d_w}$, where d_w is the embedding dimension. This transforms discrete tokens into continuous semantic vectors, enabling similarity learning across fault descriptions.

To capture n-gram patterns of varying lengths, parallel 1D convolutional filters with different kernel sizes $ks \in \{3,4,5\}$ are applied as $C_{ks} = \sigma\left(Conv1D_{ks}(\mathcal{E}_{text})\right)$. Each convolution operation is defined as Equation (46):

$$C_{ks}(\mathrm{i}) = ReLU(\mathcal{W}_{ks} \cdot \mathcal{E}_{text}[\mathrm{i}{:}\,\mathrm{i} + ks - 1] + \mathscr{b}_{ks}) \quad (46)$$

Where, $\mathcal{W}_{ks}$ the kernel weight matrix and $\mathscr{b}_{ks}$ is the bias term. This enables the model to detect short phrases ("tree fallen", "bird strike"), medium patterns ("foreign object on line"), and long contextual cues ("insulation damaged due to debris").

To retain the most salient textual features, global max-pooling is applied as $PO_{ks} = \max(C_{ks})$. The pooled features from all kernel sizes are concatenated as $\mathfrak{H}_{text}^{raw} = [PO_3 \parallel PO_4 \parallel PO_5]$. This yields a fixed-length representation independent of input sequence length.

The aggregated vector is passed through a fully connected layer, which is mathematically given by Equation (47):

$$\mathfrak{H}_{text} = Dropout\left(ReLU(\,\mathfrak{H}_{text}^{raw}W_{\mathrm{t}} + b_{\mathrm{t}})\right) \quad (47)$$

Where, W_{t} and b_{t} are learnable parameters. Dropout improves generalization and prevents overfitting to recurring textual patterns.

This representation encodes semantic fault descriptions, foreign object descriptions, and contextual operation indicators. It is completely autonomous and is sent to the fusion module without any communication with other modalities.

3.5.4. MobileViT Branch (Visual Fault and Foreign Object Diagnosis Stream)

MobileViT branch is tasked with deriving high-level visual semantics out of optimized image features, which allows detecting physical faults and foreign objects, including fallen trees, birds, kites, plastic sheets, conductor damage, and insulator cracks. The input to this branch is the optimized

visual feature subset $F_{vis}^{opt} \in \mathbb{R}^{\Re \times H \times W \times C}$, where H, W, C denote image height, width, and channels, respectively.

The input images are first processed through a lightweight convolutional stem as $\mathfrak{X}_0 = \sigma\left(Conv_{3\times3}\left(F_{vis}^{opt}\right)\right)$. This step obtains low-level visual primitives like edges, contours, and textures.

Depthwise separable convolutions are used to reduce computational cost while preserving spatial detail, which is defined as $\mathfrak{X}_{dw} = DWConv(\mathfrak{X}_0)$, $\mathfrak{X}_{pw} = PWConv(\mathfrak{X}_{dw})$. These blocks efficiently capture local patterns such as insulator shapes, wire continuity, and object silhouettes.

To integrate global dependencies, feature maps are unfolded into non-overlapping patches, which are denoted as $\mathfrak{PA} = Unfold\left(\mathfrak{X}_{pw}\right) \in \mathbb{R}^{\Re \times n_{pa} \times d_{pa}}$. Where, n_{pa} is the number of patches and d_{pa} is the patch dimension.

The patches are passed through transformer layers, which gives us Equation (48):

$$Attention(\mathfrak{PA}) = softmax\left(\frac{QTKT^T}{\sqrt{d_k}}\right)VT \tag{48}$$

Here, $QT = \mathfrak{PA}W_{QT}$, $KT = \mathfrak{PA}W_{KT}$, and $VT = \mathfrak{PA}W_{VT}$. Followed by Equation (49) and (50):

$$\mathfrak{PA}' = LayerNorm\left(\mathfrak{PA} + Attention(\mathfrak{PA})\right) \tag{49}$$

$$\mathfrak{PA}'' = LayerNorm\left(\mathfrak{PA}' + FFN(\mathfrak{PA}')\right) \tag{50}$$

This enables the model to learn object-to-object relationship, consistency of the scene context, and detection of foreign objects that are occluded.

The transformer-encoded patches are folded back into spatial form as $\mathfrak{X}_{vit} = Fold(\mathfrak{PA}'')$. A global average pooling layer produces the final visual embedding, which is represented as $\mathfrak{H}_{vis} = GAP(\mathfrak{X}_{vit})$

The output $\mathfrak{H}_{vis}$ encodes structural damage cues, foreign object presence, and visual fault patterns. It is kept strictly independent and forwarded to the fusion module.

3.5.5. Cross-Modal Attention Fusion

The learned representations of all four branches are then synchronized and integrated together through a cross-modal attention fusion module after independent modality-specific encoding. This phase is the only point of interaction between modalities, and it is the one that aligns, weights, and integrates complementary information in order to diagnose jointly. The inputs to this module are $\mathfrak{H}_{ee}, \mathfrak{H}_{gt}, \mathfrak{H}_{text}$, and $\mathfrak{H}_{vis}$, where each vector encodes modality-specific diagnostic cues.

To enable attention-based interaction, each modality is first projected into a shared embedding space of dimension $\mathfrak{d}$, which is defined as Equation (51):

$$\widetilde{\mathfrak{H}}_{my} = \mathfrak{H}_{my}\mathfrak{W}_{my} + \mathfrak{b}_{my}, \qquad my \in \{ee, gt, text, vis\} \tag{51}$$

This ensures dimensional compatibility and semantic alignment across heterogeneous modalities.

