IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 12, 2019, accepted October 7, 2019, date of publication October 11, 2019, date of current version October 28, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2946662

A MobileNets Convolutional Neural Network for GIS Partial Discharge Pattern Recognition in the Ubiquitous Power Internet of Things Context: Optimization, Comparison, and Application

YANXIN WANG[1], JING YAN[1], QIFENG SUN[2], JUNYI LI[1], AND ZHOU YANG[3]
[1]State Key Laboratory of Electrical Insulation and Power Equipment, Xi'an Jiaotong University, Xi'an 710049, China
[2]School of Foreign Studies, Xi'an Jiaotong University, Xi'an 710049, China
[3]School of Computer Science, Xi'an Jiaotong University, Xi'an 710049, China

Corresponding author: Jing Yan (yanjing@mail.xjtu.edu.cn)

ABSTRACT With the construction and promotion of the Ubiquitous Power Internet of Things (UPIoT), it is an increasingly urgent challenge to comprehensively improve the recognition accuracy of the gas-insulated switchgear (GIS) partial discharge (PD), and to incorporate the model into UPIoT intelligent terminals supported by edge computing in embedded systems. Therefore, this paper proposes a novel MobileNets convolutional neural network (MCNN) model to identify the GIS PD patterns. We first construct the PD pattern recognition classification datasets by means of experiments and FDTD simulation, and also preprocess images via binarization processing. After constructing the MCNN model, depthwise separable convolutions and an inverse residual structure are adopted to deal with the vanishing gradient of the deep convolutional neural network (DCNN) in the GIS PD pattern recognition process. Then, through the graphics standardization process, the MCNN model is trained and tested. The whole training process is visualized by Tensorboard. Compared with other deep learning models and traditional machine learning methods, MCNN particularly stands out in recognition accuracy and time consumption with a 96.5% overall recognition rate and merely 7.3 seconds in training time. This research explores how to optimize the model by improving the recognition accuracy, and by reducing its computing load, storage space and energy consumption for better incorporation into intelligent terminals in the UPIoT context.

INDEX TERMS Gas-insulated switchgear, mobilenets convolutional neural network model, partial discharge, pattern recognition, ubiquitous power Internet of Things.

I. INTRODUCTION

The ubiquitous power Internet of Things (UPIoT), proposed in January, 2019, is an important concept in Chinese grid industry, which has been exerting broad impact on power equipment operation, power quality, and power supply [1], [2]. The construction of the UPIoT is largely based on the perception layer where each device-level intelligent terminal is supported by edge computing [3]. As the core of the perception layer, online monitoring technology plays a significant role in equipment interconnection, service interoperability, and safe and reliable operation of the power grid [4], [5]. Therefore, in view of the vulnerability of the gas-insulated switchgear (GIS) in a power system, determining how to make full use of ultra-high frequency (UHF) methods for online monitoring is of great necessity to avoid potential losses. This would to a large extent effectively realize fault diagnosis and pattern recognition. Currently, there are two methods of UHF analysis, namely the Time Resolved Partial Discharge (TRPD) mode and the Phase Resolved Partial Discharge (PRPD) mode.[6]. With detailed time information, the former is more suitable for a UPIoT context, despite the latter possessing much more clear and obvious waveform features.

The associate editor coordinating the review of this manuscript and approving it for publication was Canbing Li.

Due to the randomness of partial discharge (PD), a large number of machine learning methods for pattern detection and classification have emerged, such as support vector machines (SVM), decision trees (DT), random forests, and neural networks [7]–[9]. These methods depend largely on feature engineering, so the quality of features directly affects their effectiveness in PD recognition. The existing feature construction methods mainly include Fourier transforms, wavelet transforms, empirical mode decomposition, S-parameter transformation, and polar coordinate transformation [10]–[13]. Furthermore, in order to effectively extract the most critical characteristic parameters of the PD and reduce the feature dimension, principal component analysis (PCA) and an auto-encoder have been introduced for GIS PD recognition and classification [14], [15].

Features, however, are non-migratory: a certain set of features is only valid for a particular classification method. In order to address this problem, deep learning methods have been introduced for the GIS pattern recognition and fault diagnosis, such as the Lenet5 model, Alexnet model, one-dimensional convolutional neural network model, and LSTM-based deep recurrent neural network, which have proven to be of great use [16]–[20].

The Lenet5 model has a higher recognition accuracy on PRPD. On the TRPD waveform, however, it has an unsatisfactory performance due to excessive loss of input signal and an insufficient feature extraction. The Alenet model achieves excellent results in TRPD mode through fusion decision. But it is confronted with the vanishing gradient risk on account of training from 0, and the fusion decision also enlarges the model storage and calculation amount. The one-dimensional convolutional neural network has an improvement in recognition accuracy compared with two-dimensional convolution. Nevertheless, the time is increased due to complex signal processing and gradient vanishing. The LSTM-based deep recurrent neural network method does well in the PRPD mode, whereas it is subject to the sampling rate and feature rules, which may lead to over-reliance on expert experience. Therefore, new deep learning methods are needed to improve the recognition accuracy and real-time fault processing of GIS PD pattern recognition.

A novel adversarial learning framework was proposed to make the feature representation robust, boost the generalization ability of the trained model, and avoid overfitting with a small size of labeled samples for intelligent diagnosis of mechanical faults [21]. A modified deep convolutional neural network through adjusting two set of parameters in the fully connected network was also proposed for abnormal brain image classification, achieving excellent recognition results [22]. Lastly, deep belief networks through parallel computing were proposed to slave computing nodes to learn the features of their respective subdatasets and transmit them to the master computing node achieving good results in traffic flow prediction [23].

As a novel deep learning method, the MobileNets Convolutional Neural Network (MCNN), a novel deep learning method, was proposed by Google in 2018 has a broad classification in pattern recognition [24]. It effectively solves the problem that prevented training due to the vanishing gradient during the training process, reduces the model parameters and the storage space, and improves the accuracy of model recognition. Thus far, MCNN has not been applied to classifying GIS PD pattern recognition. At the same time, considering the growing demand for UPIoT construction in the smart grid, embedding the model into mobile terminals and intelligent terminals composed of Linux systems is still an urgent challenge. Hence, in this paper, a MCNN-based method is proposed, which can not only immensely reduce the model's computation burden, storage space and energy consumption for better incorporation into intelligent terminals, but also improve the recognition accuracy and solve the vanishing gradient problem. The main contributions of this paper are as follows:

(1) A MCNN-based method for GIS PD pattern recognition under the TRPD mode is proposed. It solves the problem of the vanishing gradient when many layers are involved. The recognition accuracy is also highly and effectively improved by maximizing the random simulation of PD with the experimental data and simulation data together.

(2) The GIS PD pattern recognition method proposed in this paper is significantly improved compared with other methods in a large number of training samples. The shortest testing time guarantees real-time and fast processing of signals under UPIoT. Simultaneously, depthwise separable convolutions and reverse residual structure are adopted to greatly reduce the computation load, storage space, and energy consumption of the model. Since it can be run on the CPU, it is very suitable to be embedded into a UPIoT intelligent terminal.

