Journal of Operation and Automation in Power Engineering

Vol. XX, No. XX, Dec. 2022, Pages: XXXX (Proofed)

http://joape.uma.ac.ir

Research Paper

Optimizing Fault Identification in Power Distribution Systems by the Combination of SVM and Deep Learning Models

Garima Tiwari * and *Sanju Saini*

Deenbandhu Chhotu Ram University of Science and Technology, Murthal, Haryana, India.

Abstract— *Maintaining electrical grid stability and reliability requires the rapid diagnosis and classification of faults in power distribution systems. This study presents a hybrid model that integrates deep learning with support vector machine (SVM) methodologies to classify distribution system faults. In the proposed approach, feature extraction is performed using a convolutional neural network (CNN), and an SVM classifier is employed to identify fault patterns and establish generic fault classifications. The hybrid model is trained and evaluated using an extensive dataset comprising power distribution system fault currents under various fault types and conditions. The integration of deep learning feature extraction with SVM classification enhances fault classification effectiveness. This study aims to contribute to the overall improvement of distribution system reliability, reduction of downtime, and more efficient grid management. To achieve this, PSCAD software is utilized to simulate faults and collect images of three-phase fault current data. Initially, the fault classification problem is addressed using four pre-trained CNN models, with the collected images serving as input data. The hybrid model consists of two distinct components: an SVM block, known for its efficient and precise data classification capabilities, and a CNN block, specifically designed for feature extraction. In the MATLAB environment, a combination of four pre-trained CNN models—AlexNet, SqueezeNet, GoogLeNet, and ResNet-18—are utilized in conjunction with an SVM to create hybrid models. The hybrid SqueezeNet-SVM model has demonstrated exceptional performance, achieving an accuracy rate of 99.95%, a precision rate of 99.98%, a sensitivity rate of 99.6%, and a specificity rate of 99.7%.*

Keywords—Convolutional Neural Network, Support Vector Machines, fault classification, distribution system sensitivity, specificity, kappa score.

1. INTRODUCTION

Short-circuit faults that result in power loss pose a constant threat to distribution systems. Therefore, accurately identifying and correcting faults as quickly as possible is critical to ensure rapid restoration. The application of fault diagnosis techniques from transmission grids to distribution grids is challenged by the structural complexities inherent in distribution grids, including heterogeneity and numerous laterals [1]. Distribution lines are vulnerable to severe weather and/or environmental conditions. Electrical faults disrupt electricity and reduce the effectiveness of power networks. Hence, one of the primary objectives of distribution lines is to minimize these effects as much as possible. A fault classification approach that is highly reliable, fast, and precise is required to ensure that end users are never without power [2, 3]. This need becomes more critical as the level of automation in power distribution systems is rapidly increasing. To address these changes, artificial intelligence methods must be employed to manage information from distribution systems [4].

In recent years, significant attention has been given to studying fault classification and location estimation methods in power transmission systems [5, 6]. Authors of the paper [7], propose a protection scheme for a six-phase transmission line using a Mono ANN Module and Multi ANN Modules. The scheme utilizes the Discrete Wavelet Transform for fault location estimation. Fault detection and classification in transmission lines can be performed quickly and accurately using an Extreme Learning Machine [8]. However, these methods cannot be directly applied to power distribution systems due to differences in connecting networks, grounding, protective relays, and other factors. In a distribution system, several physical factors—such as the voltages of different generators, the phase difference between any two generators, resistance during faults, fault inception angle, fault location, and the length of distribution lines—are measurable but vary significantly. Numerous studies have employed machine learning-based fault diagnosis approaches, utilizing data from various contexts to address the uncertainties inherent in distribution systems [9]. The study in [10] presents a method to identify various types of faults in networks of distributed underground cables.

1.1. Motivation

The aim of this study is to improve the detection and identification of various faults in power distribution systems, which are essential for ensuring a reliable energy supply. System failures and power outages can cause significant disruptions, making it crucial to identify these problems quickly and accurately. This research combines two advanced technologies—deep learning models and Support-Vector-Machines (SVM)—to achieve this goal. SVM is a proven technique for data classification, particularly

Received: 30 Oct. 2023
Revised: 19 Aug. 2024
Accepted: 20 Aug. 2024
*Corresponding author:
E-mail: 18001902001garima@dcrustm.org (G. Tiwari)
DOI: 10.22098/joape.2024.13923.2072

effective with smaller datasets. On the other hand, deep learning models such as AlexNet, GoogleNet, SqueezeNet, and ResNet excel at identifying complex patterns and analyzing large volumes of data. By integrating these two approaches, the study aims to develop a more robust and reliable system for fault detection, thereby enhancing performance by reducing downtime and ensuring more consistent power distribution. This innovative combination leverages the strengths of both machine learning methods to enhance fault detection capabilities. Ultimately, this research seeks to improve the accuracy and efficiency of fault detection in power distribution systems by integrating traditional machine learning (SVM) with advanced deep learning techniques. Various challenges associated with the research problem of this study can be summarized as follows:

A. Lack of Data: A major challenge is the availability of labeled training and testing samples. Collecting accurate and sufficient fault data is difficult due to the infrequent occurrence of faults in power distribution systems. Limited data can impede the performance and accuracy of fault detection algorithms.

B. Complexity and Non-linearity: Power distribution systems are highly complex and nonlinear, comprising various interconnected components and exhibiting dynamic behaviour. So, traditional fault detection methods, which rely on simplified models and assumptions, become inappropriate.

C. Sensitivity to System Variations: Power distribution systems can experience significant variations in operating conditions, such as load fluctuations, voltage changes, and alterations in system topology. Existing fault detection methods may be sensitive to these variations, resulting in false alarms or missed detections.

D. Fault Localization: Precisely locating faults in power distribution systems can be challenging due to the distributed nature of the system, which includes numerous feeders, branches, and nodes. This complexity makes it difficult to pinpoint the exact fault location.

E. Computational Efficiency: Real-time fault detection and classification require algorithms that operate within strict time constraints. However, some existing methods may be computationally intensive, requiring substantial processing power or lengthy simulations, which can hinder the deployment of efficient and timely fault detection systems.

F. Adaptability to New Fault Types: Power distribution systems are prone to various fault types. Existing fault detection approaches may struggle to adapt to new or evolving fault types, necessitating manual reconfiguration or retraining of the system.

G. Robustness to Noisy Measurements: Measurement noise in power distribution systems can affect the accuracy of fault detection algorithms. Noise from sensors, communication systems, or environmental factors can introduce false readings and impact the reliability of fault diagnosis.

H. Scalability: As distribution systems expand and integrate renewable resources and smart grid technologies, their scale and complexity increase. Existing fault detection approaches may lack scalability, making it challenging to manage large-scale systems with an increasing number of components and data points.

1.2. Literature review

Addressing the challenges discussed in previous subsection requires advancements in fault detection techniques. Traditional methods can be time-consuming, which motivates the application of intelligent algorithms for rapid and accurate fault detection in subterranean cables. Researchers have explored various machine learning techniques for detecting and classifying faults in power distribution systems. These techniques include fuzzy logic-based methods [11], neuro-fuzzy approaches [12], Support Vector Machines (SVM) [13], Artificial Neural Networks (ANNs) [14], KNN-Bayesian method [15] and SVM combined with Principal Component Analysis (PCA) [16]. Some researchers have also integrated machine learning with signal processing techniques to enhance results. Examples include Feedforward Neural Networks (FFNN) combined with S-Transform, Adaptive Resonance Theory (ART) neural networks with time-time (T-T) transform, wavelet entropy with ANNs, fuzzy logic with Discrete Wavelet Transform (DWT) [17], and ANFIS with wavelet transform [18], among others [19, 20]. In [21], authors have proposed a quick fault diagnosis for distribution lines having dispersed generations. The study presented in [22] focuses on developing a fuzzy logic method that uses discrete wavelet transform to detect various faults in an imbalanced electrical power distribution system.