Cross-Modal Attention Mechanism

For each modality, queries are generated and attend over keys and values from all other modalities as $\mathcal{Q}u_{my} = \widetilde{\mathfrak{H}}_{my}\mathfrak{W}_{\mathcal{Q}u}$, $\mathcal{K}e_{ny} = \widetilde{\mathfrak{H}}_{ny}\mathfrak{W}_{\mathcal{K}e}$, $\mathcal{V}a_{ny} = \widetilde{\mathfrak{H}}_{ny}\mathfrak{W}_{\mathcal{V}a}$. The attention from modality my to modality ny is computed as Equation (52):

$$\mathcal{ATT}_{my\to ny} = softmax\left(\frac{\mathcal{Q}u_{my}\mathcal{K}e_{ny}^T}{\sqrt{\mathfrak{d}}}\right)\mathcal{V}a_{ny} \tag{52}$$

This operation allows electrical features to attend to geo-temporal patterns, textual cues to attend to visual evidence, and visual features to attend to sensor anomalies, etc.

For each modality, attended representations from all other modalities are aggregated as $\widetilde{\mathfrak{H}}_{my} = \sum_{ny\neq my}\mathcal{ATT}_{my\to ny}$. The enhanced modality representations are then combined as $\mathfrak{H}_{fusion} = Concat\left(\widetilde{\mathfrak{H}}_{ee}, \widetilde{\mathfrak{H}}_{gt}, \widetilde{\mathfrak{H}}_{text}, \widetilde{\mathfrak{H}}_{vis}\right)$. This concatenation is followed by a fusion projection, which is given by Equation (53):

$$\mathfrak{H}_{fusion} = \sigma\left(\mathfrak{H}_{fusion}\mathfrak{W}_f + \mathfrak{b}_f\right) \tag{53}$$

Critically, this stage does not perform classification—it only produces a unified diagnostic representation, which is passed to the reasoning and decision layers.

3.5.6. Multi-Task Output Heads (Fault Type, Severity, Foreign Object Classification)

After cross-modal fusion, the combined representation is then processed by a shared reasoning layer and broken down into task-specific heads to collectively predict fault type, level of severity, and foreign object class. This design is based on a shared-private multi-task learning paradigm. First, the fused embedding is refined through a shared transformation, which is represented as Equation (54):

$$\mathcal{Z} = \sigma\left(\mathfrak{H}_{fusion}\mathfrak{W}_s + \mathfrak{b}_s\right) \tag{54}$$

This layer captures joint fault semantics, cross-modal correlations, and contextual dependencies.

Task-Specific Output Heads

Each task has a dedicated prediction head to avoid interference while benefiting from shared context.

Fault type classification head predicts the fault category (line breakage, insulator failure, overheating, arcing, etc.), which is defined as Equation (55):

$$\hat{\mathfrak{y}}_{fault} = softmax\left(\mathcal{Z}\mathfrak{W}_{fault} + \mathfrak{b}_{fault}\right) \quad (55)$$

Severity level estimation head estimates fault severity (low, medium, high, critical) based on combined evidence, which is defined as Equation (56):

$$\hat{\mathfrak{y}}_{sev} = softmax(\mathcal{Z}\mathfrak{W}_{sev} + \mathfrak{b}_{sev}) \quad (56)$$

Foreign object classification head identifies the foreign object type (tree, bird, kite, plastic, debris), which is expressed as Equation (57):

$$\hat{\mathfrak{y}}_{obj} = softmax\left(\mathcal{Z}\mathfrak{W}_{obj} + \mathfrak{b}_{obj}\right) \quad (57)$$

Multi-Task Loss Function

The network is trained using a weighted multi-task objective, which is defined as Equation (58):

$$\mathcal{L}\mathfrak{m} = \mathrm{A}\mathcal{L}\mathfrak{m}_{fault} + \mathrm{B}\mathcal{L}\mathfrak{m}_{sev} + \Gamma\mathcal{L}\mathfrak{m}_{obj} \quad (58)$$

Where each term is categorical cross-entropy $\mathcal{L}\mathfrak{m}_{fault} = -\sum_{\mathfrak{y}_{fault}} \log\left(\hat{\mathfrak{y}}_{fault}\right)$ and $\mathrm{A}, \mathrm{B}, \Gamma$ control task importance.

Importantly, the heads operate in parallel but rely on a shared semantic backbone, enabling consistent and explainable predictions.

3.6. Localization

This stage estimates the precise geospatial coordinates of detected faults and foreign objects using the segmented visual output $\mathcal{O} = \{\hat{Y}_{seg}, \hat{Y}_{fault}\}$, where $\hat{Y}_{seg}$ denotes object masks and $\hat{Y}_{fault}$ denotes fault regions (sagging, break, flashover).

Spatial Regression Network (SRN) is a lightweight network that regresses fault regions into geographic coordinates, and is expressed as Equation (59):

$$\widetilde{LC} = \left(\widehat{lat}, \widehat{lon}\right) = \mathfrak{f}_{SRN}\left(\mathcal{O}, \mathcal{A}_{geo}, \mathcal{C}_{vis}, \mathfrak{P}_{pmu}\right) \quad (59)$$

Where, $\mathcal{A}_{geo}$ denotes geo-attention emphasizing transmission corridors, $\mathcal{C}_{vis}$ is the visual saliency centroid of fault regions, and $\mathfrak{P}_{pmu}$ signifies PMU anomaly propagation paths.

Visual centroids are computed as $\mathcal{C}_{vis} = \frac{1}{|\Omega_f|}\sum_{(x,y)\in\Omega_f}(x, y)$. The fused spatial representation is linearly regressed to coordinates $\widetilde{LC} = W_{\mathfrak{l}}\left[\mathcal{A}_{geo} \parallel \mathcal{C}_{vis} \parallel \mathfrak{P}_{pmu}\right] + b_{\mathfrak{l}}$. This module delivers fault-aware and physically consistent geolocation of sagging zones, broken conductors, and flashover regions. The Grad-Cam result is depicted in Figure 8.

