## RESEARCH ARTICLE

# An Automatic Identification Framework for Complex Power Quality Disturbances Based on Ensemble CNN

**MINGHAO WANG [ID], ZHUOFU DENG [ID], YUWEI ZHANG, AND ZHILIANG ZHU [ID]**
Software College, Northeastern University, Shenyang 110169, China
Corresponding author: Zhuofu Deng (dengzf@swc.neu.edu)

**ABSTRACT** A large number of electric vehicles (EVs) are connected to the grid, increasing the risk of power quality deterioration. Meanwhile, power quality disturbances (PQDs) directly affect EV charging safety. Intelligent identification of complex PQDs is the basis for solving the power quality problem, which is very meaningful for improving EV charging quality. This paper proposes an automatic recognition framework for complex PQDs based on ensemble convolution neural network (ECNN). Firstly, a multifusion structure on account of the time and frequency domain feature of PQDs signals is introduced. In addition, a composite convolution is proposed to reduce network complexity, which is using the standard convolution and depthwise separable convolution. Then, we design an adaptive-context mechanism to extend the versatility of ECNN. At the same time, the need to use batch normalization to accelerate training convergence and prevent training overfitting is verified. Furthermore, some visualization methods are performed to analyze the inner mode and illustrate the working mechanism of ECNN. Finally, we apply various experiments to prove the effectiveness of ECNN. Compared to other advanced deep neural networks and traditional methods, ECNN has better noise resistance, higher accuracy, and lower training cost.

**INDEX TERMS** Power quality disturbances (PQDs), ensemble convolutional neural network (ECNN), composite convolution, identification, visualization.

## I. INTRODUCTION

In recent years, with the deterioration of the global environment and the depletion of fossil energy, the electric vehicle (EV) industry has developed fast [1]. Meanwhile, the promotion and application of electric vehicle battery energy storage technology and charging facilities [2] have exacerbated the degree of voltage and current distortion in the power grid. In addition, under the influence of external conditions such as abnormal equipment operation and changes in the power grid structure, the overlapping phenomenon of power quality disturbances (PQDs) appears obviously [3] and the actual PQDs in different regions are often complicated. PQDs present a trend of complicacy and diversity, which poses serious challenges for the new energy vehicle industry.

The associate editor coordinating the review of this manuscript and approving it for publication was Nagesh Prabhu [ID].

Generally, PQDs include single and combined disturbances. Single PQDs include steady-state and transient phenomena such as voltage sag, flicker, harmonics, and notch. Common complex disturbances are dominated by transient interference overlap [4]. In recent years, a few researchers have explored the impact of PQDs on electric vehicle charging (EVC). For instance, reference [5] mentioned that voltage sag would reduce charging efficiency, improve battery temperature and shorten battery life. Reference [6] mentioned that harmonic may cause mechanical vibration of the motor, generate an additional loss, and increase the temperature of the equipment. Obviously, different types of PQDs bring different potential risks to the battery system and safe operation of electric vehicles, which seriously hinder the development of the new energy vehicle industry. Based on the identified PQDs, we can take more targeted measures to reduce the potential safety hazards posed by PQDs to EVC. It is observed that accurately identifying power

quality events in the distribution network is important for the development of the new energy vehicle industry. However, actual PQDs usually appear in combinations [7]. Therefore, an automatic identification technology for complex PQDs is urgently needed. In the former research, PQDs identification consisted of three steps: signal processing, feature selection, and classification.

At present, several signal analysis methods, such as Fourier Transform [8], S-Transform [9], Wavelet Transform [10], and Empirical Mode Decomposition [11] have been widely used to solve PQDs problems. These traditional approaches are committed to boosting time-frequency resolution in the process of signal feature extraction, but there are still inherent defects. For example, the time-frequency information of PQDs can be detected by Fourier Transform while it is unable to express its transient feature clearly by a fixed window size [12]. S-Transform uses a moving Gaussian window to analyze signals, but it is difficult to adapt to changes in all complex disturbances [13].

Feature selection is the key to achieving high classification accuracy. In literature [14], particle swarm optimization and extreme learning machine are combined to extract PQDs features. Literature [15] proposed a multi-scale sparse model and applied an orthogonal matching pursuit algorithm to obtain the sparse coefficients of each layer to form the sparse vector. It realized PQDs identification through a deep confidence network. Although the above studies have achieved good performance in power quality signal classification, they rely too much on experience and statistics to extract features, which makes the original feature set redundant or insufficient, and feature extraction becomes cumbersome and time-consuming. Currently, many classifiers have been used to process artificially selected features, such as Decision Tree [16], Support Vector Machine [17], and Artificial Neural Network [18]. These classifiers establish the mapping relationship from continuous features to discrete labels to identify various PQDs. However, these traditional methods have high recognition performance only for a single type of disturbance, but cannot effectively identify complex PQDs.

Due to the conventional signal identification process being limited to signal processing, feature selection, and classification, it is gradually difficult to deal with the increasingly complex PQDs efficiently. Therefore, many researchers have applied the new closed-loop analysis framework to classify complex PQDs, thus achieving automatic identification with high efficiency and performance.

With the excellent performance of deep learning in image feature extraction [19], many studies have pointed out that deep learning in PQDs identification has potential advantages [20]. Compared to artificial feature extraction, it not only simplifies the process but also improves the identification accuracy. Literature [21] transferred the input data of PQDs into 2-D images and then established an image classification model based on 2-D CNN. Although the feature map can be automatically extracted from signals without manual intervention, the input data is converted from 1-D time series

to 2-D images, which increases the training cost. Ref [22] used a hybrid structure combining CNN and Long Short-Term Memory (LSTM) to classify PQDs, but LSTM requires large computation and high cost. Reference [23] proposed a multifusion one-dimensional convolution neural network (MFCNN) framework, which strengthened feature selection capability by time domain and frequency domain fusion, but did not realize the automatic hyperparameter optimization, so it could not be adaptive to classify PQDs in different regions.