Recent advancements in deep learning have garnered significant attention in both academic and industrial sectors [23]. A notable development is effective use of Convolutional Neural Networks (CNNs) and transfer learning algorithms for image processing and recognition tasks [24, 25]. They are capable of learning mid- and high-level abstractions from raw data [26] and consist of layers such as convolutional, pooling, and Rectified Linear Unit (ReLU) layers. CNNs are extensively used in computer vision and offer high accuracy in image recognition [27]. Recently, CNN based methods have also been widely adopted for fault diagnosis in transmission lines [28–31] and power distribution systems [32–34]. Bayesian CNN is used for faulty line identification in [35].

While Convolutional Neural Networks (CNNs) offer several advantages over traditional neural networks, their training algorithms are similar to those of Back Propagation Neural Networks (BPNNs) and are based on the Empirical Risk Minimization Principle (ERMP). In [7], a method utilizing deep learning integrated with data pre-processing techniques is proposed. CNNs require large datasets and can be prone to overfitting [36]. To enhance the generalization capabilities of CNNs, Support Vector Machines (rooted in Statistical Learning Theory (SLT) and the principle of structural risk minimization) are often used in combination with CNNs. This approach has been applied to various tasks, including handwriting recognition, facial recognition, human action recognition, and fault detection [28]. Support Vector Machines (SVM) are designed to identify the optimal hyperplane that separates data points into distinct classes with the maximum margin [37]. They are effective in handling high-dimensional and complex datasets and are used in a variety of fields, including image classification, text classification, bioinformatics, fault classification, and finance [38, 39]. Table 1 summarizes the extensive research on fault classification in power systems. Following research gaps have been identified from this literature survey:

1) When CNNs are used alone, they may sometimes overfit, particularly when trained on small datasets due to their complex architectures. However, integrating SVMs can help mitigate this problem. SVMs balance model complexity with the ability to generalize to new, unseen data, thereby reducing the likelihood of overfitting.
2) In many previous studies, the ratio of training data to testing data for deep learning networks is set at 90%:10%. This high ratio can result in overfitting, causing the model to perform poorly under varying fault conditions.
3) Most researchers have not developed models capable of classifying all possible categories of single faults (both symmetrical and unsymmetrical faults).
4) Previous research often relied on signal processing techniques such as S-Transform, Wavelet Transform, and Principal Component Analysis (PCA) for feature extraction, which adds to the overall complexity of the models.

To address these research gaps, this paper presents a comprehensive hybrid deep learning-based system for fault classification. The innovative use of time-series fault current images with a hybrid CNN-SVM model aims to discover visual patterns for monitoring circuit states and identifying type of the fault.

Table 1. Literature survey for fault identification.

Ref No.	Year	Author	No. of faults	Part of power systems	Approach	Training/testing data ratio in (%)
[37]	2020	Ongwei. *et al.*	11	Distribution line	SVM	90:10
[28]	2020	Rai. *et al.*	10	Transmission line	CNN	90:10
[29]	2022	Nguyen *et al.*	10	Transmission line	CNN+WT	80:20
[7]	2023	Nien	11	Distribution Line	CNN+STFT	90:10
[30]	2023	Yangkui Xi	10	Transmission line	CNN+CWT+SHUFFLE	80:20
[34]	2024	Shengsoo	7	Distribution line	CNN	85:15

1.3. Novelty and contribution of the present work

In light of the identified research gaps, this paper makes the following key contributions:

1) Development of a hybrid deep learning model: The paper introduces a hybrid deep learning model capable of classifying both symmetrical and unsymmetrical faults (11 types in total) in a distribution system. 70% of the data is used for training and 30% is used for validation and testing.
2) Elimination of additional signal processing: The proposed model directly processes images of three-phase fault current time series data, removing the need for additional signal processing or feature extraction techniques.
3) Testing across multiple CNN architectures: The model is evaluated using four pre-trained CNN architectures—ResNet-18, AlexNet, GoogleNet, and SqueezeNet—and their classification accuracies are compared.
4) Model generalization: The model's generalization is demonstrated by training and testing it on data collected under various fault conditions, including different fault resistances, inception angles, and fault locations.

The novelty of this paper lies in its innovative integration of deep learning models with Support Vector Machines (SVMs) to enhance fault detection in power distribution systems. This approach grasps strengths of both methodologies, improving the accuracy and efficiency of fault detection—an essential factor in maintaining a reliable and continuous power supply. This combined method represents a relatively new approach in the context of power distribution systems and offers improvements over traditional single-model.

The remainder of this paper is organized as follows: Section 2 introduces the IEEE 13 Node test feeder and the various pre-trained CNN architectures. Section 3 details the methodology and the results and conclusions are presented in Sections 4 and 5, respectively, followed by a discussion of the challenges and limitations of the proposed approach and the conclusion.

2. THEORETICAL FRAMEWORK

This study involves modeling and simulating an IEEE radial topology (Fig. 1.) The feeder is equipped with a 4.16 kV voltage source. The PSCAD environment is used to generate data for training and testing a deep learning model.

2.1. Convolutional neural network

A Convolutional Neural Network (CNN), commonly referred to as a ConvNet, is a type of artificial neural network specifically designed for processing and analyzing visual data, such as images and videos. CNNs have proven to be highly effective in various computer vision tasks, including image classification, object detection, and facial recognition [40]. The typical process of a CNN involves feeding an image into the network, where it undergoes processing through multiple layers to extract distinctive features, eventually leading to an output, such as a classification result. During training, the network's parameters—specifically, weights and biases—are adjusted using backpropagation and optimization algorithms to minimize the difference between the network's predictions and the actual labels in the training data. CNNs are a form of supervised machine learning used to identify image features based on spatial correlations. These networks are commonly employed to analyze local relationships within data. Since CNNs can autonomously learn features, they are capable of producing accurate classifications even without extensive domain knowledge [41]. The basic CNN architecture, as shown in Fig. 2, comprises several layers, and more complex models may include additional layers to capture more intricate patterns.

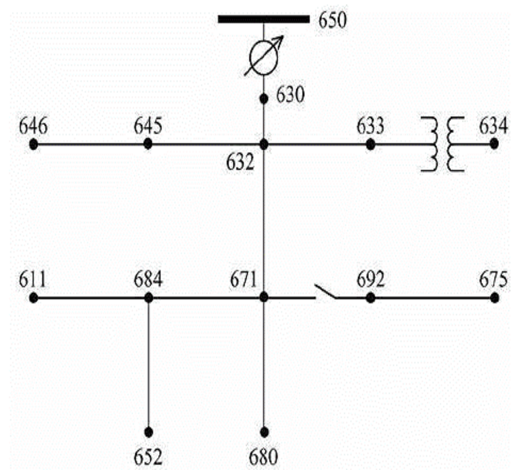


Fig. 1. IEEE 13-node test feeder.

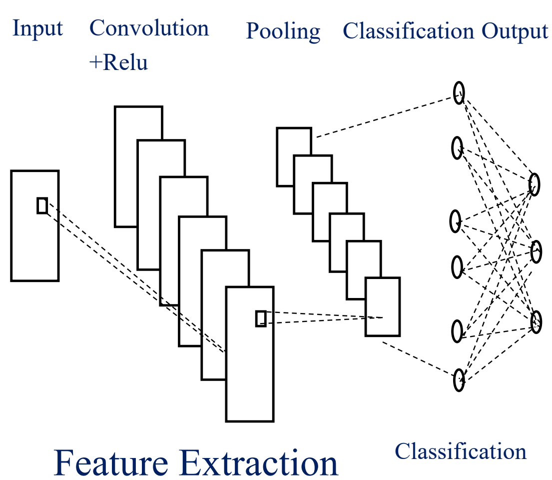


Fig. 2. Architecture of CNN.