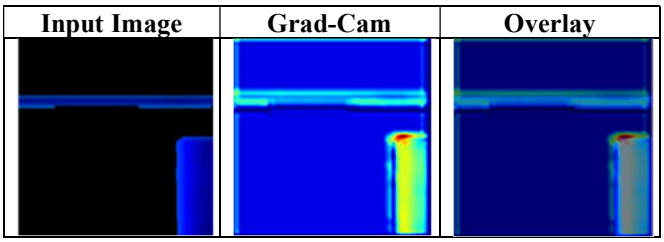

Fig. 8 Grad-Cam Outcome

3.7. Recommendation System (Maintenance Intelligence)

This step converts the results of the diagnosis into meaningful maintenance choices based on a reinforcement learning-based reasoner and scheduler.

3.7.1. Cause-Effect Reasoning Engine

The policy-based reinforcement learning agent is a mathematical model that captures the causality between observed fault patterns and maintenance decisions that are captured in the following equation (60):

$$\pi^*(\mathfrak{a}|\mathfrak{s}) = \arg\max_{\pi} \mathbb{E}\left[\sum_{\mathfrak{t}} \xi^{\mathfrak{t}}\mathfrak{R}(\mathfrak{s}_{\mathfrak{t}}, \mathfrak{a}_{\mathfrak{t}})\right] \quad (60)$$

Where the state $\mathfrak{s}$ is constructed from $\mathfrak{s} = \left[\hat{\mathfrak{y}}_{fault}, \hat{\mathfrak{y}}_{sev}, \hat{\mathfrak{y}}_{obj}, \widetilde{LC}\right]$ and the action $\mathfrak{a}$ corresponds to maintenance operations (trimming, replacement, re-tensioning). The reward function encodes operational effectiveness by $\mathfrak{R} = \alpha \cdot Fault\ Mitigation - \beta \cdot Cost - \delta \cdot Downtime$. This enables context-aware recommendations such as vegetation clearance, insulator replacement, conductor re-tensioning, and reconductoring.

3.7.2. Priority Scheduling via PPO Optimization

To optimally sequence maintenance tasks, a Proximal Policy Optimization (PPO) scheduler solves $\min(Downtime, Risk, Cost)$, $\max(Reliability)$. The PPO objective is defined as Equation (61):

$$L^{PPO} = \mathbb{E}[\min(\mathfrak{r}_{\mathfrak{t}}(\theta)A_{\mathfrak{t}}, clip(\mathfrak{r}_{\mathfrak{t}}(\theta), 1-\epsilon, 1+\epsilon)A_{\mathfrak{t}})] \quad (61)$$

Where, $\mathfrak{r}_{\mathfrak{t}}(\theta)$ is the policy ratio and $A_{\mathfrak{t}}$ is the advantage estimate. This module generates a risk-conscious, cost-conscious, and reliability-optimized maintenance plan, such that the most important faults are given priority within the constraints of operation.

4. Results and Discussion

4.1. Experimental Setup

The proposed framework has been implemented using Python and tested on a fused multimodal dataset built on the Power System Faults Dataset and the Smart Grid Phasor

Measurement Unit Fault Images. This integrated data has facilitated a thorough evaluation of fault detection as well as foreign object detection in cross-domain conditions. The proposed method has been contrasted with state-of-the-art methods, such as DF-YOLO [17], TL-YOLO [21], YOLOv7-CWFD [24], and CSPD-YOLO [26], in the case of performance benchmarking. The analysis has been performed based on conventional classification measures, i.e., Accuracy (percent), Sensitivity (percent), Precision (percent), Specificity (percent), F1-Score (percent), Negative Predictive Value (NPV (percent)), Matthews Correlation Coefficient (MCC (percent)), False Positive Rate (percent), and False Negative Rate (percent). The graphical results, such as accuracy/loss vs. epochs, confusion matrix, and ROC curve analysis of the proposed framework, are depicted in Figures 9, 10, and 11, respectively.