(3) Tensorboard can visualize the whole model training process, which addresses the "black box" problem of the neural network model. Image preprocessing technology, such as binary processing, can effectively tackle the problem of large data transmission and huge storage under the UPIoT circumstance, thereby shortening the model training and testing time.

II. DATA ACQUISITION

A. GIS PD EXPERIMENT

In order to effectively simulate the true conditions of GIS PD, in this paper we select four typical GIS PD defects for pattern recognition and fault classification: free metal particle defects, metal tip defects, floating electrode defects, and insulation void defects [25]. The simulation model of the four types of defects is shown in Fig. 1.

The main instruments and key parameters in the experiment are provided in Table 1. The schematic wiring diagram of the data acquisition experiment system is shown in Fig. 2. In the PD simulation experiment, the instruments in Table 1 are used for data acquisition, and the obtained GIS PD TRPD signals are shown in Fig. 3.

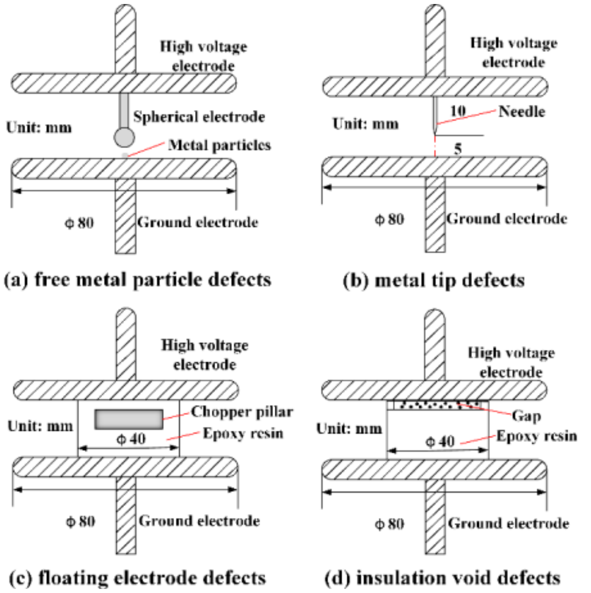


FIGURE 1. Typical PD defects model.

TABLE 1. Main equipment and key parameters in the experiment.

Instrument type	Key parameters
AC Test Transformer	Rated voltage is 250 kV, rated capacity is 50 kVA
UHF sensor	The detection frequency band is 300MHz-2000MHz, composed of an amplifier, a high-pass filter, a detector and a shielding case. The working bandwidth of the amplifier is 300MHz-1500Mhz, and the gain is 40dB.
Oscilloscope	The sampling rate is single channel 10GS/s, and the analog bandwidth is 2GHz.

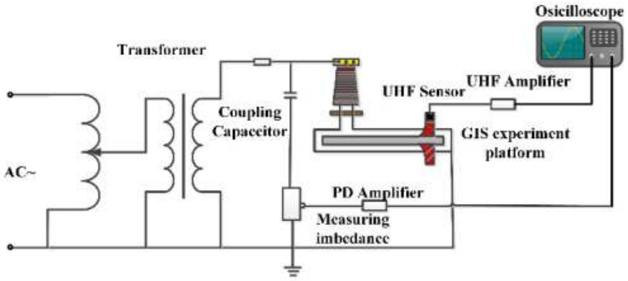


FIGURE 2. A schematic wiring diagram of the experiment system.

B. FDTD SIMULATION

In order to fully simulate the random GIS PD process and obtain sufficient data to verify the performance of the proposed MCNN model, we follow the practice of previous studies [26], [27], and make use of FDTD simulation software to simulate GIS PD. Guo *et al.* [28] and Li *et al.* [29] have both verified the feasibility of FDTD simulation.

The simulation model is shown in Fig. 4. Its center conductors and tanks are 120 mm and 400 mm in diameter, respectively, whereas the tank wall is 10 mm in thickness and 2.2 m in length. In the simulation, a wide-band Gaussian pulse signal with a -30dB attenuation at -3GHz and a continuous stationary frequency distribution is used as the local discharge current waveform, and its frequency distribution is changed by adjusting the waveform width parameter [30]. The boundary is set as the perfect matching layer (PML) to reduce the impact of the boundary structure on the simulation calculations. The attenuation of the wave energy caused by the structure is extremely slight, compared to the attenuation resulting from the heat loss during the electromagnetic wave propagation caused by the materials of the outer casing and the center conductor. The energy attenuation caused by the propagation structure itself is not considered in the simulation.

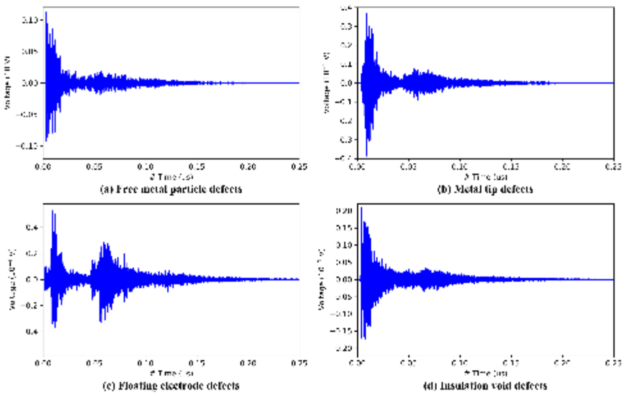


FIGURE 3. PD TRPD waveform.

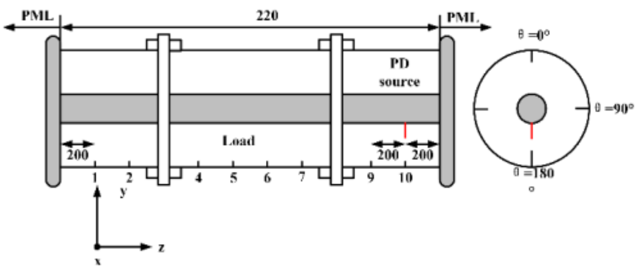
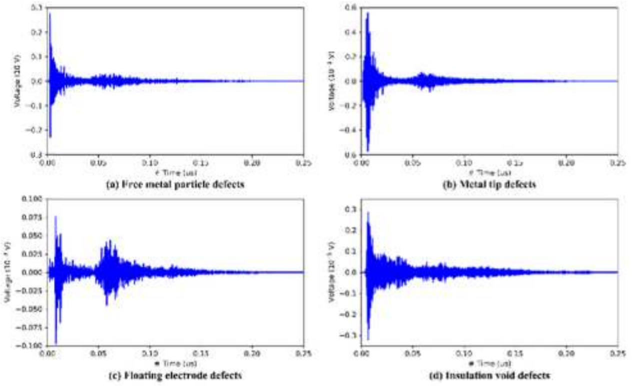


FIGURE 4. The GIS PD simulation model.