In view of the above-mentioned difficulties, the main contributions of this paper are as follows:

1) To automatically analyze complex PQDs, an ensemble convolutional neural network (ECNN) framework is proposed. Unlike general CNN, ECNN separates the raw signal into a high-frequency signal and low-frequency signal in the time domain based on Discrete Wavelet Transform (DWT), extracting frequency domain characteristics by Fast Fourier Transform (FFT).

2) Considering the complexity of actual PQDs in different regions, we designed an adaptive-context mechanism to improve the applicability of the model.

3) To reduce the complexity of ECNN and keep computational costs down on the basis of better feature diversity extraction, we proposed a composite convolution based on the depth separable convolution (DSC) and standard convolution (SAC).

4) We have conducted a lot of comparative experiments with advanced networks and traditional methods under different noises to verify the effectiveness of ECNN, including CNN-LSTM [22], DeepCNN [24], CNN-GRU [25], and MFCNN. In addition, we use a hardware platform for simulating real PQDs to verify the performance of measured data.

The remaining of this paper is structured as follows: Section II introduces the background of the proposed method. Section III presents the proposed ECNN, in which composite convolution and Bayesian optimization are embedded in the model. Section IV conducts visual analysis and many comparative experiments. Finally, Section V summarizes this paper roundly.

## II. DESCRIPTION OF METHODOLOGY
### A. DISCRETE WAVELET TRANSFORM
Wavelet transform (WT) is a time-frequency analysis method of signals, which has the characteristics of multi-resolution analysis and is suitable for analyzing and extracting the local feature of electrical signals. The approximate component and detail component are commonly used in wavelet transform, that is, the low-frequency information and high-frequency information of signals.

For a real function $\varphi(t)$, when its domain is compactly supported and both the mean and the higher moments are zero, $\varphi(t)$ is called the basis wavelet. The set of wavelet basis functions is constructed by translating and scaling $\varphi(t)$

$$\varphi_{m,n}(t)|\varphi_{m,n}(t) = m^{-\frac{1}{2}}\varphi(\frac{t-n}{m}), m > 0, m, n \in R \quad (1)$$

For $f(t) \in L^2(R)$, its WT is $f_{WT}(m, n) = < f, \varphi_{m,n} >$, its inverse transform is

$$f(t) = \frac{1}{A_\phi} \iint_0^{+\infty} f_{WT}(m, n)\phi_{m,n}(t) \, dm \, dn \qquad (2)$$

Among them,

$$A_\phi = \int \frac{|\varphi(w)|^2}{|w|} \mathrm{d}w, \, \varphi(w) = \int_{-\infty}^{+\infty} \phi(t)e^{-iwt} \, dt \qquad (3)$$

$w$ is frequency variable.

To minimize redundancy and ensure the integrity of the raw signal, DWT is developed. $m = m_0^r (m_0 > 0, r \in z)$ is selected and let $m$ be discrete, meanwhile, the translation factor $n$ is discretized uniformly, $n = pn_0(p \in z)$. $n_0$ must make $WT_n(r, p)$ construct $\varphi(t)$. $\varphi_{r,p}(t)$ can be expressed as

$$\varphi_{r,p}(t) = m_0^{-\frac{j}{2}} \varphi[m_0^{-r}(t - pm_0^r n_0)] = m_0^{-\frac{r}{2}} \varphi(m_0^{-r} - pn_0) \qquad (4)$$

its discrete wavelet transform is

$$f_{DWT}(m_0^r, pn_0) = \int_{-\infty}^{+\infty} f(t)\overline{\varphi_{m_0^r pn_0}}(t)dt \qquad (5)$$

The rebuild formula is as follows

$$f(t) = M \sum_{-\infty}^{\infty} \sum_{-\infty}^{\infty} f_{DWT}(r, p)\varphi_{r,p}(t) \qquad (6)$$

where M is a constant independent of the signal.

In this paper, DWT is used to decompose electrical signals into approximation components and details components, which are fed into the network for training. We choose Daubechies wavelet as the Mother wavelet. In detail, the db4 wavelet is used as the basic wavelet, the smooth error introduced by db4 is not easy to be detected, which makes the signal reconstruction process smooth. In addition, using wavelet transform denoising of data needs to select a suitable threshold $\lambda$ denoising is used to control precision. Rigrsure estimation is selected as the threshold method. The steps are as follows Take the absolute value of each element in signal $s(i)$ and order it from small to large. Then square each element to get a new signal sequence:

$$S(k) = (sort(|s|))^2, k = 0, 1 \cdots, N - 1 \qquad (7)$$

Then the risk vector is

$$R(k) = [N - 2k + \sum_{i=1}^{k} S(j) + (N - k)S(k)]/N] \qquad (8)$$

Find k that minimizes the value of R, and the threshold $Th$ is equal to

$$Th = \sqrt{S(k)} \qquad (9)$$

After the threshold of Gaussian white noise in the wavelet coefficient is determined, it is necessary to have a threshold function to filter the wavelet coefficient containing noise

coefficient and remove the Gaussian noise coefficient. Since the power quality disturbance signals simulated have a large amount of transient information, we need to ensure signal continuity and stability after DWT. We finally adopt the soft thresholding function to denoising, which has better overall continuity than hard thresholding.

### B. CONVOLUTIONAL NEURAL NETWORK
CNN is widely used in image recognition, semantic segmentation, and other fields [26]. It has achieved remarkable results [27]. 1-D CNN is used to analyze PQDs in this paper.