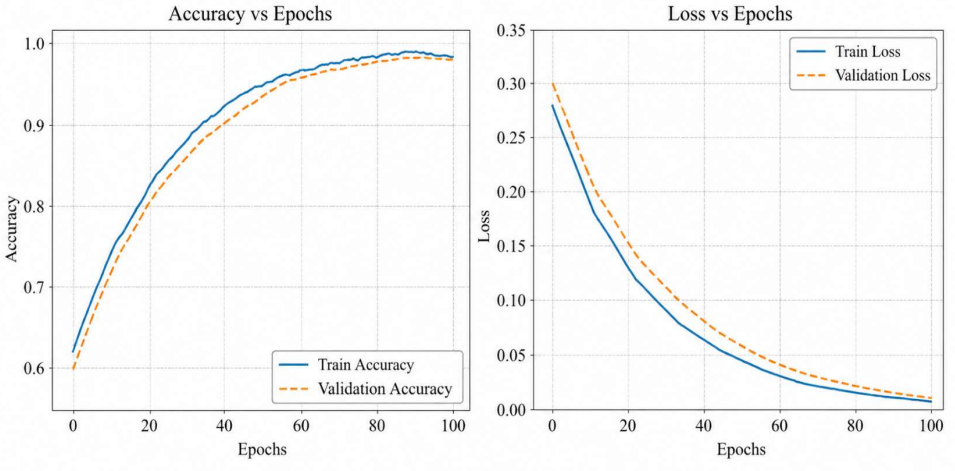

Fig. 9 Accuracy/Loss vs. Epochs

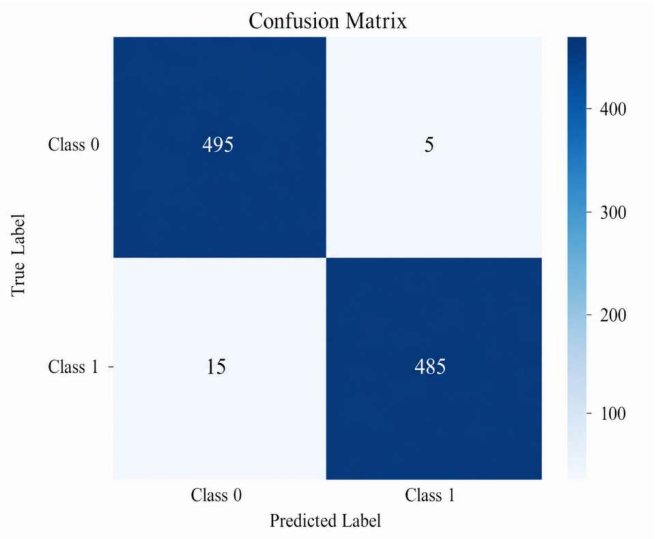

Fig. 10 Confusion Matrix Analysis

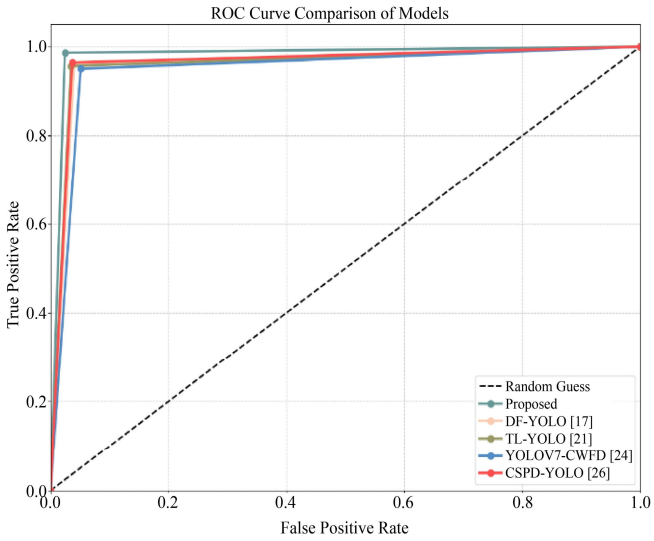

Fig. 11 ROC Curve Analysis

4.2. Performance Comparison Analysis

Table 2 and Figure 12 demonstrate the comparative performance of the proposed framework with DF-YOLO [17], TL-YOLO [21], YOLOv7-CWFD [24], and CSPD-YOLO [26]. The proposed method has the best Accuracy of 98.8, which is better than DF-YOLO (96.52%), TL-YOLO (95.89%), YOLOv7-CWFD (95.35%), and CSPD-YOLO (96.17%). This is mainly due to the LineGuard-SegNet-based line segmentation, which allows the accurate isolation of fine conductors and fault areas, and background interference is greatly minimized. Regarding Precision, the suggested approach achieves 97.53%, which is higher than DF-YOLO (95.93%), TL-YOLO (95.42%), YOLOv7-CWFD (94.8%), and CSPD-YOLO (96.26%). This improvement is due to the efficacy of fault-conscious edge modeling and contour extraction, which inhibit false detection of vegetation and background structures. Equally, the Sensitivity of 98.62% is higher than any competing method, meaning better fault recall, and this is because the TSTM-AttNet multimodal diagnosis module jointly uses electrical, geo-temporal,

textual, and visual cues. The proposed framework also has the best Specificity (97.88) and NPV (98.44), which illustrate strong discrimination between faulty and healthy regions. This is a direct consequence of the RHO-MTS feature selection strategy, which removes redundant and noisy features, leaving only discriminative attributes in each of the modalities. In the case of F1-Score, the suggested approach achieves 97.57, which is better than DF-YOLO (96.01%), TL-YOLO (96.54%), YOLOv7-CWFD (94.91%), and CSPD-YOLO (96.34%). This equal enhancement of accuracy and recall is made possible by cross-modal attention fusion in TSTM-AttNet, which aligns complementary information across heterogeneous sources. It is worth noting that the proposed approach has the lowest FPR of 2.74% and the lowest FNR of 1.66% as compared to DF-YOLO (4.27%, 3.96%), TL-YOLO (5.83%, 4.59%), YOLOv7-CWFD (5.33%, 4.87%), and CSPD-YOLO (4.94%, 3.75%). Moreover, the MCC of 98.59% proves that there is a strong correlation between predicted and actual labels, which is much higher than DF-YOLO (95.96%), TL-YOLO (95.58%), YOLOv7-CWFD (94.86%), and CSPD-YOLO (96.37%), which confirms the overall robustness and stability of the proposed framework.

Table 2. Performance Comparison of the Proposed Method with Existing Techniques

Model	Accuracy (%)	Precision (%)	Sensitivity (%)	Specificity (%)	F1 Score (%)	NPV (%)	MCC (%)	FPR (%)	FNR (%)
Proposed	**98.8**	**97.53**	**98.62**	**97.88**	**97.57**	**98.44**	**98.59**	**2.74**	**1.66**
DF-YOLO [17]	96.52	95.93	96.09	96.4	96.01	95.78	95.96	4.27	3.96
TL-YOLO [21]	95.89	95.42	95.66	95.78	96.54	95.3	95.58	5.83	4.59
YOLOv7-CWFD [24]	95.35	94.8	95.03	95.49	94.91	94.69	94.86	5.33	4.87
CSPD-YOLO [26]	96.17	96.26	96.42	96.61	96.34	96.12	96.37	4.94	3.75