Considering the electromagnetic energy leakage caused by the discontinuity of the shielding at the basin insulator, a simplified insulator structure is set at 0.7 m away from the excitation source, with the material relative dielectric constant $\varepsilon_r = 4.0$. Moreover, in order to ignore the losses on the conductor walls, an ideal conductor is applied for the high-voltage conductors and cavities. The relative dielectric constant of SF_6 filled in the GIS cavity is 1.00205, and the density is set as 23.7273 kg/m^3 under a pressure of 0.4 MPa-abs. The relative magnetic permeability and electrical conductivity are 1 and 1.1015×10^{-5} S/m respectively. The highest frequency calculated is 3 GHz and the cell size is set as 10 mm × 10 mm × 10 mm [31]. The simulation time is 300 ns and the time step is $9.661674 \times 10^{-6}\mu$s. The FDTD simulation waveforms for the four defects are presented in Fig. 5.



(a) Free metal particle defects

(b) Metal tip defects

(c) Floating electrode defects

(d) Insulation void defects

FIGURE 5. FDTD simulation waveforms for the four defects.

III. METHOD

A. THE CONVOLUTIONAL NEURAL NETWORK MODEL

As a typical deep learning method, the convolutional neural network (CNN) has been rapidly developing in recent years and has become an efficient method for pattern recognition [32]–[34]. A CNN generally consists of a fully connected layer, a Softmax layer, and a feature extraction layer composed of a convolutional layer and a pooling layer. The convolution calculation in the feature extraction layer possesses the characteristics of sparse interaction, parameter sharing, and isomorphic representation [35]. Sparse interactions make the size of the convolution kernel much smaller than that of the input. Parameter sharing ensures that only one parameter set needs to be learned, which significantly reduces the storage requirements of the model and provides it with a translational equivariant feature. In addition, by using pooling to ensure that when the input varies a bit of the translation, the input representation is approximately unchanged. The deep convolutional neural network (DCNN) model is used as a pattern recognition classifier, which not only avoids the complicated process of artificial feature engineering and the problem of insufficient utilization of features, but can also effectively improve the accuracy and generalization ability of a diagnosis.

In each feature extraction layer, the feature map first convolutedly calculates with multiple convolution kernels, and then is connected to the next layer via bias calculation, activation function, and pooling operation. The operation of each feature extraction layer can be expressed as:

$$x_j^l = f(\sum_{i\in M_j} x_i^{l-1} * k_{ij}^l + b_j^l) \tag{1}$$

where $*$ is the convolution operation; x_i^{l-1} and x_j^l are respectively the input and output of the l-th layer; b_j^l is the bias; M_j is the j-th choice in the feature map; k_{ij}^l is the weight; and f is the element in the kernel function.

For convolutional calculations, the input feature map is calculated by convolution and then is output through the activation function, which can be expressed as:

$$M_j^l = \sigma(M_j^{l-1} * W_j^{l-1} + b_j^{l-1}) \tag{2}$$

where σ is the activation function; $*$ is the convolution operation; M_i^{l-1} and M_j^l are respectively the input and output of the l-th layer; W_j^{l-1} is the weight matrix of the j-th layer convolution kernel; and b_j^{l-1} is the offset vector of the j-th layer.

In the pooling layer, the calculation process can be expressed as:

$$M_j^l = pooling(M_j^{l-1}) \tag{3}$$

In the fully-connected layer, the feature maps of the previous layer are processed with the weighted sum method. The output feature map can be attained by the activation function, which can be expressed as:

$$Z_j^l = \sigma(Z_j^{l-1} \cdot W_j^{l-1} + b_j^{l-1}) \tag{4}$$

where $\cdot$ is the matrix multiplication operation; σ is the activation function; b_j^l is the bias; Z_i^{l-1} and Z_j^l are respectively the input and output of the l-th layer; W_j^{l-1} is the weight matrix of the j-th layer convolution kernel; and b_j^{l-1} is the offset vector of the j-th layer.

The training goal of convolutional neural network is to minimize the loss function. When used for classification problems, the loss function uses cross entropy, as shown in (5). When used for regression problems, the loss function uses the mean square error function, as shown in (6).

$$J(\Theta) = -\frac{1}{m}\sum_{i=1}^{m}[y^i \log \hat{y}^i + (1-y^i)\log\left(1-\hat{y}^i\right)] \tag{5}$$

$$J(\Theta) = \frac{\sum_{i=1}^{m}((y^i-\hat{y}^i)^2}{m} \tag{6}$$

where x^i is the i-th input; y^i is the true value of the i-th input; and $\hat{y}^i$ is the predicted value of the i-th input.

B. THE MCNN MODEL

The MCNN model is a lightweight deep neural network model proposed by Google for embedded devices [36]. Currently, there are two versions, MobileNet-V1 and MobileNet-V2. Their major modules are respectively shown in Fig. 6. MobileNet-V1 adopts the depthwise separable convolution method to largely reduce the redundancy of the convolution kernel compared to that of traditional 3D convolution. It not only optimizes the delay, and reduces the size of the model, but also improves the model's recognition accuracy [37]. Based on the structure of Mobile Net-V1, MobileNet-V2 is added with another two basic structures, the linear bottleneck layer as well as a reverse residual structure, which can accelerate convergence and prevent degradation. In view of the potential risk of a vanishing gradient in Mobile Net-V1, in this paper we use MobileNet-V2 to detect and classify GIS PD patterns.

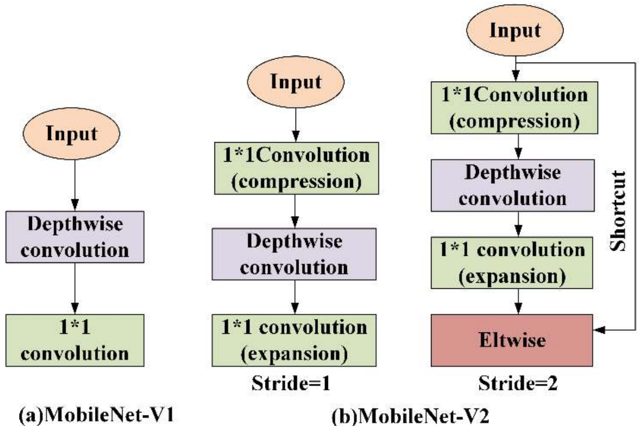


FIGURE 6. The structural module of MobileNet-V1 and MobileNet-V2.

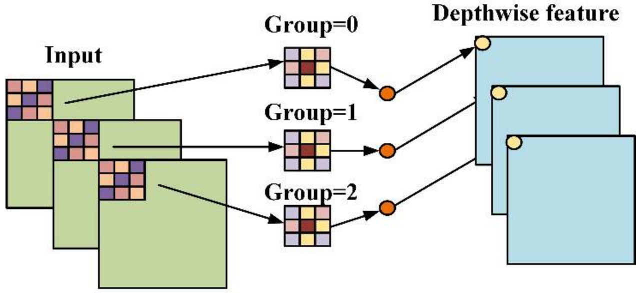


FIGURE 7. The calculation process of deep convolution.