Compared with traditional methods, 1-D CNN PQDs identification has the advantage that it does not require complex preprocessing. CNN completes feature extraction and classification through a series of operations in a black box such as convolution and pooling. The core of the 1-D CNN classification model is the structural design of the network. It has better performance than traditional algorithms, and the feature selection and classification steps have the advantages of a small number of parameters and short training time compared to the multidimensional characteristics of the 2-D CNN. However, there are still some problems with the application of the 1-D CNN network in PQDs identification, such as the uniqueness of features, so we need a more efficient framework.

### C. BAYESIAN OPTIMIZATION
Since no CNN model can optimally generalize various datasets, hyperparameter adjustment is required before applying CNN to new datasets. Bayesian optimization [28] provides an effective method for optimizing unknown blackbox functions. Different from random search and grid search, Bayesian optimization has fewer iterations and faster operation speed, which makes it possible to find the best possible parameter settings faster. The specific process is as follows:

#### 1) GAUSSIAN PROCESS REGRESSION
For a random process, if the distribution of random variables at each position is Gaussian, the random process is called a Gaussian process. Given $X$ and $X'$, save the $g(x)$ and $g(x')$ output by the function into the vector and assume that they are obtained by multivariate Gaussian function, as shown in the equation below:

$$\begin{bmatrix} g(x) \\ g(x') \end{bmatrix} \sim N \left( \begin{bmatrix} \varphi(x) \\ \varphi(x') \end{bmatrix}, \begin{bmatrix} \Sigma(x, x) & \Sigma(x, x') \\ \Sigma(x', x) & \Sigma(x', x') \end{bmatrix} \right) \qquad (10)$$

We introduce a constant $\varphi$ with $||x - x'||$ for decreasing kernel function, the purpose of introducing kernel function is to make the fitted function more smoothly. In addition, we find that when the conditional probability $g(x)$ remains unchanged, $g(x')$ is a normal distribution. Gaussian process adopts the same computational method and extends it to

multiple dimensions:

$$\begin{bmatrix} g(x_1) \\ \vdots \\ g(x_k) \end{bmatrix} \sim N \left( \begin{bmatrix} \varphi(x_1) \\ \vdots \\ \varphi(x_k) \end{bmatrix}, \begin{bmatrix} \Sigma(x_1, x_1) & \cdots & \Sigma(x_1, x_k) \\ \vdots & \ddots & \vdots \\ \Sigma(x_k, x_1) & \cdots & \Sigma(x_k, x_k) \end{bmatrix} \right) \tag{11}$$

Then, Bayesian linear regression is performed on the existing first $k-1$ observations to calculate the posterior value of the new $k^{th}$ point and obtain the remaining observations. The implementation is as follows:

$$\varphi_k = \Sigma(x_k, x_{1:k-1})\Sigma(, x_{1:k-1}, x_{1:k-1})^{-1}g(x_{1:k-1}) \tag{12}$$

$$\sigma_k^2 = \Sigma(x_k, x_k) \\ - \Sigma(x_k, x_{1:k-1})\Sigma(x_{1:k-1}, x_{1:k-1})^{-1}\Sigma(x_{1:k-1}, x_k) \tag{13}$$

Finally, we compute the distribution of any desired point $X_k$ with a mean of $\varphi_k$ and a variance of $\sigma_k^2$.

### 2) ACQUISITION FUNCTION

The acquisition function is used to achieve the optimal solution of the objective function. According to the partial construction of posterior probability, the evaluation point with the greatest potential to be the optimal solution is selected by maximizing the acquisition function in the next iteration. This paper uses mathematical expectation to construct the acquisition function, as shown in Equation (11)

$$E_i = [f^* - \xi_n(x)]^+ + \eta_n(x)\varphi\left(\frac{f^* - \xi_n(x)}{\eta_n(x)}\right), \\ - [f^* - \xi_n(x)]\phi\left(-\frac{[f^* - \xi_n(x)]}{\eta_n(x)}\right) \tag{14}$$

where $n$ means that the evaluation has been completed $n$ times, and $f^*$ is the best value after the first $n$ evaluation, $\xi_n$ is the mean, $\eta_n$ is the standard deviation.

## III. THE PROPOSED PQDs CLASSIFICATION METHOD
### A. PRINCIPLE OF ECNN

To adaptively extract features from electrical signals in different regions and preserve the multiplicity of features, ECNN is designed for the automatic identification of PQDs. The model uses the high and low-frequency information of the raw signal and its frequency feature as inputs, corresponding to three submodels. The main contribution of the proposed method is to integrate features from different angles.

The PQDs signal is expressed as $X(n)$, the parallel FFT signal is shown as

$$X(n) = \sum_{n=0}^{N-1} x(n)e^{-i2\pi nj/N}, j = 0, 1 \cdots, N-1 \tag{15}$$

The high and low-frequency signals separated by DWT can be shown as

$$Y(n) = \sum_{n=0}^{N-1} x(n)F_0(m - 2n) \tag{16}$$

$$Z(n) = \sum_{n=0}^{N-1} x(n)F_1(m - 2n) \tag{17}$$

where $N$ represents the signal length and $F_0, F_1$ is a low and high-frequency filter to process the original signals. The inputs to the three submodels are $X(n)$, $Y(n)$, and $Z(n)$. We set $G_m = \{X(n), Y(n), Z(n)\}$, then the output of the convolution layer is calculated as

$$O_c = f(W^t * G_m + c_m^t) \tag{18}$$

where $W^t = \{W_x^t, W_y^t, W_z^t\}$ indicates the weight of the kernel in the convolution layer for $G_m$, the bias term is expressed as $c^t = \{c_x^t, c_y^t, c_z^t\}$, the sign * presents the convolution operation.