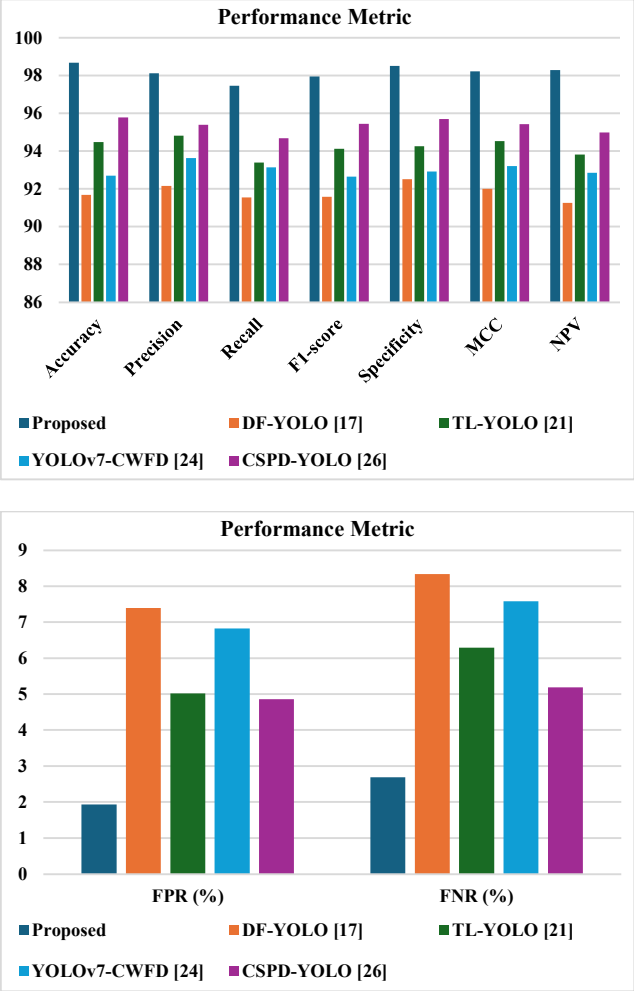

Fig. 12 Graphical Comparison of Performance Metrics for Proposed and Existing Methods

4.3. K-Fold Comparison Analysis

The strength and generalization ability of the suggested framework are tested with the help of 5-fold cross-validation, and the comparative findings are presented in Table 3 and graphically in Figure 13. The suggested approach has the highest performance in all folds with 98.62% (K1), 98.75% (K2), 98.91% (K3), 98.84% (K4), and 98.88% (K5). This stability means that it is highly resistant to bias in data partitioning and ensures that learning is reliable across a variety of fault patterns.

Conversely, DF-YOLO [17] achieves 96.31, 96.48, 96.55, 96.62, and 96.64 on K1-K5, whereas TL-YOLO [21] gets 95.63, 95.78, 95.85, 95.94, and 95.95. Likewise, YOLOv7-CWFD [24] is less consistent with values of 95.12, 95.26, 95.33, 95.44, and 95.6, and CSPD-YOLO [26] has 95.98, 96.09, 96.15, 96.23, and 96.4.

The high and consistent performance of the proposed framework on all folds is explained by the LineGuard-SegNet-based line-aware segmentation, which guarantees the consistent conductor isolation, and the RHO-MTS feature selection strategy, which eliminates fold-specific noise and redundancy. In addition, the TSTM-AttNet multimodal diagnosis module preserves discriminative representation learning across different training-testing splits, which improves cross-domain generalization.

Table 3. K-Fold Cross-Validation Performance Comparison of Proposed and Existing Methods

Model	K1	K2	K3	K4	K5
Proposed	98.62	98.75	98.91	98.84	98.88
DF-YOLO [17]	96.31	96.48	96.55	96.62	96.64
TL-YOLO [21]	95.63	95.78	95.85	95.94	95.95

YOLOv7-CWFD [24]	95.12	95.26	95.33	95.44	95.6
CSPD-YOLO [26]	95.98	96.09	96.15	96.23	96.4

The high level of performance demonstrated by the proposed framework can be explained by several design factors. To begin with, the LineGuard-SegNet module allows fine-tuning of the thin transmission lines and fault area, which is very beneficial in minimizing the background interference and maximizing the localization accuracy. Second, the RHO-MTS feature selection strategy removes duplicate and noisy features between modalities, hence increasing the discriminative ability of the model. Third, the TSTM-AttNet architecture conducts cross-modal attention-based fusion, which enables the use of the supplementary information in electrical, environmental, textual, and visual data to be used together. Such a multimodal combination greatly enhances the ability to be robust in complicated environmental circumstances when single-domain approaches are likely to fail. Conversely, current state-of-the-art methods mainly use visual characteristics and do not fuse effectively across domains, resulting in lower generalization and increased false detection. Thus, structured multimodal learning, optimized feature selection, and attention-based fusion allow the proposed method to outperform the existing ones in all metrics of evaluation all the time.

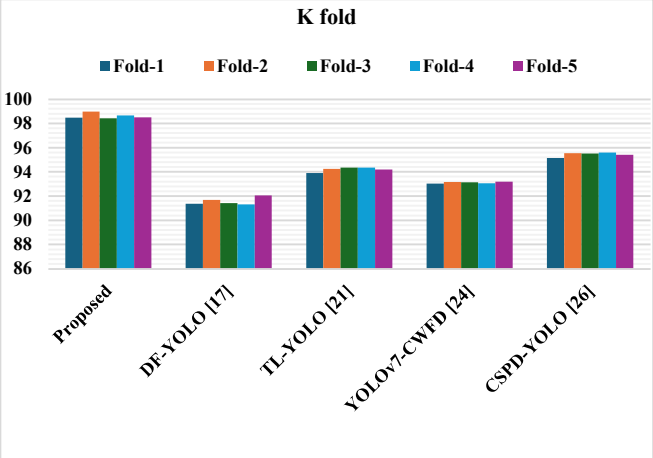

Fig. 13 Graphical Comparison of K-Fold Accuracies for Proposed and Existing Methods

4.4. Impact of Pre-processing

The impact of the suggested pre-processing pipeline on the detection performance is examined and presented in Figure 14. The proposed framework has an Accuracy of 96.8% with pre-processing and 95.81% without pre-processing, which is a definite improvement in performance. The same is true of Precision (95.53% vs. 94.54%), Sensitivity (96.62% vs. 95.63%), and Specificity (95.88% vs. 94.89%), which proves the efficiency of the sequential cleaning, normalization, and enhancement steps.