Convolutional Neural Networks (CNNs) are primarily used in the field of deep learning for tasks like image recognition, object detection, and more. They consist of multiple layers, including convolutional layers, pooling layers, and fully connected layers. Key equations and mathematical models used for CNNs [42].

A) Convolution operation

Given an input image I and a filter F with dimensions m times n, the 'convolution operation' at a specific location (i, j) is calculated by using Eq. (1).

$$(I * F)\,(i.j) = \sum_{P=0}^{m-1} \sum_{q=0}^{n-1} I(i+p+q).F(p.q) \tag{1}$$

Where $I(i + p, j + q)$ represents the pixel values of the input image at location $(i + p, j + q)$.

B) Activation function

Typically, an activation function (like ReLU (Rectified Linear Unit)) is applied element-wise and is defined as Eq. (2). This function helps CNNs learn complex patterns by introducing non-linearities.

$$ReLU(x) = max(0, x) \tag{2}$$

C) Pooling operation

Pooling layers are used to reduce the spatial dimensions of the input, thus reducing the computational complexity and controlling overfitting. Max pooling is a commonly used pooling operation, where the maximum value within a window is selected. Given an input tensor 'A' and a pooling window of size p times q, the max pooling is represented as Eq. (3).

$$\mathrm{Max\,Pooling(A)}_{\mathrm{(i,j)=}} \max_{\substack{p-1 \\ m=0}} \max_{\substack{q-1 \\ n=0}} \mathrm{A}\;(\mathrm{i+p+q}) \tag{3}$$

D) Fully connected layer

After the convolutional and pooling layers, fully connected layers are employed for classification or regression tasks. In a fully connected layer, each neuron is connected to every neuron in the previous layer. For an input 'x' to a fully connected layer, with weights 'W' and biases 'b', the output 'z' of the fully connected layer is computed as shown in Eq. (4).

$$z = Wx + b \tag{4}$$

These are some of the fundamental equations and mathematical models used in CNNs. However, there are many variations and enhancements to CNN architectures, such as residual connections, batch normalization, and dropout, which further contribute to the complexity and effectiveness of these models. Transfer learning is a technique for efficiently transferring knowledge from one model to another. It is particularly useful for enhancing the performance of a model trained on a smaller dataset and is commonly applied for domain adaptation. This method leverages a pre-trained model to improve accuracy. The effectiveness of transfer learning varies depending on factors such as the compatibility of datasets, the size of the pre-trained model's original training dataset, and the available computational resources. The following subsections provide details on four distinct pre-trained CNN models used in this study.

2.2. AlexNet

The architectural design of AlexNet comprises a total of 25 layers, whereby three layers are fully linked and five levels are convolutional. The architecture makes use of the Rectified_Linear_Unit (ReLU) activation-function. The dimensions of the RGB input images utilized by AlexNet are 227 by 227 pixels, with a depth of 3 channels. As illustrated in Fig. 3, the first

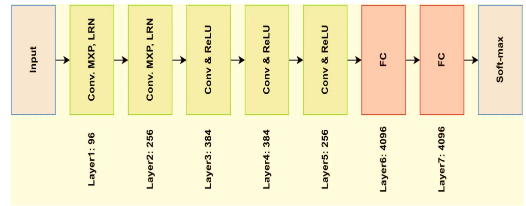


Fig. 3. Layer structure of AlexNet.

convolutional layer employs 96 different receptive filters of size 11×11 (LRN) to carry out convolution and maximum pooling.

To do a greater number of pooling operations, three filters are employed. In the second layer, a total of 55 filters are employed to execute identical operations. The third to fifth layer of convolutional employ feature maps size of 384, 384, and 296, respectively. The dropout technique is used in the model architecture by incorporating two fully linked layers followed by a Softmax layer [43].

2.3. GoogleNet

The GoogLeNet model, developed by Christian Szegedy at Google, was designed to reduce computational complexity. The proposed method entailed the integration of "Inception Layers" that encompassed diverse receptive fields achieved through the utilization of different kernel sizes, as illustrated in Fig. 4. The GoogLeNet architecture consists of a notable 22 layers, a rather large number. However, in comparison to the AlexNet model, which possesses a network parameter count of 60 million, Google utilizes substantially fewer parameters, specifically 7 million. In addition, it is worth noting that the computational requirements of GoogleNet were also found to be significantly lower than those of AlexNet, with a value of 1.53G MACs [44].

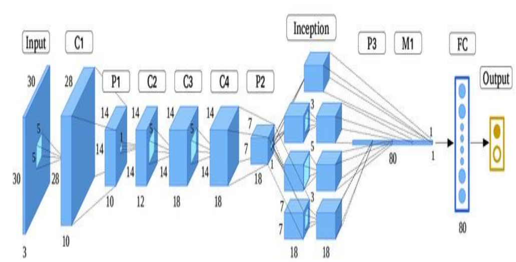


Fig. 4. Inception layer for GoogleNet with dimension reduction.

2.4. SqueezeNet

SqueezeNet v1.0 a convolutional neural network [45], despite having 50 times fewer parameters, is capable of attaining accuracy levels comparable to AlexNet on the ImageNet dataset. SqueezeNet is composed of 18 convolutional neural layers, as depicted in Fig. 5. After undergoing training, the network can categorize photos into a diverse range of 1000 object categories, encompassing various species. The network exhibits the capacity to acquire intricate feature representations for a diverse range of photographs. The SqueezeNet v1.1 networks exhibit a level of accuracy that is similar to that of the SqueezeNet v1.0 networks. However, they achieve this accuracy while demanding a reduced number of floating-point operations per prediction, as shown in Ref. [46].

2.5. ResNet-18

Among the ResNet (Residual Networks) models, ResNet-18 is a convolutional neural network design known for its compact

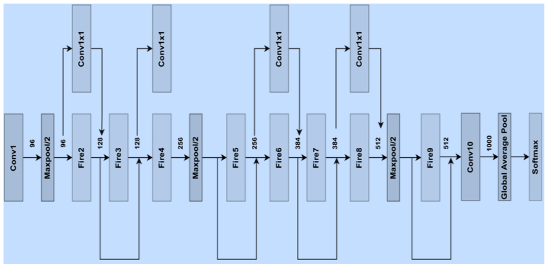


Fig. 5. SqueezeNet layer architecture [47].

architecture. As one of the smallest versions of ResNet, ResNet-18 is commonly used for image classification tasks. It introduces the concept of residual blocks, which include shortcut connections (also known as skip connections) that facilitate easier gradient flow during training. These shortcut connections enable the training of very deep neural networks and help mitigate the vanishing gradient problem.

ResNet-18 is popular for a variety of computer vision tasks, such as object detection, image segmentation, and image classification.Compared to larger ResNet variants like ResNet-50 or ResNet-101, ResNet-18 is more computationally efficient and better suited for applications with limited resources [48].

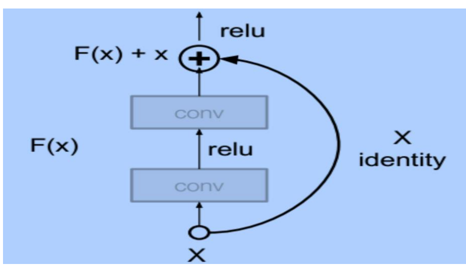


Fig. 6. Residual block [49].

The training of ResNet-18 involves backpropagation, which computes gradients of the loss function with respect to the network's weights and biases and updates them using an optimization algorithm like stochastic gradient descent. This network incorporates residual mapping ($H(x) = F(x)+x$) instead of the desired underlying mapping ($H(x)$). ResNet-50 is a variation with 50 layers, and Fig. 6 illustrates a residual block for the ResNet-18 architecture. ResNet-18's architecture consists of stacked 3x3 convolutional layers. It can train deeper networks without sacrificing performance and is easier to optimize [49].