Each submodel combines the convolutional layer, BN layer, and pooling layer as feature extractors. The three submodels convolve data from different angles respectively through the convolution filter and deepen the network by layers. This structure allows the network to pay attention to the details of the PQDs signal, so that features can be extracted more fully. Max pooling is used to extract significant features and reduce network parameters simultaneously. The BN layer is embedded between the convolutional layer and the pooling layer to normalize data, prevent overfitting, and boost the network convergence speed.

To prevent feature loss, this paper combines the output of different convolutional layers. The input of the fusion layer is three feature vectors. The fusion layer splices the three feature vectors into a feature vector $R^t$ to achieve feature fusion as follows

$$\begin{aligned} R^t &= max\{0, f(W_x^t * X(n) + c_x^t) \\ &\quad + f(W_y^t * Y(n) + c_y^t) + f(W_z^t * Z(n) + c_z^t)\} \\ &\leq max\{0, f(W_x^t * X(n) + c_x^t)\} \\ &\quad + max\{0, f(W_y^t * Y(n) + c_y^t)\} + f(W_z^t * Z(n) + c_z^t) \\ &= O_p(X(n)) + O_p(Y(n)) + O_p(Z(n)) \end{aligned} \tag{19}$$

The fusion layer produces a more comprehensive output than the convolutional layer after the feature merge. $R^t$ involves the time and frequency domain characteristics of signals, and the former includes high-frequency and low-frequency characteristics. Compared to the single model, the combination of three features can better distinguish different PQDs. Therefore, this fusion is helpful for ECNN to extract appropriate features and ultimately enhance the ability of ECNN to identify PQDs.

### B. COMPOSITE CONVOLUTIONAL KERNEL

Adopting an efficient convolution can help reduce the computational cost of ECNN while preserving various features.
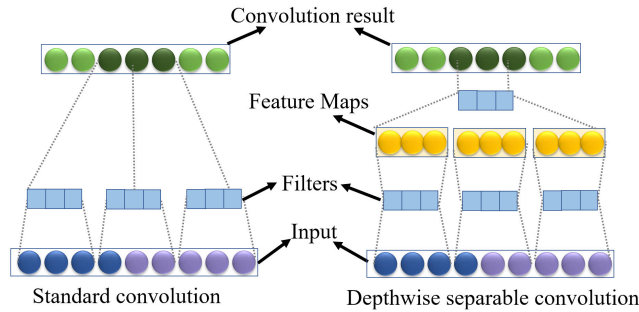
**FIGURE 1.** Principle of DSC.

Therefore, a composite convolution is proposed in this paper where both DSC and SAC are used in the convolution process. Different from the SAC, DSC decomposes the spatial correlation of the convolutional layer and channel correlation into depth convolution and pointwise convolution [29]. Compared to SAC, it can greatly reduce the number of parameters and calculations under the condition of ensuring little loss of accuracy.

For processing the same layer of data, the computational pairs of SAC and DSC are defined as

$$\frac{O_t(DSC)}{O_t(SAC)} = \frac{F_D^2}{F^2 C_o} + \frac{1}{K^2} \qquad (20)$$

and the comparison of parameter numbers can be presented as

$$\frac{O_s(DSC)}{O_s(SAC)} = \frac{K^2+1}{K^2C+1}\frac{C}{C_o} + \frac{C+1}{K^2C+1} \qquad (21)$$

where $O_t(SAC)$ is the calculation of SAC; $F^2$ is the pixel size of the output feature map; $K^2$ denotes convolutional kernel size; $C$ is the number of input channels; $C_o$ is the number of output channels; $O_t(DSC)$ is the computation of DSC; $F_D^2$ is the pixel size of the depth convolution output feature map; $O_s(SAC)$ and $O_s(DSC)$ is the parameter numbers of SAC and DSC respectively.

Fig.1 shows the principle of DSC, let $*d$ presents the DSC operation, the output of DSC is denoted as

$$O_d = f(W_d^t *_d *G_m + c_d^t) \qquad (22)$$

where $W_d^t$ and $c_d^t$ are DSC weight and bias. Moreover, considering that simply using DSC instead of SAC leads to performance degradation, since we define $\lambda \in [0, 1]$ to control the ratio of DSC, the output of the composite convolution is

$$O_c = [\lambda * f(W^t * G_m + c_m^t), (1 - \lambda) * f(W_d^t * G_m + c_d^t)] \qquad (23)$$

The function of composite convolution is to make $(1 - \lambda) * d$ features generated by DSC and SAC produces the other features in the convolution process of depth $d$. In this way, we can reduce the number of model parameters while ensuring that the recognition effect is not affected.

## C. FRAMEWORK OF ECNN
Referring to IEEE standard 1159 power quality monitoring standard [30], there are a total of 22 types of PQDs used to validate ECNN. Table 1 lists the specific disturbance types, including 5 single types such as sag, flicker, notch, and 17 complex disturbances such as sag with harmonics, transients with harmonics, etc. The sampling frequency of each signal is 12.8kHz, the sampling point is 256, and the analog period is 0.2s.

Fig.1 presents the framework for the proposed ECNN-$I_f J_s K_t$. ECNN-$I_f J_s K_t$ includes three submodels and Bayesian optimization module, where $I$, $J$, and $K$ represent the submodels with $I$, $J$, and $K$ convolution layers respectively. The three submodels are a simple CNN, which includes composite convolutional layers, pooling layers, and BN layers respectively. The final features are all fused and output to a full connection layer, then classified through the softmax layer. The first submodel is used to extract features from low-frequency signals and combine the convolutional results of the second submodel and the third submodel. The high-frequency features and the frequency domain features are extracted in the second and third submodels. All convolution operations use composite convolution. The outputs of the second and third submodels are fused directly with the first submodel, which combines different features of PQDs. ECNN-$3_f 3_s 2_t$ includes two-step fusion layers. We adjust the pooling layer to ensure that the feature map size of the fusion process is consistent. To make the ECNN more universally applicable, an adaptive context mechanism is designed. Bayesian optimization is embedded in three submodels to search for the best hyperparameters. PQDs identification based on ECNN is divided into the following steps.