All comparison methods exhibit a consistent trend. DF-YOLO [17] increases its accuracy by 93.53 to 94.52, TL-YOLO [21] by 92.9 to 93.89, YOLOv7-CWFD [24] by 92.36 to 93.35, and CSPD-YOLO [26] by 93.18 to 94.17 with pre-processing. These improvements confirm that noise reduction, contrast, and data normalization are effective features that enhance the quality of features among models.

The higher quality of the offered approach can be explained by the domain-adaptive image enhancement, geographical normalization, and textual cleaning steps, which enhance the next LineGuard-SegNet segmentation and TSTM-AttNet diagnosis modules.

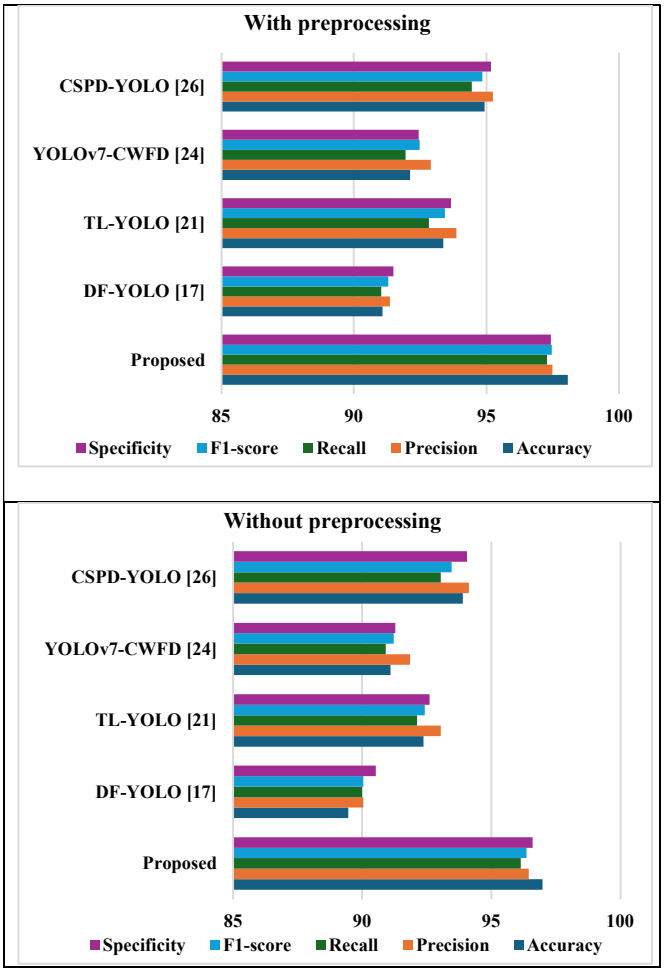

Fig. 14 Graphical Comparison of Performance Metrics with and Without Pre-processing

4.5. Impact of Feature Extraction

The effect of the feature extraction step on the performance of the system is assessed and represented in Figure 15.

The Accuracy of the proposed framework with feature extraction is 97.68%, and without feature extraction is 96.7, which is a definite improvement. The same improvement is

also noted in Precision (96.41% vs. 95.43%), Sensitivity (97.5% vs. 96.52%), and Specificity (96.76% vs. 95.78%), which proves the efficiency of the multi-branch feature learning modules. All baseline methods show consistent improvement when feature extraction is enabled. DF-YOLO [17] improves from 94.42% to 95.4% accuracy, TL-YOLO [21] from 93.79% to 94.77%, YOLOv7-CWFD [24] from 93.25% to 94.23%, and CSPD-YOLO [26] from 94.07% to 95.05%. These results validate that structured feature encoding enhances discriminative capability across models.

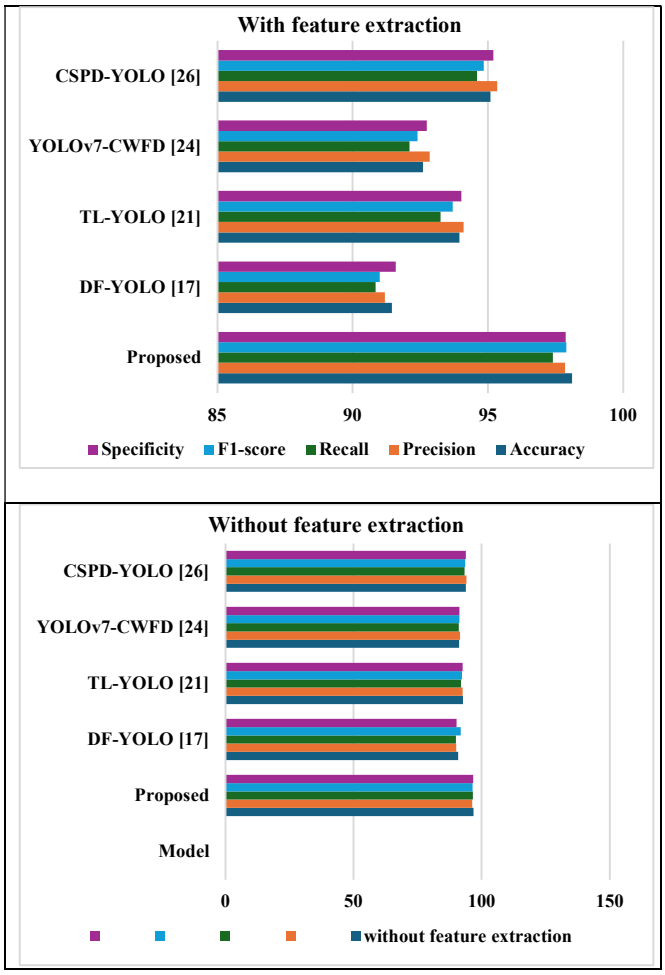

Fig. 15 Graphical Comparison of Performance Metrics with and Without Feature Extraction

4.6. Impact of Feature Selection

Effects of feature selection on the model performance are investigated and presented in Figure 16. The performance of the proposed framework is at 97.78% with feature selection and 96.81% without feature selection, which represents a certain enhancement.