2.6. Support Vector Machine

The Support Vector Machine (SVM) is a widely used supervised machine learning algorithm for classification and regression tasks. It is a non-parametric, discriminative learning method that aims to identify the optimal boundary (or hyperplane) that separates data points into distinct groups. The goal of SVM is to find the optimal hyperplane as shown in Fig. 7. The SVM algorithm selects the hyperplane that maximizes this margin, which enhances the model's ability to generalize to new, unseen data. SVMs can also handle non-linearly separable data by mapping it to a higher-dimensional feature space where it may become linearly separable. This is achieved through a technique known as the kernel trick, which allows for efficient computation without explicitly working in the high-dimensional space. SVMs are widely applied in various fields, including text classification, image classification, and bioinformatics.

They have also been designed to deal with problems involving several categories of categorization and regression analysis.

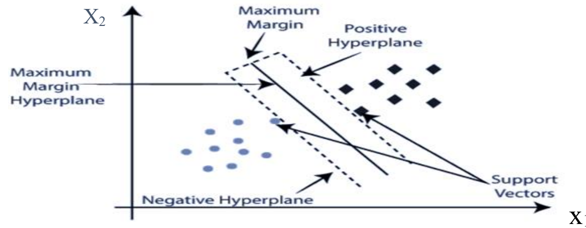


Fig. 7. Hyperplane of SVM.

Table 2. Different conditions for fault creation.

Parameter	Values
Faulty nodes	632, 633, 634, 650, 671, 675, 680, 692 nodes
Different inception angles	10*, 35*, 60*, 85*, 110*, 135*, and 185* degree
Resistances value	0, 0.5, 50, 100, 500, 1000, 1500 ohm

3. Methodology

The diagram in Fig. 8 shows the complete sequence of tasks. The process initiates with fault currents time series data collection using PSCAD simulation software. This data is subsequently imported into MATLAB, where images are generated to be utilized as input for convolutional neural networks (CNNs).

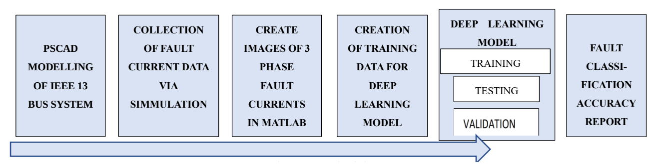


Fig. 8. Methodology.

The training dataset for CNNs consists of labeled data corresponding to various types of faults. The CNN/HYBRID-CNN model is used to train and assess fault classification accuracy based on the provided input-output data.

3.1. Fault current measurements and creating training/testing data

In any power distribution system, there are five distinct types of faults: L-G fault, L-L fault, L-L-G fault, L-L-L fault, and L-L-L-G fault. These fault categories cover a comprehensive range of 11 different failure scenarios when analyzing three lines. This study involves measuring and utilizing three-phase electrical currents under various fault conditions, as detailed in Table 2. A simulation model is created using the PSCAD/EMTDC framework to achieve this. The simulation runs for 1.3 seconds, with fault events occurring between 0.2 and 1 second. In total, 5,568 cases are recorded: 5,120 cases (8 nodes * 8 resistances * 8 inception angles) for each of the 10 fault types, and an additional 448 cases (8 nodes * 7 resistances * 8 inception angles) for the ABC fault. These cases include time series data for a three-phase current signal recorded during each fault.

The collected samples are then organized into vectors with 5568 rows, which serve as training/testing/validation datasets. Output training data is composed of a label that represents the type of fault in the input data.

3.2. Hybrid CNN-SVM model

This section provides a comprehensive discussion of the hybrid deep learning model employed in this study. The process consists of two main components. Models such as Alexnet, Googlenet, ResNet, or SqueezeNet are employed to extract deep feature maps. These maps are then converted into feature vectors and subsequently sent to the second phase, as depicted in Fig. 9. In the subsequent phase, 'SVM' model is responsible for partitioning the feature maps obtained in the preceding stage into distinct clusters, to facilitate the classification process. One rationale for employing these hybrid technologies is in their ability to operate effectively with computer specifications of moderate cost, in contrast to CNN models that necessitate high-priced computer specifications. In addition, hybrid methodologies exhibit expedited training of datasets and employ elementary calculations, whereas CNN models necessitate prolonged training periods for datasets and include intricate computations [50].

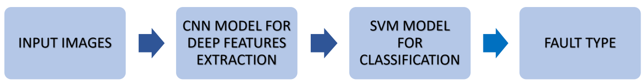


Fig. 9. Hybrid CNN-SVM model.

In summary, hybrid model involves using a deep neural network (DNN), such as a Convolutional Neural Network (CNN), for feature extraction and a Support Vector Machine (SVM) for classification. Here's a more detailed breakdown of the methodology:

A) Data preprocessing

a. Dataset collection:

A dataset comprising voltage and current signals under various fault and non-fault conditions in a power distribution system is collected. The dataset is designed to be diverse, covering different fault types, resistances, inception angles, and locations.

b. Data labeling:

The dataset is annotated to indicate the occurrence and characteristics of faults. This labeled data will be used for supervised training.

c. Data segmentation:

The voltage and current signals are divided into segments, taking into account the temporal aspects of the data.

B) Deep neural network (CNN) for feature extraction

a. Model architecture:

Four pre-trained CNN models—AlexNet, SqueezeNet, GoogleNet, and ResNet-18—are utilized in this work.

b. Training:

The CNN models are trained on the labeled dataset, enabling the network to extract discriminative features from the fault current signals. Categorical cross-entropy is selected as the loss function during training, and dropout regularization is applied to prevent overfitting.

C) Feature extraction and SVM classification

a. Feature extraction: The trained CNN is employed as a feature extractor. The output from one of the intermediate layers of the CNN is used as the feature vector for each signal segment.

b. Flattening or pooling features: Global pooling is applied to the extracted features to create a vector representation for each segment.

c. SVM classification: The flattened or pooled features are fed into an SVM for classification. The SVM is trained on these feature vectors, with labels indicating the fault types.

D) Model evaluation and validation

a. Dataset splitting:

The dataset is divided into training and testing sets, with a 70% to 30% split.

b. Performance metrics:

The performance of the combined model is evaluated using standard metrics such as accuracy, precision, kappa score, sensitivity, and specificity.

3.3. Performance evaluation metrics

The following Metrics are used to evaluate the performance of the proposed methodology:

A) Classification accuracy

Testing data is presented to examine the accuracy of the predictions by reviewing the labels. Accuracy is obtained according to Eq. (5).

$$Accuracy = \left(\frac{\text{Ncorrect}}{\text{Ntotal}}\right) * (100)\,\% \tag{5}$$

In this context, $N_{correct}$ represents the mixture no. of accurately classified cases, while N_{total} = total cases within 'test dataset'.

B) Kappa score

An additional performance parameter employed for evaluating the efficacy of our technique is the kappa score. The method's effectiveness is widely recognized as being supported by statistically credible evidence [51]. The kappa score is commonly measured on a scale from 1 to -1, where a value of 1 indicates flawless categorization. The kappa Scores in this study are computed using Eq. (6).

$$K = \frac{(Acc) - (Vacc)}{1 - (Vacc)} \tag{6}$$

In this case, random accuracy is shown by V_{acc}, while obtained accuracy is denoted by Acc. The calculation of random accuracy is shown in Eq. (7).

$$Vacc = 1/N \tag{7}$$

Here, 'N' represents the total classes

C) Precision

Precision is a statistical measure of how many true positives a classifier or model predicts out of all the positive guesses it makes. The value of precision is calculated by Eq. (8).