1) Data preprocessing: The original signal $S(n)$ is decomposed into low-frequency parts $X(n)$ and high-frequency info $Y(n)$ by DWT. Then, the FFT of $X(n)$ is calculated and normalized.

2) Design and train ECNN model: Design an ECNN model and use the data obtained in Step1 is used as inputs for training and verification.

3) Adaptive-context mechanism processing: The hyperparameters are adjusted adaptively by Bayesian optimization, including learning rate, momentum term, and batch size.

4) PQDs identification: ECNN is trained to identify various PQDs.

Since the proposed model contains Bayesian optimization, it can adaptively update the model parameters for PQDs data in different regions. In addition, the use of DWT and the fusion of time domain information make ECNN have good noise resistance. Compared with other models, ECNN can effectively solve the problem that complex disturbance data with diverse features in different regions need to be adjusted repeatedly to perform identification.

## IV. FEASIBILITY ANALYSIS OF FEATURE EXTRACTION
We construct the ECNN-$3_f 3_s 2_t$ model for feature visualization analysis to investigate the effect of ECNN on feature
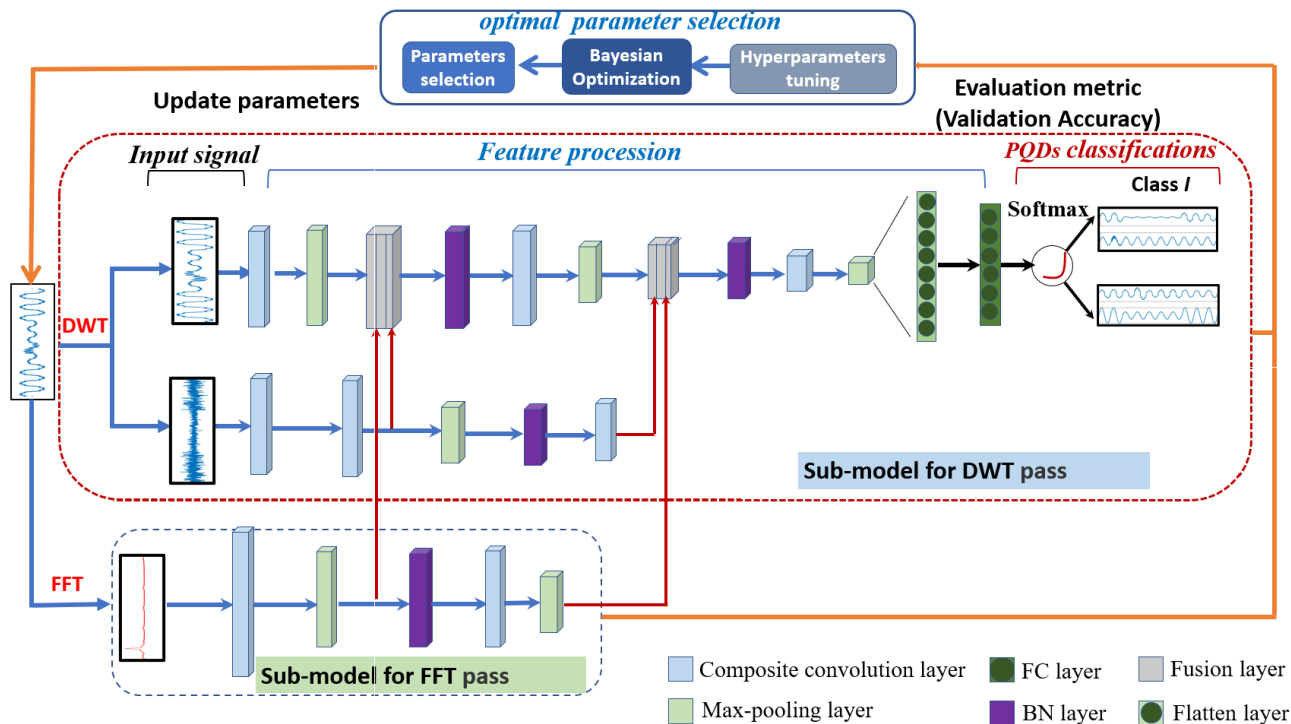
**FIGURE 2.** Framework of ECNN-$I_fJ_sK_t$, I = 3, J = 3, K=2.



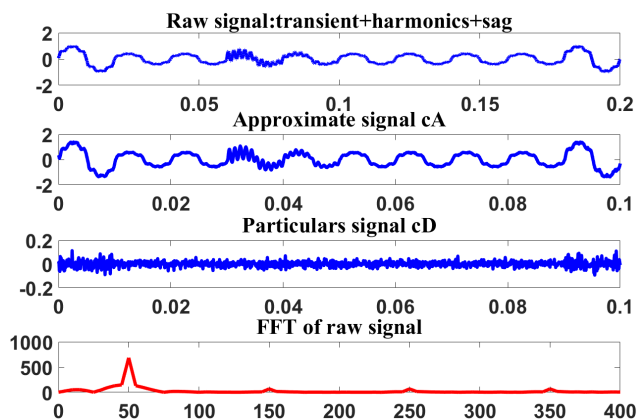**FIGURE 3.** Signal processed by DWT and FFT.



**FIGURE 4.** Features visualization.

extraction. The kernel-size(k) is 13, and depth(d) is 8 of all convolution layers. Meanwhile, we set the SAC and DSC ratio factor $\lambda$ to 0.5 and 100 nodes in the FC layer.