1) DEPTHWISE SEPARABLE CONVOLUTION

The core layer of MCNN utilizes depthwise separable convolution instead of traditional standard convolution, which reduces the redundant representation of convolution kernels. The depthwise separable convolution can be decomposed into depthwise convolution and pointwise convolution.

Depthwise convolution filters the input channels without increasing the channel number. As is shown in Fig. 7, assuming that the input feature map is H × W × N and the convolution kernel space size is K × K; in order to convolve with each group of input feature data, the number of convolution kernel channels is 1; providing that the number of convolution kernels is N, and each set of feature data corresponds to a convolution kernel with respective K × K convolution. Therefore, the computational load of the depthwise separable convolution is $HWNK^2$, which is $1/M$ of the standard convolution. The amount of parameters is K^2N, which is $1/M$ of the standard convolution. The training time of MCNN is significantly reduced by ignoring the convolution in the channel dimensions, for to some extent depthwise convolution is similar to collecting the spatial characteristics of each channel separately.

Pointwise convolution is a special convolution operation with 1 × 1 convolution kernel in the spatial size. Specifically, it means that M sets of 1 × 1 × N standard convolution are performed on the input of H × W × N, where H × W is the input feature space size, N represents the channel number of input features and the convolution kernel, and M is the number of the convolution kernels. As is presented in Fig. 8, pointwise convolution is mainly used to change the output channel feature dimension. Pointwise convolution "mixes" information among channels, which can cope with "unsmooth information flow" caused by deep convolution and packet convolution.

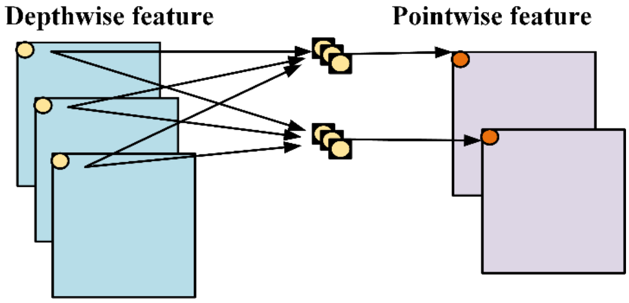


FIGURE 8. The calculating process of point-by-point convolution.

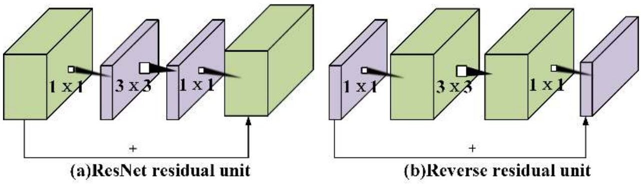


FIGURE 9. The structural module of MobileNet-V1 and MobileNet-V2.

2) REVERSE RESIDUAL STRUCTURE

The reverse residual structure is also a residual structure. Usually, the structure of the residual blocks reduces the feature map channel by the bottleneck layer.[38]. Since the shortcut is connected to the feature map with the reduced channel dimension, it is a reverse residual structure, as shown in Fig. 9. The reverse residual model in this paper firstly expands feature channel through the bottleneck layer; and then, transforms features via the deep convolutional layer; finally, compresses features through the bottleneck layer. In the reverse residual operation, the deep convolutional layer is only used to complete the feature nonlinear transformation, so the residual connection is added between the two bottleneck layers. But in general, the function of the residual connection in this structure is still the same as the residual blocks, which is used to speed up convergence and prevent degradation.

Suppose the channel dimension of the input feature is C, which includes all the useful information. The structure first expands the features into $t \cdot c$ dimensions (t is greater than or equal to 1), and the useful features will be distributed in the subspace of the $t \cdot c$ dimensional space; ReLU activation function can retain useful features; finally, to avoid information loss, the last activation function is removed, and features are only compressed by a linear bottleneck layer.

TABLE 2. **The overall structure of MobileNet-V2 model.**

Input	Operator	t	c	n	s
224 × 224 × 3	Conv2d	-	32	1	2
112 × 112 × 32	Bottleneck	1	16	1	1
112 × 112 × 16	Bottleneck	6	24	2	2
56 × 26 × 24	Bottleneck	6	32	3	2
28 × 28 × 32	Bottleneck	6	64	4	2
14 × 14 × 64	Bottleneck	6	96	3	1
14 × 14 × 96	Bottleneck	6	160	3	2
7 × 7 × 160	Bottleneck	6	320	1	1
7 × 7 × 320	Conv2d 1 × 1	-	1280	1	1
7 × 7 × 1280	Avgpool 7 × 7	-	-	1	-
1 × 1 × 1280	Conv2d 1 × 1	-	1000	-	-

TABLE 3. **The Realization of the Core Module Construction in MobileNet-V2.**

Input	Operator	Output
$H \times W \times N$	1 × 1conv2d, ReLU6	$H \times W \times tN$
$H \times W \times tN$	3 × 3dwise s = $s, ReLU6$	$H/s \times W/s \times tN$
$H/s \times W/s \times tN$	linear 1 × 1conv2d	$H/s \times W/s \times M$

For a feature of h × w in size, the expansion factor is t, the kernel size is k, the input channel is d_1, and the output channel is d_2. The total computation amount is:

$$Q = h \cdot \omega \cdot d_1 \cdot t(d_1 + k^2 + d_2) \qquad (7)$$

3) MCNN MODEL STRUCTURE

The overall structure of MobileNet-V2 model is shown in Table 2. Each row describes one or more sequences of the same layer. With being repeated *n* times, all layers sharing the same sequences have the same number of output channels *c*. The step size of the first layer of the sequence is *s* and the other layers' steps are 1. All spatial convolution kernel is 3 × 3 in size, and the expansion rate *t* is always applied to the input features described in Table 3. MobilNet-V2 contains the full convolutional layer with 32 convolution kernels, followed by 17 reverse residual bottleneck modules, with ReLU6 as a nonlinear activation function, for ReLU6 activation function is more suitable for low-precision calculations.[39]. Use 3 × 3 convolution kernels and use Dropout and Batch Normalization during training.

The Hyper-Parameter Width Factor α and the Resolution Factor β are used in MobileNet-V2 to further compress the model. Unlike MobileNet-V1, whose width factor value is less than or equal to 1, that of MobileNet-V2 is from 0.35 to 1.4. Except the last convolution layer, the width factor is applied to all layers, which can to a large extent improve the performance of small models [40].