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} * 100 \tag{8}$$

D) Sensitivity

Sensitivity is a statistical variable that measures how many true positives there are in a set of positive cases. In other words, sensitivity is a measure of how well a model can pick out good cases. Value of sensitivity calculated by Eq. (9).

$$Sensitivity = \frac{\text{TP}}{\text{TP} + \text{FN}} * 100 \tag{9}$$

E) Specificity

A high specificity number means that the model can correctly find most of the wrong cases in the dataset and has a low rate of false positives. On the other hand, a low specificity number means that the model is making a lot of false positive predictions and isn't doing a good job of finding negative cases. The value of specificity is calculated by Eq. (10).

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} * 100 \tag{10}$$

Name of the CNN architecture/parameters	SqueezeNet	AlexNet	GoogleNet	ResNet-18
Size of image	227*227*3	227*227*3	224*224*3	224*224*3
Min. batch size	23	23	23	23
Max epoch	3750	3750	3750	3750
Initial learning rate	0.0003	0.0003	0.0003	0.0003
Validation frequency	250	250	250	250
Training time	87m7sec	109m1sec	71m3sec	65m31sec
Execution environment	CPU	CPU	CPU	CPU

Table 3a. Training parameters of CNN architectures

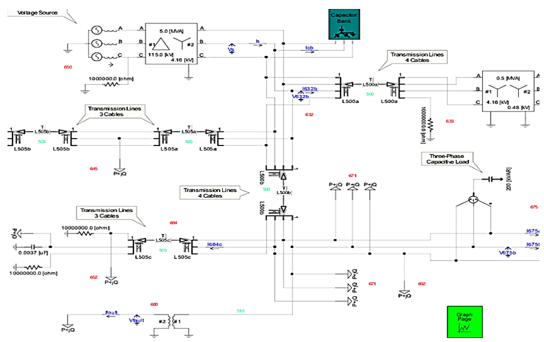


Fig. 10. PSCAD Simulink model of the IEEE 13 node radial distribution topology.

Parameter	Value
C parameter	1
gamma parameter (γ)	0.0999
The sigma parameter (δ_2)	0.6
Kernel function K (x, y)	RBF

Table 3b. Training parameters of CNN architectures

4. Simulation Results with Discussion

This research paper introduces a ‘PSCAD-Simulink-Model’ of IEEE 13-node radial topology, as shown in Fig. 10. The model is designed to simulate 11 different types of faults that can occur at eight specific nodes. The simulation platform can accurately replicate various fault categories, encompassing a wide range of fault resistance and inception angle values. The dataset used for training a Convolutional Neural Network (CNN) consists of visual representations depicting the sequential progression of three-phase fault currents. The images presented in this study were generated using MATLAB software version 2021a.

4.1. Splitting input dataset for training and testing

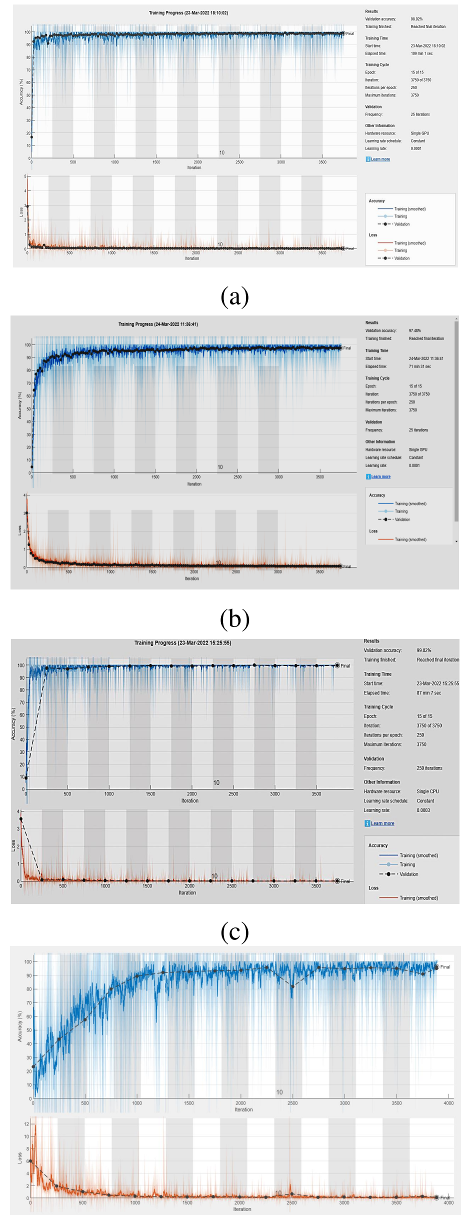
A data splitting strategy is employed where 75% of the available data is allocated for training, while the remaining 25% is reserved for testing and validation. A total of 5,568 incidents were observed across 11 distinct fault categories. Out of these, 5,120 cases involve 10 types of faults, determined by combinations of 10 resistance values, 8 nodes, and 8 inception angles. The remaining 448 cases pertain specifically to the ABC fault, defined by 7 resistance values, 8 nodes, and 8 inception angles. The comprehensive dataset includes fault category labels and 5,568 images depicting three-phase fault current data. Initially, the data is divided into a training set of 3,898 images, a validation set of 1,115 images, and a testing and prediction set of 555 images. Following this, a series of preprocessing operations—such as scaling, rotation, and reflection—are applied. Image scaling ensures that the input scale for the deep learning network remains consistent. To mitigate potential overfitting, random preprocessing techniques are employed to enhance the diversity of the visual input.

Table 4. Comparison of performance of AlexNet, GoogleNet, ResNet and SqueezeNet.

Name of the model	SqueezeNet	AlexNet	GoogleNet	ResNet
Accuracy	99.82%	98.92%	97.48%	96.24%
Kappa score	0.9981	0.9873	0.9745	0.961
Precision	99.85	99.2	97.90	97.05
Sensitivity	99.83	99.09	98	97.01
Specificity	99.84	99.10	98.01	97.03

Table 5. Performance results for hybrid CNN-SVM models.

Name of the model	SqueezeNet with SVM	AlexNet with SVM	GoogleNet with SVM	ResNet with SVM
Accuracy	99.95%	99.6%	98.9%	99.30%
Kappa score	0.998	0.995	0.987	0.99
Precision	99.981	97.2	99.0	98.9
Sensitivity	99.86	97.1	98.9	99.2
Specificity	99.87	97.15	98.75	99.25



(a) (b) (c) (d)

Fig. 11. Training progress: a) AlexNet, b) GoogleNet, c) SqueezeNet d) ResNet-18.

4.2. CNN & SVM training parameters’ selection

Training parameters of CNN & SVM can have direct effect on the performance of the proposed model. Values of these parameters is chosen in the proposed model via extensive hit and trial while considering their effect(s) on the performance of the model.

A) CNN training parameters

The performance of a deep learning model for fault classification can be significantly influenced by various training parameters as

Table 6. Comparison of accuracy with literature.

Reference no.	Year	Technique	Overall accuracy (%) (without noise)
Proposed technique	**2023**	**SqueezeNet+SVM**	**99.95%**
[34]	2024	CNN	99.46%
[30]	2023	CNN+CWT+SHUFFLE	99.90%
[31]	2023	CNN+STFT	99.36%
[23]	2023	SVM	99.08%
[20]	2022	Robust Semi-Supervised Prototypical Network (RSSPN)	91.10%
[15]	2021	KNN- Bayesian method	97%
[33]	2021	Deep CNN	99.3%
[35]	2021	CNN	72%
[37]	2020	SVM	99.52%
[27]	2018	2-D CNN	89%

discussed ahead.