Precision (96.51% vs. 95.54%), Sensitivity (97.6% vs. 96.63%), Specificity (96.86% vs. 95.89%), and F1-Score (96.55% vs. 95.58%) also show improvements, which proves the advantage of optimized feature subset selection.

All baseline methods show similar trends. DF-YOLO [17] improves from 94.53% to 95.5% accuracy, TL-YOLO [21] from 93.9% to 94.87%, YOLOv7-CWFD [24] from 93.36% to 94.33%, and CSPD-YOLO [26] from 94.18% to 95.15% when feature selection is applied. However, the proposed approach consistently outperforms all existing techniques across all metrics.

This high performance can be attributed to the fact that the RHO-MTS feature selection strategy combines ReliefF filtering with the Multi-Verse Optimizer for global exploration and Tabu Search for local exploitation. This hybrid optimization is useful in eliminating redundant and noisy features and maintaining highly discriminative features, resulting in better class separability and more robust learning.

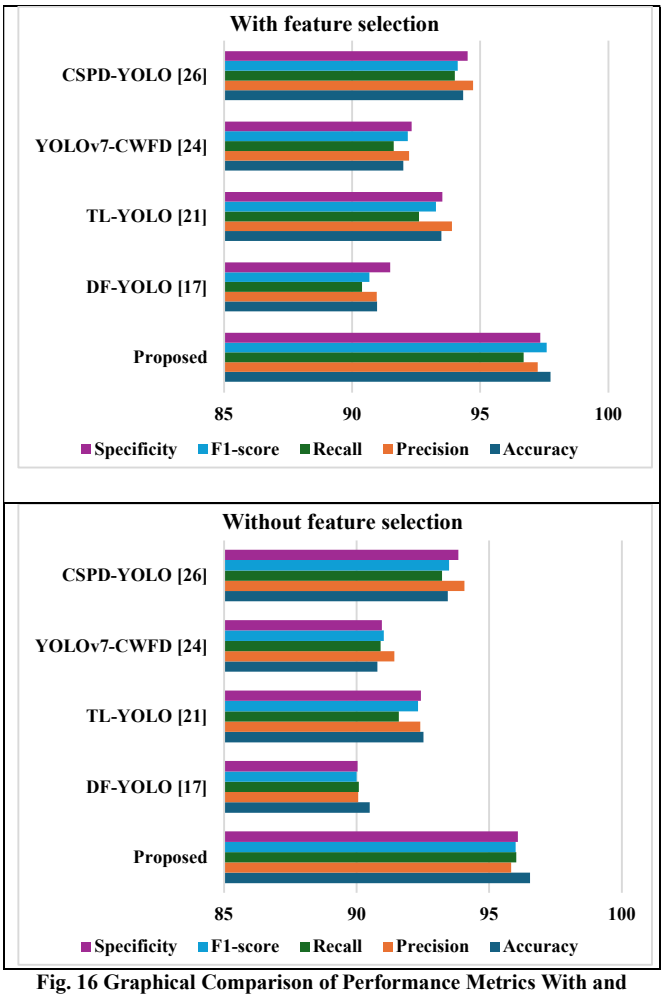

Fig. 16 Graphical Comparison of Performance Metrics With and Without Feature Selection

5. Conclusion

This work introduced a superior deep learning system to detect faults and foreign objects on power transmission lines across domains. The proposed pipeline combined structured power system data and PMU fault images in a Unified Event-Centric Data Model (UECDM) to successfully align electrical,

environmental, textual, visual, geo-spatial, and temporal data. Isolation Forest-based outlier removal, temporal KNN imputation, normalization, WordPiece tokenization, and enhanced image enhancement with LineGuard-SegNet segmentation were used to perform pre-processing. The statistical, time-frequency, stability indices, BERT-Tiny embeddings, and fault-aware visual descriptors are used to extract discriminative features. The RHO-MTS strategy was used to optimize feature selection, and it was a combination of ReliefF, Multi-Verse Optimizer, and Tabu Search. The TSTM-AttNet architecture with parallel Tabular Transformer, ST-GNN, TextCNN, and MobileViT branches was used to diagnose faults and foreign objects, and then the cross-modal attention fusion was performed. A spatial regression network was used to localize faults accurately, and a PPO-based reinforcement learning engine was used to offer intelligent maintenance suggestions and priority scheduling. All the implementation was done in Python. The results of the experiment demonstrated improved performance with an accuracy of 98.8% and a sensitivity of 98.62% that justifies the usefulness of the proposed approach. This work contributed to improving the reliability of the power grid, the safety of the population, and infrastructure resilience as it permitted the detection of faults in a timely and accurate manner. The framework can be extended into real-time edge deployment, extensive grid integration, and autonomous learning of evolving fault patterns in the future.

Data Availability

The data sets, which are involved in the present work, are publicly available. The structured power system operational and fault records are acquired through the Power System Faults Dataset at https://www.kaggle.com/datasets/ziya07/power-system-faults-dataset.

The visual inspection images depicting line sagging, broken conductors, flashover, vegetation intrusion, and foreign object scenarios are sourced from the Smart Grid Phasor Measurement Unit Fault Images dataset available at https://www.kaggle.com/datasets/pythonafroz/smart-grid-phasor-measurement-unit-faulty-data. Both datasets can be freely downloaded from Kaggle for research and benchmarking purposes.