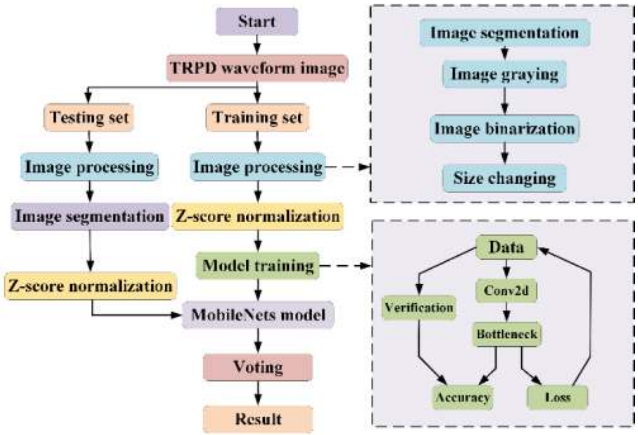


FIGURE 10. **The calculating process of point-by-point convolution.**

C. MCNN MODEL FOR GIS PD PATTERN RECOGNITION

In this paper we use the aforementioned MobileNet-V2 to identify and classify the GIS PD patterns. First, binarization processing is conducted on the TRPD three-channel map to obtain a single-channel binary image, which can reduce the model parameters, shorten the training time, and shrink the image from 600 × 438 to 224 × 224. In the model output layer, Softmax is used as a classifier, and one-hot encoding is used to identify four kinds of PD initial maps. All active functions in the model are ReLU6 functions.

In Fig. 10, we can see the clear PD pattern recognition process of MobileNet-V2 based MCNN model. Specific steps are as follows:

(1)Data preprocessing. Before training, the time domain map is converted into a single channel image by binarization, and is shrinked from 600 × 438 to 224 × 224. The inputs after preprocessing is shown in Fig. 11.

(2) Data enhancement. Data enhancement is used to randomly select 20% training data for image generation to improve the generalization ability of the model.

(3) Data standardization. The input data is normalized by Z-score standardization method to establish the comparability of data from different sources.

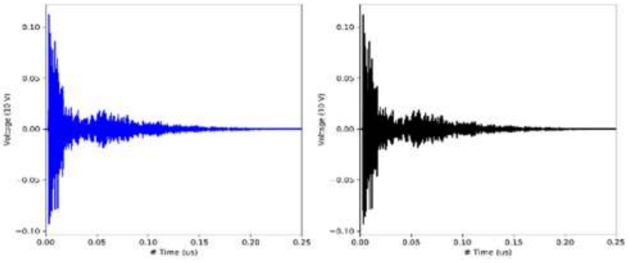
(4) Model training. Model training uses a backpropagation algorithm and a stochastic gradient descent algorithm, and also uses Dropout and Batch Normalization to improve training performance.

(5) Model testing. Model testing is to verify the generalization ability, fault recognition accuracy and testing time of the model.

(6) Model visualization. Tensorboard is used to visualize the model training process and the feature extraction process.

IV. RESULTS AND ANALYSIS

In this paper, We trained our model with Keras (Tensorflow backend) on a machine that has a GeForce RTX 2060 GPU, an Intel i7-8700 CPU and 16 gigabytes of RAM. In order to verify the performance of the proposed method,



(a) The original image

(b) The preprocessed image

FIGURE 11. The inputs after preprocessing.

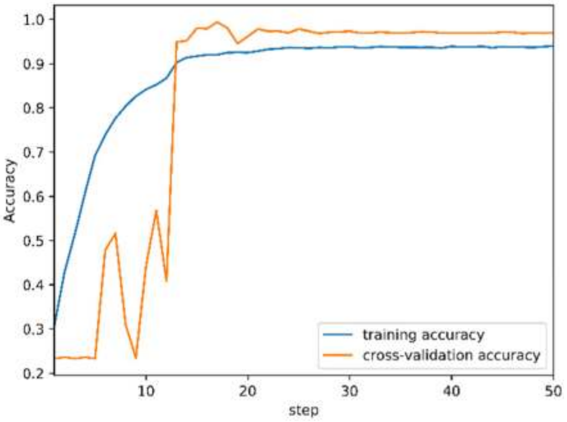


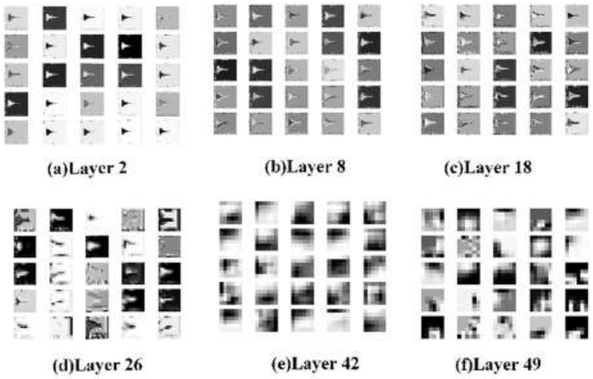
FIGURE 12. The training accuracy and verification accuracy curves during the model training.

we conduct GIS PD recognition by SVM, DT, BP neural network (BPNN), Lenet5 model, Alexnet model, VGG16 model, Resnet18 model and the MCNN model proposed in this paper. For the traditional machine learning methods, the max value, root mean square deviation, standard deviation, skewness, kurtosis, and the peak-to-peak value are designated as the feature parameters.[41].

A. MODEL TRAINING AND VISUALIZATION

In order to fully verify the feasibility of the proposed method, 80% of the 5000 datasets in this paper are selected for model training (70% for training, 10% for verification) and the remaining 20% for model testing. In order to visualize the training process and extracted features, Tensorboard is adopted for visualization. Fig. 12 shows the training accuracy and verification accuracy curves during the model training.

It can be seen from Fig. 12 that the training accuracy gradually increases and then levels off with the increase of the number of training steps, whereas the verification accuracy is on the increase in general, but decreases at some certain points. After the large fluctuation, the verification accuracy curve tends to level out. The model converges on training. Therefore, the MCNN model performs well in GIS PD pattern recognition. Since the model test accuracy has a tendency to stabilize after a slight decrease at around 18, the early stop technique is adopted in the training process to fully bring out the advantages of the model.



(a)Layer 2

(b)Layer 8

(c)Layer 18

(d)Layer 26

(e)Layer 42

(f)Layer 49

FIGURE 13. The extracted features of some layers of the MCNN model.

Since the extracted features cannot be directly observed, the CNN is also called a ‘‘black box’’ model. In order to address this problem, we uses Tensorboard to visualize the features extracted automatically by the MCNN model. The extracted features of some layers of the MCNN model are presented in Fig. 13. We visualize the operation results of different convolutional layers in the MCNN after extracting them. The specific process is as follows: the input is a planar two-dimensional image of the previous layer, and the output is a planar two-dimensional image after the convolution operation of the layer. After the multi-layer convolution operation, the original image becomes more and more blurred, which indicates that as the network layer deepens, the features extracted by our network structure are more abstract and high-level. Obviously, the convolution filter after learning is relatively smooth in space, indicating that training is sufficient. In addition, it can be seen from the feature visualization results that the features extracted by the convolutional neural network are sensitive to time. In the initial feature map, the contours of the waveform are mainly extracted, and then the specific combination of waveform features is formed. The smooth feature map further illustrates the feasibility of the model.

B. PATTERN RECOGNITION ACCURACY ANALYSIS

In order to verify the recognition accuracy of the model, respective 1250 sets of data for free metal particle defects (M type defects), metal tip defects (N type defects), air gap defects (P type defects) and floating electrode defects (O type defects) were selected for GIS PD pattern recognition. The pattern recognition results are shown in Table 4.