1) **Size of image:** Larger images contain more detailed information, potentially enabling the model to learn more discriminative features. However, larger images also require more memory and computational resources for processing, which can increase training time and complexity.
2) **Min. batch size:** The batch size determines the number of samples processed before updating the model's parameters during training. A larger batch size can lead to more stable gradients and faster convergence but may require more memory. A smaller batch size introduces more noise into the gradient estimation but may help the model generalize better, especially with limited data.
3) **Max epochs:** The maximum number of epochs indicates how many times the entire training dataset is processed through the model during the training phase. To find the optimal number of epochs, it is common to monitor the model's performance on a separate validation dataset and halt training once the performance starts to decline."
4) **Initial learning rate:** The initial learning rate determines the size of the steps taken during gradient descent optimization. A higher learning rate can lead to faster convergence but may cause instability or overshooting. A lower learning rate may converge more slowly but could potentially find a better optimum and prevent divergence.
5) **Validation frequency:** More frequent 'validation' allows for early detection of overfitting or training issues.

Table 3a displays the training parameters of CNN models. The convolutional neural network (CNN) models utilized in this research encompass ResNet-18, AlexNet, GoogleNet, and SqueezeNet.

4.3. SVM training parameters

To develop an SVM model, several parameters must be specified. The key SVM training parameters and their effects on model performance are as follows:

1) **C parameter:** The C parameter controls the trade-off between achieving a high classification accuracy on the training data and maintaining a larger margin that separates different classes [52]. A higher C value imposes a greater penalty for errors, resulting in a narrower margin and better classification accuracy for all training points. Conversely, a lower C value allows for a larger margin, leading to a simpler decision function and potentially improved generalization.
2) **Gamma parameter (γ):** The gamma parameter controls the trade-off between the error reduction and the smoothness of the decision function. A high gamma value indicates that the model fits the training data closely, while a low gamma value suggests that the model may have better generalization capabilities by reducing the likelihood of overfitting.
3) **Sigma parameter (σ^2):** A higher sigma value results in a smoother and more flexible SVM decision function, which can better accommodate variations in the data.
4) **Kernel function K(x, y):** SVM uses a kernel function to map training data points into a higher-dimensional feature space using non-linear mapping. Four types of kernel functions can be used in SVM. The Radial Basis Function (RBF) kernel is often chosen as the optimal option due to its numerical stability and ability to handle non-linear relationships effectively. Table 3b displays the training parameters of SVM.

4.4. Training of deep learning model

Deep learning models using four types of pre-trained CNN models were trained for fault classification, with the training curves for each model shown in Figs. 11-(a-d). These training curves provide insights into the models' learning processes during training. They typically include metrics such as training loss, validation loss, training accuracy, and validation accuracy, all plotted against the number of training epochs. The training loss curve reflects the model's error on the training dataset as training progresses, with a decreasing training loss indicating effective learning and fitting to the training data. Similarly, an increasing training accuracy suggests that the model is improving its performance on the training data. In these charts, blue lines represent training accuracy, black lines represent validation accuracy, and orange lines represent loss. Table 3a, shown above, details all the parameters of the CNN architectures.

4.5. Results and discussion

In this section, the accuracy of fault identification achieved by the hybrid model is presented and compared with the accuracy of pre trained deep learning model acting alone.

A) Comparison of performance of AlexNet, GoogleNet, ResNet and SqueezeNet

Firstly, four pretrained deep learning models are used to identify fault types in the IEEE 13-bus test feeder. The confusion matrices for the test data are shown in Fig. 12-(a-d) for the AlexNet, GoogleNet, SqueezeNet, and ResNet-18 models. In these matrices, the classes labeled 'a' through 'k' represent specific types of short-circuit faults, categorized as follows: 'A to B', 'A to B to C', 'A to B to C to G', 'A to B to G', 'A to G', 'B to C', 'B to C to G', 'B to G', 'C to A', 'C to A to G', and 'C to G'. These confusion matrices represent the test data, comprising a total of 553 cases.

In Fig. 10-(c), which corresponds to the SqueezeNet CNN model, only one case is misclassified. The true class is 'c', but it is predicted as 'b'. Similarly, the confusion matrices for AlexNet, GoogleNet, and ResNet show 4, 8, and 7 misclassifications, respectively. Table 4 displays the 'Classification Accuracy', 'Precision', 'Sensitivity', 'Specificity', and 'Kappa Score' for the four types of transfer learning architectures of CNN.

B) Performance of CNN-SVM hybrid model

Using the CNN and SVM training parameters discussed in Subsection 4.2, the proposed model was trained on 70% of the total data. The performance matrix for the hybrid model, evaluated

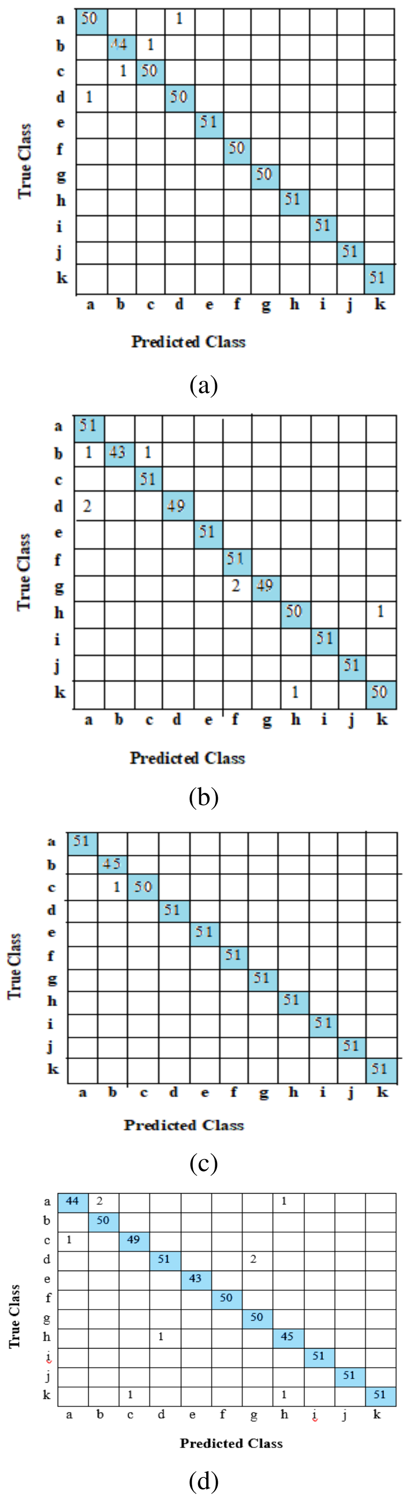


Fig. 12. CM (Confusion Matrices): a) AlexNet, b) GoogleNet, c) SqueezeNet d) ResNet-18.

on the test data (30% of the total data), is presented in Table 5. In this model, features extracted from the CNN model are used as input for the SVM classifier. The pre-trained CNN model SqueezeNet, when paired with the SVM classifier, achieved the best classification results, as shown in Table 5.

A comparison of Tables 4 and 5 indicates that this combined approach results in higher fault identification accuracy compared to using the CNN alone. Deep Learning models, especially Convolutional Neural Networks (CNNs) can capture data's intricate patterns and dependencies.This capability is essential for identifying subtle fault signatures in power distribution systems. These models can automatically learn hierarchical feature representations, which makes them powerful for tasks requiring complex pattern recognition. In the hybrid method, CNN effectively captures the complex patterns in the data, while the SVM provides robust classification, resulting in enhanced fault identification accuracy. The SVM's ability to handle high-dimensional spaces complements the Deep Learning model's capability to capture complex patterns, leading to a more powerful and effective fault identification system in power distribution networks. Moreover, SVM uses regularization techniques that help prevent overfitting, especially in high-dimensional spaces, essential for maintaining generalization capability. Table 6 gives a comparison of the accuracy of the suggested technique to those of other procedures used in the literature.

As shown in Table 6. It is self-evident that the proposed strategy is capable of diagnosing the faults in a distribution grid with maximum accuracy and is better than the other techniques presented in the literature. Additionally, this method has shown positive results regardless of the fault resistance magnitude or fault inception angle. measure and location of faulty node.