Fig.3 shows the raw signal and input of the three submodels. After DWT processing, the raw signal is decomposed into an approximate signal and a particulars signal whose length is halved, which are used as ECNN inputs. It can be clearly observed that the first two submodels mainly contain the temporal and transient features of the signal. While the input of the third submodel shows the feature of the original signal in the frequency domain. The high-frequency signal is confused and irregular but contains transient information. The low-frequency info retains the basic characteristics of
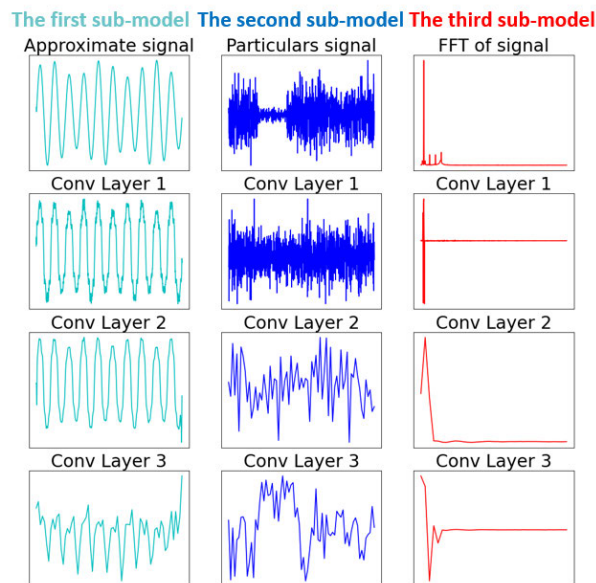
the raw signal. And we can find that the frequency of PQDs is concentrated at 50Hz through the frequency domain. Due to the particularity of the ensemble structure, ECNN can increase the distinction between temporal and transient parts of PQDs.

For ECNN, Fig.4 shows the visualization results of the abstract features from different layers. It proves that the
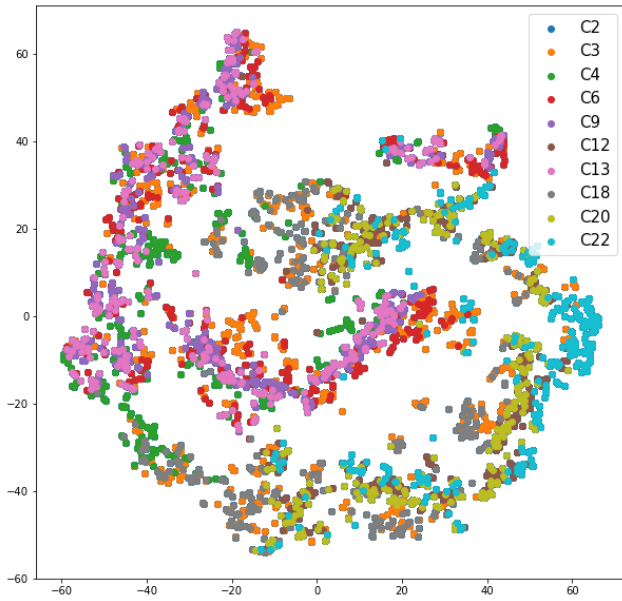
**FIGURE 5.** Visualizations of the orignal input data.



**FIGURE 6.** Visualizations of FC layer after training.

features are similar to the original input in the first convolutional layer. As convolution deepens, the feature map appears more abstract, but the main extracted features are still obvious. For example, in the first submodel, the characteristics of the fundamental signal are preserved by three convolutions. In the second submodel, the noise is gradually reduced and the transient features are more and more obvious with the deepening of the layers. The position of the fundamental frequency points is captured in the third submodel. Finally, after two fusions, the model has a strong automatic feature extraction capability for different types of raw signals.

Therefore, to further verify the classification effect of ECNN on PQDs with multiple temporal states or transient overlaps, a visual comparison analysis is conducted on the features of the raw signals and the output of FC layers. We use 10 types of PQDs as inputs, including single, double, and triple disturbances. Each disturbance type contains 2000 samples, and we use t-Distributed Stochastic Neighbor Embedding (T-SNE) to reduce dimension and visualize [31]. The visualization results of input data $S(n)$ and FC layer features are demonstrated in Fig.5 and Fig.6. It is clear that the raw data are very confusing and mixed together. For example, C12 and C20 both contain transient features, but it is difficult to distinguish the categories and they are superimposed directly on each other. Conversely, there is almost no overlap for all types of data in Fig.5, indicating that ECNN is able to effectively improve identification performance for different types of power quality disturbances.