It can be seen from Table 4 that the overall recognition rate of the MCNN model is 96.5%, significantly higher than 86.5% of Resnet18, 83.4% of VGG16, 81.9% of Alexnet, 74.8% of Lenet5, 92.5% of SVM, 83.7% of BPNN, and 93.1% of DT. When training from 0, the VGG16 and Alexnet models have no change in training accuracy and loss curve after adjusting the hyper-parameters in the training process.

TABLE 4. GIS PD pattern recognition results.

Index	Model	Defect type				Overall (%)
		M(%)	N(%)	P(%)	O(%)	
1	MCNN	99.6	99.2	89.6	97.6	96.5
2	Resnet18	84.8	92.4	84.4	94.4	86.5
3	VGG16	82.4	86.6	77.2	87.4	83.4
4	Alexnet	81.6	84.2	78.0	83.6	81.9
5	Lenet5	80	82.49	64.4	72.4	74.8
6	SVM	95.	95.6	88.0	90.4	92.5
7	BPNN	85.6	84.2	78.4	86.6	83.7
8	DT	95.0	91.2	89.6	96.4	93.1

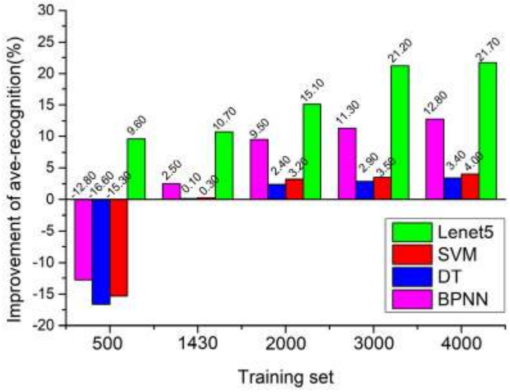


FIGURE 14. The improvement of the recognition accuracy of MCNN in different training samples compared to traditional machine learning algorithms.

The reason is that the vanishing gradient and the model cannot be trained, which prevents it from detecting GIS PD patterns. To this end, we use transfer learning to train VGG16 and Alexnet. The reason why the Lenet5 model has low recognition of partial discharge patterns is that model's feature learning is far from enough. It also results from the excessive loss of the image feature due to the 28 × 28 input size of the model. The shallow model depth also accounts for its failure to fully extract the GIS PD feature information. For traditional machine learning methods, the feature parameters result in their lower recognition rate, for after the feature dimension reduction, excessive manual interventions give rise to insufficient utilization of the GIS PD feature information.

Among all pattern recognition methods, the recognition rate of P type defects is relatively low. Because small gaps in the molding resin or voids in the layered regions between the insulating materials and the metal inserts tend to accumulate electric fields with time going by, which may further give rise to the instability of the PD[42]. MCNN recognizes more than 99% of M type and N type defects. Therefore, on the whole, MCNN is the best pattern recognition method among the above models. In order to further compare the accuracy of the MCNN, Lenet5 model and traditional machine learning methods in the recognition accuracy, Fig. 14 reports the improvement of the recognition accuracy of MCNN in different training samples compared to traditional machine learning algorithms.

We can see from Fig. 14 that when the training samples are 500, compared with Lenet5, SVM, DT, and BPNN, the improvement of recognition accuracy of the MCNN is respectively 9.6%, −15.3%, −16.6%, −12.8%. SVM and DT significantly outperform DCNN model in recognition accuracy, which demonstrates the evident advantages of traditional machine learning methods in small samples.

When the samples reach 1430, the recognition accuracy of the MCNN respectively improves by 20.7%, 0.3%, 0.1%, and 2.5%, which means the DCNN model has begun to demonstrate its advantages in recognition accuracy. When the data set reaches 4000, that of the MCNN respectively improves by 21.7%, 4%, 3.4%, and 12.8%. The MCNN significantly outperforms the traditional machine learning methods. With the increase of the training data sets, the deep learning method demonstrates growingly significant advantages over the traditional machine learning methods in recognition accuracy. Thus, the overall performance of MCNN is obviously better than any other traditional machine learning methods, which may possess promising prospect in big data and UPIoT context.

C. PARAMETERS COMPARISON AND TIME ANALYSIS

The parameters and training parameters storage consumption of the MCNN model, the Alexnet model, the VGG16 model, and the Resnet18 model are listed in Table 5. We can see from Table 5 that the MCNN model's parameter number is 2.24 million, which is about one-fiftieth of that of the VGG16 model, and about one-fifth of that of the Resnet18 model. It possesses the smallest number of parameters. In terms of the storage of training parameters, the MCNN model has the least memory footprint in all deep learning models with only 12.8 MB, significantly smaller than VGG16. Thereby, it can be easily ported to embedded systems, other mobile applications, and the UPIoT device level intelligent terminal.

The length of time spent in model training and testing directly determines whether the model can be applied under the UPIoT context. The long testing time cannot realize the in-time and quick processing under the online monitoring of the UPIoT. The long training time also negatively affects the updating ability of the model, which makes it difficult to update for higher accuracy when excessive training samples are acquired to form a larger historical knowledge database. In order to verify the time efficiency of MCNN model, we compare the training and testing time of MCNN and other machine learning methods. Fig. 15 shows the training time and testing time distribution of different models based on the 5000 TRPD datasets.

As can be seen from Fig. 15, the training time of MCNN, Resnet18, VGG16, Alexnet, Lenet5, SVM, DT, and BPNN

TABLE 5. The parameters and training parameters storage consumption of the models.

Model	Parameter(million)	Weight storage(MB)
MCNN	2.24	12.8
Resnet18	11.18	43.8
VGG16	107.01	418.1
Alexnet	24.72	96.6
Lenet5	14.43	56.4

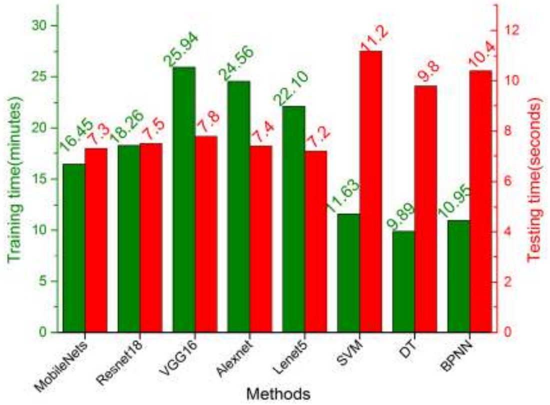


FIGURE 15. Training time and testing time distribution of different models based on 5000 TRPD datasets.

are 16.45 minutes, 18.26 minutes, 26.98 minutes, 25.45 minutes, 22.1 minutes, 11.63 minutes, 9.89 minutes, 10.95 minutes, respectively. MCNN's training time is significantly shorter than other deep learning models, but longer than traditional machine learning methods. The testing time of the above models are respectively 7.3 seconds, 7.5 seconds, 7.8 seconds, 7.4 seconds, 7.2 seconds, 11.2 seconds, 9.8 seconds, 10.4 seconds. Deep learning models spend significantly less testing time than traditional machine learning methods on account of the feature engineering in machine learning. Meanwhile, among all deep learning models, MCNN has the shortest testing time. In the actual engineering application, training does not occur frequently, whereas testing happens from time to time. Therefore, MCNN demonstrates the greatest advantage in time efficiency. Through the comparison of the parameter number, the model storage and the model time complexity, MCNN has achieved superior performance and is more suitable for embedded systems and mobile terminals.