C) *Challenges and limitations in applying the proposed model for fault classification in real-world power distribution systems*

Despite being highly accurate, applying the proposed model for fault classification in real-world power distribution systems presents several challenges and limitations as follows:

1) **Data availability and quality:** Acquiring labeled fault data from real-world power distribution systems can be challenging due to privacy concerns, limited access to real fault data, and the high cost of collecting labeled datasets. The quality of the collected data may vary due to factors such as noise, sensor errors, and missing or incomplete labels.
2) **Imbalanced data:** Imbalanced data can bias model's learning process and result in poor performance, especially for minority fault classes.
3) **Generalization to new environments:** CNN-SVM model trained on one power distribution system may not generalize well to different distribution systems with distinct fault characteristics, topologies, and operating conditions. Adapting pre-trained models to new environments requires additional labeled data.
4) **Model complexity:** As the proposed model involves complex architecture and requires significant computational resources for training and inference, especially when dealing with large-scale datasets and high-resolution images. Deploying such model in real-time fault detection systems embedded within power distribution infrastructure could present challenges related to computational efficiency.
5) **Robustness to adversarial attacks:** Adversarial attacks are particularly concerning in critical systems like power distribution, where security and reliability are paramount.
6) **Integration with existing infrastructure:** Integrating CNN-SVM models into existing power distribution infrastructure and operational workflows may require significant changes to data acquisition systems, communication protocols, and decision-making processes. Compatibility with legacy systems and standards, as well as regulatory compliance, must be considered during integration efforts.

Addressing these challenges and limitations requires a multidisciplinary approach involving collaboration between power system engineers, machine learning researchers, data scientists, and domain experts.

5. Conclusions and Future Scope

The hybrid model that combines Convolutional Neural Network (CNN) and Support Vector Machine (SVM) techniques offers significant advantages for fault classification in power distribution systems by leveraging the complementary strengths of both methods. The proposed hybrid approach, which integrates deep learning (DL) with SVM, has demonstrated substantial improvements in fault classification accuracy compared to conventional techniques. This enhanced accuracy is crucial for the reliable operation of electricity distribution systems.

CNNs are particularly effective at automatically extracting relevant features from raw data, such as waveform signals or sensor measurements, thereby reducing the need for manual feature engineering. This enhances the system's ability to adapt to various types of faults. Moreover, the hybrid model has the advantage of generalizing well to previously unseen fault types and adapting to changing system conditions—a critical capability in real-world power distribution systems where fault patterns can evolve over time. The high classification accuracy of the proposed hybrid model enables optimized maintenance schedules, early anomaly detection, and reduced false alarms, leading to minimized operational disruptions, improved overall system reliability, and more efficient grid management.

This study employed AlexNet, GoogLeNet, ResNet-18, and SqueezeNet for fault classification in a distribution system. The hybrid model combining CNNs with SVM further improved the results. Specifically, the combination of the AlexNet architecture with SVM achieved 99.10% accuracy, while the highest accuracy, i.e., 99.95%, was achieved by using SqueezeNet in conjunction with SVM. Thus, this combination is considered the optimal choice for classifying fault types within the IEEE 13-bus radial topology.

In summary, this work represents a significant advancement in fault classification for power distribution systems by integrating deep learning and SVM models into a hybrid approach. The method reduces false alarms while enhancing accuracy, adaptability, and system dependability. To effectively implement the proposed model in real-world power distribution systems, careful data collection, continuous monitoring, and regular maintenance are essential. Future research could explore real-time fault detection and classification systems, the use of advanced sensor technologies, and experimenting with different combinations of deep learning and traditional machine learning algorithms.

References

[1] L. C. Acácio, P. A. Guaracy, T. O. Diniz, D. R. Araujo, and L. R. Araujo, "Evaluation of the impact of different neural network structure and data input on fault detection," in *2017 IEEE PES Innovative Smart Grid Technol. Conf.-Lat. Am.*, pp. 1–5, IEEE, 2017.

[2] M. Bjelić, B. Brković, M. Žarković, and T. Miljković, "Fault detection in a power transformer based on reverberation time," *Int. J. Electr. Power Energy Syst.*, vol. 137, p. 107825, 2022.

[3] M. Khoshbouy, A. Yazdaninejadi, and T. G. Bolandi, "Transmission line adaptive protection scheme: A new fault detection approach based on pilot superimposed impedance," *Int. J. Electr. Power Energy Syst.*, vol. 137, p. 107826, 2022.

[4] J. Zheng, P. Li, K. Xu, X. Kong, C. Wang, J. Lin, and C. Zhang, "A distance protection scheme for hvdc transmission lines based on the steady-state parameter model," *Int. J. Electr. Power Energy Syst.*, vol. 136, p. 107658, 2022.

[5] A. Prasad and J. B. Edward, "Application of wavelet technique for fault classification in transmission systems," *Procedia Comput. Sci.*, vol. 92, pp. 78–83, 2016.

[6] P. Rajaraman, N. Sundaravaradan, R. Meyur, M. J. B. Reddy, and D. Mohanta, "Fault classification in transmission lines using wavelet multiresolution analysis," *IEEE Potentials*, vol. 35, no. 1, pp. 38–44, 2016.

[7] G. Vikram Raju and N. Venkata Srikanth, "Mono ann module protection scheme and multi ann modules for fault location estimation for a six-phase transmission line using discrete wavelet transform," *J. Oper. Autom. Power Eng.*, vol. 12, no. 4, pp. 337–351, 2024.

[8] M. O. F. Goni, M. Nahiduzzaman, M. S. Anower, M. M. Rahman, M. R. Islam, M. Ahsan, J. Haider, and M. Shahjalal, "Fast and accurate fault detection and classification in transmission lines using extreme learning machine," *e-Prime Adv. Electr. Eng. Electron. Energy*, vol. 3, p. 100107, 2023.

[9] M. He and D. He, "Deep learning based approach for bearing fault diagnosis," *IEEE Trans. Ind. Appl.*, vol. 53, no. 3, pp. 3057–3065, 2017.

[10] G. Tiwari and S. Saini, "Estimation of location and fault types detection using sequence current components in distribution cable system using ann," in *Mobile Radio Commun. 5G Net. Proc. Second 2021*, pp. 119–128, Springer, 2022.

[11] A. Saber, A. Emam, and H. Elghazaly, "A backup protection technique for three-terminal multisection compound transmission lines," *IEEE Trans. Smart Grid*, vol. 9, no. 6, pp. 5653–5663, 2017.

[12] G. Tiwari and S. Saini, "Evaluation of transfer learning techniques for fault classification in radial distribution systems: A comparative study," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 11, no. 11, pp. 627–633, 2023.

[13] X. Deng, R. Yuan, Z. Xiao, T. Li, and K. L. L. Wang, "Fault location in loop distribution network using svm technology," *Int. J. Electr. Power Energy Syst.*, vol. 65, pp. 254–261, 2015.

[14] M. B. Hessine, H. Jouini, and S. Chebbi, "Fault detection and classification approaches in transmission lines using artificial neural networks," in *MELECON 2014-2014 17th IEEE Mediterr. Electrotech. Conf.*, pp. 515–519, IEEE, 2014.

[15] Y. Liang, K.-J. Li, Z. Ma, and W.-J. Lee, "Multilabel classification model for type recognition of single-phase-to-ground fault based on knn-bayesian method," *IEEE Trans. Ind. Appl.*, vol. 57, no. 2, pp. 1294–1302, 2021.

[16] P. Chopra and S. K. Yadav, "Pca and feature correlation for fault detection and classification," in *2015 IEEE Recent Adv. Intell. Comput. Syst.*, pp. 195–200, IEEE, 2015.

[17] A. A. M. Zin, M. Saini, M. W. Mustafa, A. R. Sultan, *et al.*, "New algorithm for detection and fault classification on parallel transmission line using dwt and bpnn based on clarke's transformation," *Neurocomputing*, vol. 168, pp. 983–993, 2015.