Conflicts of Interest

The authors declare that there is no conflict of interest.

Funding Statement

Not Applicable.

Acknowledgments

Thanks to all participants and contributors for making this study possible.

Reference

[1] Iyke Maduako et al., "Deep Learning for Component Fault Detection in Electricity Transmission Lines," *Journal of Big Data*, vol. 9, no. 1, pp. 1-34, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[2] Swarit Anand Singh, and K.A. Desai, "Automated Surface Defect Detection Framework using Machine Vision and Convolutional Neural Networks," *Journal of Intelligent Manufacturing*, vol. 34, no. 4, pp. 1995-2011, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] Qingxue Liu et al., "Research on Deep Learning-based Multi-Level Cross-Domain Foreign Object Detection in Power Transmission Lines," *Sensors*, vol. 25, no. 16, pp. 1-19, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[4] Gaoyi Zhu et al., "Combining Unsupervised Domain Adaptation and Semi-Supervised Learning for Power Line and Transmission Tower Segmentation," *IEEE Transactions on Instrumentation and Measurement*, vol. 74, pp. 1-17, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[5] Priyanka Khirwadkar Shukla, and K. Deepa, "Deep Learning Techniques for Transmission Line Fault Classification-A Comparative Study," *Ain Shams Engineering Journal*, vol. 15, no. 2, pp. 1-9, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[6] Baoyu Xu et al., "Development of Power Transmission Line Detection Technology based on Unmanned Aerial Vehicle Image Vision," *SN Applied Sciences*, vol. 5, no. 3, pp. 1-15, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Ziran Li et al., "Design and Application of a UAV Autonomous Inspection System for High-Voltage Power Transmission Lines," *Remote Sensing*, vol. 15, no. 3, pp. 1-24, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[8] Min Liu et al., "A Fast and Accurate Method of Power Line Intelligent Inspection based on Edge Computing," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1-12, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[9] Min He et al., "Bolt-YOLO: Research on an Algorithm Framework for Detecting Bolt Defects in Transmission Lines," *IEEE Transactions on Power Delivery*, vol. 40, no. 3, pp. 1718-1729, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[10] Shuai Hao, Jing Li, and Xu Ma, "MRA-YOLOv8: A Transmission Line Fault Detection Algorithm Integrating Multi-Scale Feature Fusion," *Sensors*, vol. 25, no. 24, pp. 1-13, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[11] E. Guangxun et al., "A Novel Attention Temporal Convolutional Network for Transmission Line Fault Diagnosis Via Comprehensive Feature Extraction," *Energies*, vol. 16, no. 20, pp. 1-21, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[12] Khalfan Al Kharusi, Abdelsalam El Haffar, and Mostefa Mesbah, "Fault Detection and Classification in Transmission Lines Connected to Inverter-based Generators using Machine Learning," *Energies*, vol. 15, no. 15, pp. 1-23, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[13] Arash Moradzadeh et al., "Hybrid CNN-LSTM Approaches for Identification of Type and Locations of Transmission Line Faults," *International Journal of Electrical Power and Energy Systems*, vol. 135, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[14] Jibin B. Thomas et al., "CNN-based Transformer Model for Fault Detection in Power System Networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1-10, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[15] Hyeyeon Choi et al., "Attention-based Multimodal Image Feature Fusion Module for Transmission Line Detection," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 11, pp. 7686-7695, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[16] Omer Kullu, and Eyup Cinar, "A Deep-Learning-based Multi-Modal Sensor Fusion Approach for Detection of Equipment Faults," *Machines*, vol. 10, no. 11, pp. 1-17, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[17] Shao Jia Li et al., "DF-YOLO: Highly Accurate Transmission Line Foreign Object Detection Algorithm," *IEEE Access*, vol. 11, pp. 108398-108406, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[18] Hongbin Sun et al., "Power Transmission Lines Foreign Object Intrusion Detection Method for Drone Aerial Images based on Improved Yolov8 Network," *Drones*, vol. 8, no. 8, pp. 1-26, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[19] Pengyu Zhang et al., "Multi-Scale Feature Enhanced Domain Adaptive Object Detection for Power Transmission Line Inspection," *IEEE Access*, vol. 8, pp. 182105-182116, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[20] Yaru Wang et al., "Cross-Domain Multi-Level Feature Adaptive Alignment R-CNN for Insulator Defect Detection in Transmission Lines," *IEEE Transactions on Instrumentation and Measurement*, vol. 74, pp. 1-12, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[21] Yeqin Shao et al., "TL-YOLO: Foreign-Object Detection on Power Transmission Line based on Improved Yolov8," *Electronics*, vol. 13, no. 8, pp. 1-18, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[22] Stefano Frizzo Stefenon et al., "Enhanced Insulator Fault Detection using Optimized Ensemble of Deep Learning Models based on Weighted Boxes Fusion," *International Journal of Electrical Power and Energy Systems*, vol. 168, pp. 1-16, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[23] Su Yan et al., "An Algorithm for Power Transmission Line Fault Detection based on Improved YOLOv4 Model," *Scientific Reports*, vol. 14, no. 1, pp. 1-8, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[24] Lincong Peng et al., "YOLOv7-CWFD for Real-Time Detection of Bolt Defects on Transmission Lines," *Scientific Reports*, vol. 15, no. 1, pp. 1-17, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[25] Jiangpeng Zheng et al., "GEB-YOLO: A Novel Algorithm for Enhanced and Efficient Detection of Foreign Objects in Power Transmission Lines," *Scientific Reports*, vol. 14, no. 1, pp. 1-13, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[26] Chuanyang Liu et al., "Insulator Faults Detection in Aerial Images from High-Voltage Transmission Lines based on Deep Learning Model," *Applied Sciences*, vol. 11, no. 10, pp. 1-20, 2021. [CrossRef] [Google Scholar] [Publisher Link]