V. CONCLUSION

In this paper we propose a MCNN based method for GIS PD pattern recognition, which not only fully utilizes the GIS PD information to improve the recognition accuracy rate, but also addresses the problem that some convolutional neural network models cannot be trained due to vanishing gradient after increasing the depth of layers. This model also to a large extent reduces the model parameters and storage space, and shortens the testing time. More importantly, it can be run on the CPU, so it can easily be ported and integrated into mobile and embedded systems. The specific conclusions are as follows:

1) By using MCNN model, we can address the problem that with the increase of model depth, the CNN models, such as VGG16 as well as Alexnet, cannot be trained due to the vanishing gradient problem. The problem of insufficient utilization of feature information can also be solved. MCNN model possesses 96.5% recognition accuracy, which is respectively 21.7%, 4%, 3.4% and 12.8% higher than that of Lenet5, SVM, DT, and BPNN.

2) MCNN has the superior performance. With depth-wise separable convolutions and reverse residual structure, the MCNN model has the shortest testing time with only 7.3 seconds. Moreover, compared with other deep learning methods, the model parameter number is 2.24 million and the storage space is 12.8 MB, which greatly reduces the parameter number, calculation amount, storage space and energy consumption. More importantly, it can be run on the CPU, so it can easily be ported and integrated into mobile and embedded systems.

3) By adopting Tensorboard, we can visualize the model training process and feature maps, which addresses the "black box" problem of the deep learning model. Meanwhile, the visualization also verifies the feasibility of the MCNN model for GIS PD pattern recognition.

REFERENCES

[1] X. Niu, S. Shoa, C. Xiu, J. Zhou, S. Guo, X. Chen, and F. Qi, "Workload allocation mechanism for minimum service delay in edge computing-based power Internet of Things," *IEEE Access*, vol. 7, pp. 83771–83784, 2019.

[2] W. Hu, W. Yao, Y. Hu, and H. Li, "Selection of cluster heads for wireless sensor network in ubiquitous power Internet of Things," *Int. J. Comput. Commun. Control*, vol. 14, no. 3, pp. 344–358, 2019.

[3] R. Piyare, "Internet of Things: Ubiquitous home control and monitoring system using Android based smart phone," *Int. J. Internet Things*, vol. 2, no. 1, pp. 5–11, 2013.

[4] S. Lanzisera, A. R. Weber, A. Liao, D. Pajak, and A. K. Meier, "Communicating power supplies: Bringing the Internet to the ubiquitous energy gateways of electronic devices," *IEEE Internet Things J.*, vol. 1, no. 2, pp. 153–160, Apr. 2014.

[5] M. Yun and B. Yuxin, "Research on the architecture and key technology of Internet of Things (IoT) applied on smart grid," in *Proc. Int. Conf. Adv. Energy Eng. (ICAEE)*, Jun. 2010, pp. 69–72.

[6] L. Li, J. Tang, and Y. Liu, "Partial discharge recognition in gas insulated switchgear based on multi-information fusion," *IEEE Trans. Dielectr. Electr. Insul.*, vol. 22, no. 2, pp. 1080–1087, Apr. 2015.

[7] Q. Khan, S. S. Refaat, H. Abu-Rub, and H. A. Toliyat, "Partial discharge detection and diagnosis in gas insulated switchgear: State of the art," *IEEE Elect. Insul. Mag.*, vol. 35, no. 4, pp. 16–33, Jul./Aug. 2019.

[8] R. Umamaheswari and R. Sarathi, "Identification of partial discharges in gas-insulated switchgear by ultra-high-frequency technique and classification by adopting multi-class support vector machines," *Electr. Power Compon. Syst.*, vol. 39, no. 14, pp. 1577–1595, 2011.

[9] Y.-B. Wang, D.-G. Chang, S.-R. Qin, Y.-H. Fan, H.-B. Mu, and G.-J. Zhang, "Separating multi-source partial discharge signals using linear prediction analysis and isolation forest algorithm," *IEEE Trans. Instrum. Meas.*, to be published.

[10] M. Wu, H. Cao, J. Cao, H.-L. Nguyen, J. B. Gomes, and S. P. Krishnaswamy, "An overview of state-of-the-art partial discharge analysis techniques for condition monitoring," *IEEE Elect. Insul. Mag.*, vol. 31, no. 6, pp. 22–35, Nov./Dec. 2015.

[11] Z. Ren, M. Dong, M. Ren, H.-B. Zhou, and J. Miao, "The study of partial discharge in GIS under impulse voltage based on time-frequency analysis," in *Proc. Int. Conf. Condition Monitor. Diagnosis (CMD)*, Sep. 2012, pp. 694–697.

[12] D. Dai, X. Wang, J. Long, M. Tian, G. Zhu, and J. Zhang, "Feature extraction of GIS partial discharge signal based on S-transform and singular value decomposition," *IET Sci., Meas. Technol.*, vol. 11, no. 2, pp. 186–193, Mar. 2017.

[13] J. Xue, X.-L. Zhang, W.-D. Qi, G.-Q. Huang, B. Niu, and J. Wang, "Research on a method for GIS partial discharge pattern recognition based on polar coordinate map," in *Proc. IEEE Int. Conf. High Voltage Eng. Appl. (ICHVE)*, Sep. 2016, pp. 1–4.

[14] Y. Khan, "Partial discharge pattern analysis using PCA and back-propagation artificial neural network for the estimation of size and position of metallic particle adhering to spacer in GIS," *Electr. Eng.*, vol. 98, no. 1, pp. 29–42, Mar. 2016.

[15] S. Li, Y. Man, C. Zhang, Q. Fang, S. Li, and M. Deng, "PRPD data analysis with auto-encoder network," in *Proc. Int. Conf. Water Res. Environ. Manag.* vol. 81, Jan. 2019, Art. no. 01019.

[16] S. Hui, J. Dai, G. Sheng, and X. Jiang, "Gis partial discharge pattern recognition via deep convolutional neural network under complex data source," *IEEE Trans. Dielectr. Electr. Insul.*, vol. 25, no. 2, pp. 678–685, Apr. 2018.

[17] G. Li, X. Wang, X. Li, A. Yang, and M. Rong, "Partial discharge recognition with a multi-resolution convolutional neural network," *Sensors*, vol. 18, no. 10, p. 3512, Oct. 2018.