[18] G. Tiwari and S. Saini, "Radial power distribution system fault classification model based on anfis," *Int. J. Recent Innovation Trends Comput. Commun.*, vol. 11, no. 6, 2023.

[19] M. Shafiullah and M. A. Abido, "S-transform based ffnn approach for distribution grids fault detection and classification," *IEEE Access*, vol. 6, pp. 8080–8088, 2018.

[20] T. Zheng, Y. Liu, Y. Yan, S. Xiong, T. Lin, Y. Chen, Z. Wang, and X. Jiang, "Rsspn: robust semi-supervised prototypical network for fault root cause classification in power distribution systems," *IEEE Trans. Power Delivery*, vol. 37, no. 4, pp. 3282–3290, 2021.

[21] M. Dehghani, M. H. Khooban, and T. Niknam, "Fast fault detection and classification based on a combination of wavelet singular entropy theory and fuzzy logic in distribution lines in the presence of distributed generations," *Int. J. Electr. Power Energy Syst.*, vol. 78, pp. 455–462, 2016.

[22] M. Jamil, R. Singh, and S. K. Sharma, "Fault identification in electrical power distribution system using combined discrete wavelet transform and fuzzy logic," *J. Electr. Syste. Inf. Technol.*, vol. 2, no. 2, pp. 257–267, 2015.

[23] J. James, Y. Hou, A. Y. Lam, and V. O. Li, "Intelligent fault detection scheme for microgrids with wavelet-based deep neural networks," *IEEE Trans. Smart Grid*, vol. 10, no. 2, pp. 1694–1703, 2017.

[24] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nat.*, vol. 521, no. 7553, pp. 436–444, 2015.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[26] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, pp. 1–9, 2015.

[27] S. Senemmar and J. Zhang, "Deep learning-based fault detection, classification, and locating in shipboard power systems," in *2021 IEEE Electr. Ship Technol. Symp.*, pp. 1–6, IEEE, 2021.

[28] P. Rai, N. D. Londhe, and R. Raj, "Fault classification in power system distribution network integrated with distributed generators using cnn," *Electr. Power Syst. Res.*, vol. 192, p. 106914, 2021.

[29] N. N. Bon *et al.*, "Fault identification, classification, and location on transmission lines using combined machine learning methods.," *Int. J. Eng. Technol. Innovation*, vol. 12, no. 2, 2022.

[30] Y. Xi, W. Zhang, F. Zhou, X. Tang, Z. Li, X. Zeng, and P. Zhang, "Transmission line fault detection and classification based on sa-mobilenetv3," *Energy Rep.*, vol. 9, pp. 955–968, 2023.

[31] A. Moradzadeh, H. Teimourzadeh, B. Mohammadi-Ivatloo, and K. Pourhossein, "Hybrid cnn-lstm approaches for identification of type and locations of transmission line faults," *Int. J. Electr. Power Energy Syst.*, vol. 135, p. 107563, 2022.

[32] Y. Ma, D. Oslebo, A. Maqsood, and K. Corzine, "Dc fault detection and pulsed load monitoring using wavelet transform-fed lstm autoencoders," *IEEE J. Emerging Sel. Top. Power Electron.*, vol. 9, no. 6, pp. 7078–7087, 2020.

[33] Y. Ma, D. Oslebo, A. Maqsood, and K. Corzine, "Pulsed-power load monitoring for an all-electric ship: Utilizing the fourier transform data-driven deep learning approach," *IEEE Electrif. Mag.*, vol. 9, no. 1, pp. 25–35, 2021.

[34] M. N. I. Siddique, M. Shafiullah, S. Mekhilef, H. Pota, and M. Abido, "Fault classification and location of a pmu-equipped active distribution network using deep convolution neural network (cnn)," *Electr. Power Syst. Res.*, vol. 229, p. 110178, 2024.

[35] J. Zhu, L. Mu, D. Ma, and X. Zhang, "Faulty line identification method based on bayesian optimization for distribution network," *IEEE Access*, vol. 9, pp. 83175–83184, 2021.

[36] P. Venkata, V. Pandya, and A. Sant, "Data mining and svm based fault diagnostic analysis in modern power system using time and frequency series parameters calculated from full-cycle moving window," *J. Oper. Autom. Power Eng.*, vol. 12, no. 3, pp. 206–214, 2024.

[37] O. W. Chuan, N. F. Ab Aziz, Z. M. Yasin, N. A. Salim, and N. A. Wahab, "Fault classification in smart distribution network using support vector machine," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 18, no. 3, pp. 1148–1155, 2020.

[38] J. Li, Z. Weng, H. Xu, Z. Zhang, H. Miao, W. Chen, Z. Liu, X. Zhang, M. Wang, X. Xu, *et al.*, "Support vector machines (svm) classification of prostate cancer gleason score in central gland using multiparametric magnetic resonance images: A cross-validated study," *Eur. J. Radiol.*, vol. 98, pp. 61–67, 2018.

[39] F. Sultana, A. Sufian, and P. Dutta, "Advancements in image classification using convolutional neural network," in *2018 Fourth Int. Conf. Res. Comput. Intell. Commun. Net.*, pp. 122–129, IEEE, 2018.

[40] L. Ali, F. Alnajjar, H. A. Jassmi, M. Gocho, W. Khan, and M. A. Serhani, "Performance evaluation of deep cnn-based crack detection and localization techniques for concrete structures," *Sens.*, vol. 21, no. 5, p. 1688, 2021.

[41] R. Andonie and A.-C. Florea, "Weighted random search for cnn hyperparameter optimization," *arXiv Prepr. arXiv:2003.13300*, 2020.

[42] A. Ibrahim, F. Anayi, and M. Packianather, "New transfer learning approach based on cnn network for fault diagnosis," *Eng. Proc.*, vol. 4, 2022.

[43] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, B. C. Van Esesn, A. A. S. Awwal, and V. K. Asari, "The history began from alexnet: A comprehensive survey on deep learning approaches," *arXiv Prepr. arXiv:1803.01164*, 2018.

[44] K. Sudha and P. Sujatha, "A qualitative analysis of googlenet and alexnet for fabric defect detection," *Int. J. Recent Technol. Eng.*, vol. 8, no. 1, pp. 86–92, 2019.

[45] B. Liu, Z.-F. Hao, and X.-W. Yang, "Nesting support vector machinte for muti-classification [machinte read machine]," in *2005 Int. Conf. Mach. Learn. Cybern.*, vol. 7, pp. 4220–4225, IEEE, 2005.

[46] F. N. Iandola, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size," *arXiv Prepr. arXiv:1602.07360*, 2016.

[47] F. N. Iandola, A. E. Shaw, R. Krishna, and K. W. Keutzer, "Squeezebert: What can computer vision teach nlp about efficient neural networks?," *arXiv Prepr. arXiv:2006.11316*, 2020.

[48] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, pp. 770–778, 2016.

[49] M. Yin, X. Li, Y. Zhang, and S. Wang, "On the mathematical understanding of resnet with feynman path integral," *arXiv Prepr. arXiv:1904.07568*, 2019.

[50] Q. Wu, T. Gao, Z. Lai, and D. Li, "Hybrid svm-cnn classification technique for human–vehicle targets in an automotive lfmcw radar," *Sens.*, vol. 20, no. 12, p. 3504, 2020.

[51] F. Islam, K. Prakash, K. A. Mamun, A. Lallu, and H. R. Pota, "Aromatic network: A novel structure for power distribution system," *IEEE Access*, vol. 5, pp. 25236–25257, 2017.

[52] "The c parameter in support vector machines | baeldung on computer science." https://www.baeldung.com/cs/ml-svm-c-parameter. Accessed: 2024-08-04.