## V. EXPERIMENTAL RESULT
### A. DATASET DESCRIPTION AND EXPERIMENTAL SETUP
In practice, we simulated 22 different types of PQDs based on IEEE Standard 1159 to verify ECNN performance, as shown in Table 1. We use the cross-validation method to prove the
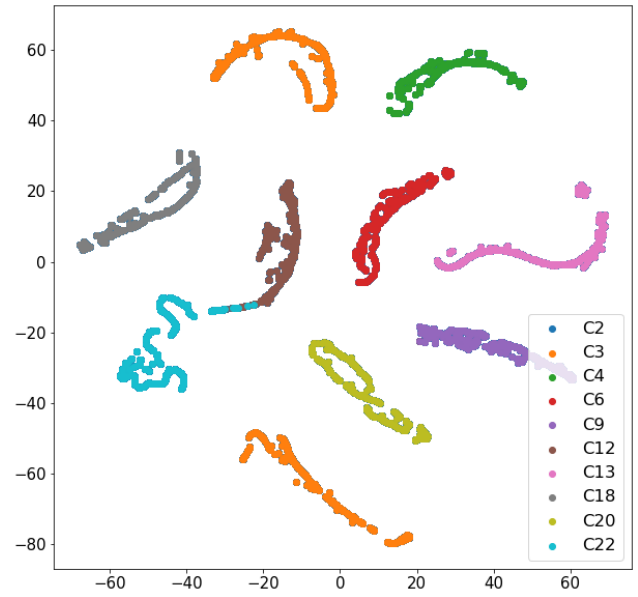
robustness of the method. There are 3000 samples for each type of PQDs, of which the training set contains 2000 samples, and the test set and validation set contain 500 samples, respectively. As we all know, data collected in the real environment always contains noise, so Gaussian white noise is superimposed on the raw signal randomly. Gaussian noise is characterized by a defined signal-to-noise ratio (SNR) of 20 dB, 30 dB, and 40 dB, respectively. This experiment is completed on the RTX3060 GPU hardware platform, and the environment is based on the Keras [32]. We set the epoch to 150, and the batch size of all 5 types of networks is set to 64, so that the training cost of different methods can be fairly compared. Partial hyperparameters of the model are determined by Bayesian optimization.

**TABLE 1.** 22 Types of PQDs.

| Class | PQDs | Class | PQDs |
|-------|------|-------|------|
| C1 | Normal | C12 | Transient+notch |
| C2 | Sag | C13 | Harmonic+transient |
| C3 | Notch | C14 | Harmonic+notch |
| C4 | Flicker | C15 | Harmonic+flicker |
| C5 | Spike | C16 | Flicker+notch |
| C6 | Sag+swell | C17 | Flicker+transient |
| C7 | Sag+transient | C18 | Flicker+harmonics+interrupt |
| C8 | Sag+notch | C19 | Flicker+harmonics+swell |
| C9 | Sag+hamonic | C20 | Transient+harmonics+flicker |
| C10 | Sag+flicker | C21 | Transient+harmonics+swell |
| C11 | Sag+interrupt | C22 | Transient+harmonics+sag |

In addition, due to the strong randomness and complexity of PQDs in the distribution network, the data simulated by MATLAB is always quite different from measured signals. Therefore, we have integrated a PQDs generation system to enhance the authenticity and reliability of the simulated data. The system has the functions of signal generation, waveform inversion, data acquisition, and visualization, corresponding
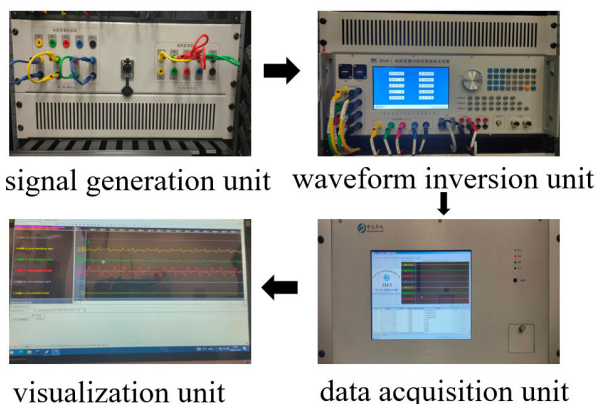
signal generation unit    waveform inversion unit

visualization unit    data acquisition unit

**FIGURE 7.** Experimental hardware framework.

to four units. Fig. 7 shows the system framework. Power quality standard equipment DANDICK DKLN-1 includes a signal generation and waveform inversion unit, which is mainly used to simulate and output PQDs signals. The data acquisition unit is Zhongyuan Huadian's ZH5 fault recorder, which can collect the time series of PQDs. TP-Link TL-SF1005 industrial network switch is connected to the computer host to form a visual processing unit, which is conducive to intuitive interactive interface operation. The sampling frequency of the hardware device can be applied to 256, 512, and 1024. Since the 256 sampling frequency is adopted for the analog signals, the sampling frequency used by this device is also set to 256. For each type of simulating PQDs, 500 samples are fed into the device for simulation, which is used to further compare the performance of ECNN with other methods.

## B. OPTIMIZATION OF MODEL PARAMETERS

In this paper, we use a parameter optimization method to optimize hyperparameters, which allows ECNN to adapt to diverse PQDs datasets from different regions. We set the optimal learning rate range $[10^{-3}, 10^{-2}, 0.1, 0.2, 0.3, 0.5, 0.8, 1]$. The optimization range of batch sample size is $[8, 16, 32, 64, 128, 256, 512]$. To explore which optimization method is more suitable for ECNN, Grid Search (GS), Particle Swarm Optimization (PSO), and Bayesian Optimization (BO) algorithms are used to optimize the ECNN model. The results are shown in Table 2.

**TABLE 2.** Hyperparameter optimization of search results.

| Hyper-parameter | Range | GS | PSO | BO |
|---|---|---|---|---|
| Learning rate | $(10^{-6}, 1)$ | $10^{-3.13}$ | $10^{-2.54}$ | $10^{-2.78}$ |
| Momentum Term | $(10^{-4}, 1)$ | 0.305 | 0.406 | 0.432 |
| Batch size | $[16,32,64,128]$ | 64 | 64 | 64 |

Table 3 shows the training results after the above three optimization methods. It is evident that the results of all three methods are quite accurate, which is related to the network structure of ECNN. The most obvious difference lies in the number of iterations in the optimization process. BO only

**TABLE 3.** Training results of three optimization methods.

| Method | Accuracy (%) | Iterations | Cost time(s) |
|---|---|---|---|
| GS | 99.31 | 256 | 300 |
| PSO | 99.28 | 307 | 350 |
| BO | 99.35 | 21 | 30 |

needs 30 iterations to find the optimal solution, while GS and PSO need more than 300 iterations, which takes about ten times as long as BO. Therefore, it is reasonable to choose BO to optimize the hyperparameters of the model.