[18] X. Wan, H. Song, L. Luo, Z. Li, G. Sheng, and X. Jiang, "Pattern recognition of partial discharge image based on one-dimensional convolutional neural network," in *Proc. Condition Monit. Diagnosis (CMD)*, Sep. 2018, pp. 1–4.

[19] M.-T. Nguyen, V.-H. Nguyen, S.-J. Yun, and Y.-H. Kim, "Recurrent neural network for partial discharge diagnosis in gas-insulated switchgear," *Energies*, vol. 11, no. 5, p. 1202, May 2018.

[20] S. Barrios, D. Buldain, M. P. Comech, I. Gilbert, and I. Orue, "Partial discharge classification using deep learning methods—Survey of recent progress," *Energies*, vol. 12, no. 13, p. 2485, Jun. 2019.

[21] T. Han, C. Liu, W. Yang, and D. Jiang, "A novel adversarial learning framework in deep convolutional neural network for intelligent diagnosis of mechanical faults," *Knowl.-Based Syst.*, vol. 165, pp. 474–487, Feb. 2019.

[22] D. J. Hemanth, J. Anitha, A. Naaji, O. Geman, D. E. Popescu, and L. H. Son, "A modified deep convolutional neural network for abnormal brain image classification," *IEEE Access*, vol. 7, pp. 4275–4283, 2018.

[23] L. Zhao, Y. Zhou, H. Lu, and H. Fujita, "Parallel computing method of deep belief networks and its application to traffic flow prediction," *Knowl.-Based Syst.*, vol. 163, pp. 972–987, Jan. 2019.

[24] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 4510–4520, 2018.

[25] N. Achatz, J. Gorablenkow, U. Schichler, B. Hampton, and J. Pearson, "Features and benefits of UHF partial discharge monitoring systems for GIS," in *Proc. Int. Sympos. Electr. Insul. Mater. (ISEIM)*, vol. 3, Jun. 2005, pp. 722–725.

[26] H. Tanaka, D. Tanahashi, Y. Baba, N. Nagaoka, H. Ohki, and M. Takeuchi, "Finite-difference time-domain simulation of partial discharges in a gas insulated switchgear," *High Volt.*, vol. 1, no. 1, pp. 52–56, Apr. 2016.

[27] T. Hoshino, S. Maruyama, and T. Sakakibara, "Simulation of propagating electromagnetic wave due to partial discharge in GIS using FDTD," *IEEE Trans. Power Del.*, vol. 24, no. 1, pp. 153–159, Jan. 2009.

[28] H. Guo, F. Lu, and K. F. Ren, "Simulation and measurement of PD-induced electromagnetic wave leakage in GIS with metal belt," *IEEE Trans. Dielectr. Electr. Insul.*, vol. 21, no. 4, pp. 1942–1949, Aug. 2014.

[29] X. Li, X. Wang, A. Yang, and M. Rong, "Partial discharge source localization in GIS based on image edge detection and support vector machine," *IEEE Trans. Power Del.*, vol. 34, no. 4, pp. 1795–1802, Aug. 2019.

[30] T. Yan, H. Zhan, S. Zheng, B. Liu, J. Wang, C. Li, and L. Deng, "Study on the propagation characteristics of partial discharge electromagnetic waves in 252kV GIS," in *Proc. Int. Conf. Condition Monitor. Diagnosis (CMD)*, Sep. 2012, pp. 685–689.

[31] T. Li, X. Wang, C. Zheng, D. Liu, and M. Rong, "Investigation on the placement effect of UHF sensor and propagation characteristics of PD-induced electromagnetic wave in GIS based on FDTD method," *IEEE Trans. Dielectr. Electr. Insul.*, vol. 21, no. 3, pp. 1015–1025, Jun. 2014.

[32] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss, and W. Wilson, "CNN architectures for large-scale audio classification," in *Proc. Int. Conf. Acoust. Speech Signal Process.*, Mar. 2017, pp. 131–135.

[33] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, Jan. 2013.

[34] D. Scherer, A. Müller, and S. Behnke, "Evaluation of pooling operations in convolutional architectures for object recognition," in *Proc. Int. Conf. Artif. Neural Netw.*, 2010, pp. 92–101.

[35] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Comput.*, vol. 15, no. 6, pp. 1373–1396, 2003.

[36] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," Apr. 2017. *arXiv:1704.04861*. [Online]. Available: https://arxiv.org/abs/1704.04861

[37] S. Kornblith, J. Shlens, and Q. V. Le, "Do better ImageNet models transfer better?" May 2018, *arXiv:1805.08974*. [Online]. Available: https://arxiv.org/abs/1805.08974

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

[39] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4700–4708.

[40] M. Tan, B. Chen, R. Pang, V. Vasudevan, M. Sandler, A. Howard, and Q. V. Le, "Mnasnet: Platform-aware neural architecture search for mobile," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 2820–2828.

[41] J. Tang, D. Wang, L. Fan, R. Zhuo, and X. Zhang, "Feature parameters extraction of GIS partial discharge signal with multifractal detrended fluctuation analysis," *IEEE Trans. Dielectr. Electr. Insul.*, vol. 22, no. 5, pp. 3037–3045, Oct. 2015.

[42] G. Ueta, J. Wada, S. Okabe, M. Miyashita, C. Nishida, and M. Kamei, "Insulation characteristics of epoxy insulator with internal void-shaped micro-defects," *IEEE Trans. Dielectrics Electr. Insul.*, vol. 20, no. 2, pp. 535–543, Apr. 2013.

YANXIN WANG was born in Chengde, Hebei, China, in 1995. He received the B.S. degree from Northeast Forestry University, in 2018. He is currently pursuing the M.S. degree in electrical engineering with Xi'an Jiaotong University, Xi'an, China. His current research interests include partial discharge detection and fault diagnosis for electrical equipment.

JING YAN was born in Tangshan, Hebei, China, in 1973. He received the B.S., M.S., and Ph.D. degrees in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 1996, 1999, and 2006, respectively, where he is currently with the School of Electrical Engineering. His current research interests include the theory and application of high voltage switchgear, intelligent electric apparatus, and power equipment condition evaluation.

QIFENG SUN was born in Shaoxing, Zhejiang, China, in 1996. He received the B.A. degree in English language and literature from Chongqing University, in 2018. He is currently pursuing the M.A. degree with Xi'an Jiaotong University, Xi'an, China. His current research interests include American literature and English for electrical engineering.

JUNYI LI was born in Chengdu, Sichuan, China, in 1995. He received the B.S. degree from Xi'an Jiaotong University, Xi'an, China, in 2017, where he is currently pursuing the M.S. degree in electrical engineering. His current research interests include partial discharge detection and fault diagnosis for electrical equipment.

ZHOU YANG was born in Xi'an, Shaanxi, China, in 1995. He received the B.S. degree in computer science from Xidian University, in 2018. He is currently pursuing the M.S. degree in computer science with Xi'an Jiaotong University, China. His current research interests include feature engineering and deep learning, especially the application domain of computer vision for business intelligence.