Then, this paper uses the superimposed 30dB PQDs data to train the model with different convolutional layers, DSC ratio $\lambda$, convolution kernel size $k$, and the number of fusion layers to verify the influence of different parameters. Different ECNN models set the same number of FC layer nodes and convolution depth to ensure the fairness of the experiment.

We first adopt a dual-fusion ECNN to explore the influence of convolutional kernel size and DSC ratio on the model accuracy. As can be found in Fig.8, with the increase of kernel size, the accuracy shows an upward trend, at the same time, the calculation cost is also increasing. Therefore, the benefit of using an appropriate kernel size is higher than that of a small or large convolutional kernel. With the increase of $\lambda$, the performance of different ECNN models does not change significantly, while the accuracy is generally higher when $\lambda$ approaches 0.8. Meanwhile, the effect of 2-layer convolution is significantly lower than that of 3 and 4-convolution layers, which indicates that the more layers of convolution, the more obvious the effect of composite convolution is. In summary, we chose to set $\lambda$ and $k$ to 0.8 and 16.

Considering the particularity of the ensemble structure of ECNN, it is necessary to validate the effect of convolutional layers on the identification accuracy of PQDs. Table 4 shows the results. All models adopt two fusions, and it is evident that the precision is increasing with the increase of convolutional layers. At the same time, the number of parameters is also increasing, resulting in high computational complexity. To ensure as much accuracy as possible and reduce computational complexity, we finally chose ECNN-$3_f 3_s 2_t$ as the most appropriate training structure.

To verify the validity of the fusion numbers, we verify ECNN with 1, 2, and 3 fusion structures. Performance is shown in Table 5. From the perspective of accuracy, when the three submodels perform three convolutional layers and double fusions, the effect is the best. Therefore, we adopt two fusions on the three submodels to achieve the highest performance.

To prove the need for batch normalization of data in the training process, we draw the accuracy and loss curve for ECNN with BN and without BN in Fig.8. It shows that when the number of iterations reaches 20, the accuracy and loss of the ECNN with BN begin to stabilize, while the corresponding curves without a BN layer become stable when it gets 40 iterations. Therefore, the BN layer helps accelerate the convergence rate of the model and has a lower verification

**TABLE 4.** Accuracy of ECNN under different structures.

| ECNN | $2_f1_s1_t$ | $2_f2_s1_t$ | $2_f2_s2_t$ | $3_f2_s1_t$ | $3_f2_s2_t$ | $3_f3_s1_t$ | $3_f3_s2_t$ | $3_f3_s3_t$ | $4_f2_s1_t$ | $4_f3_s2_t$ | $4_f4_s3_t$ | $4_f4_s4_t$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Accuracy(%) | 96.21 | 96.93 | 97.26 | 99.13 | 99.59 | 99.4 | 99.59 | 98.35 | 99.35 | 96.76 | 99.00 | 99.37 |
| Parameter | 38497 | 38815 | 38993 | 38991 | 39177 | 39231 | 39541 | 39781 | 39239 | 39789 | 40193 | 40357 |

**TABLE 5.** Accuracy of ECNN under different fusion layers.

| Structure | $3_f1_s1_t$ | $3_f2_s2_t$ | | $3_f3_s3_t$ | | |
|---|---|---|---|---|---|---|
| Num of fusion | 1 | 1 | 2 | 1 | 2 | 3 |
| Accuracy(%) | 98.63 | 98.97 | 98.97 | 99.03 | 99 .00 | 96.02 |

**TABLE 6.** Structure and parameters of ECNN.

| Model | Conv1 | Conv2 | Conv3 | Kenel Size | FC layer |
|---|---|---|---|---|---|
| submodel1 | 8/1280 | 10/640 | 10/80 | 16 | 100 |
| submodel2 | 8/1280 | 10/640 | 10/80 | 16 | - |
| submodel3 | 8/2560 | 10/640 | - | 16 | - |
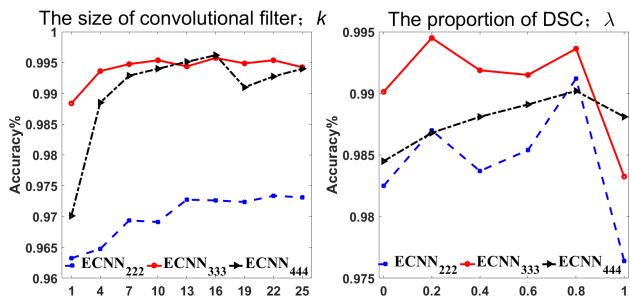


**FIGURE 8.** Influence of two main parameters $\lambda$, $k$, and the number of convolution layers on ECNN.



**FIGURE 9.** Training process of ECNN.

loss, which indicates that the BN layer is more conducive to feature extraction.

Based on the above experiments, we determine the optimal number of convolution layers, DSC ratio, fusion times, and convolution kernel size to construct the model. ECNN-$3_f3_s2_t$ is used as the best model structure considering accuracy and cost. The final detailed parameters are given in Table 6.

## C. COMPARATIVE ANALYSIS WITH OTHER CNNs

We select several advanced models for comparative analysis with the proposed method, which are CNN-LSTM, CNN-GRU, DCNN, and MFCNN. They are briefly described as follows.

1) CNN-LSTM: Two CNNs are used to learn spatial information, and then combined with a recursive LSTM layer to extract temporal features. Finally, the output of LSTM is transmitted to the fully connected layer. CNN depth is set to 64 and 128 and the model has 50 LSTM memory blocks, including multiple gates and activation functions.

2) CNN-GRU: The multi-fusion structure of the time and frequency domain feature is adopted. The raw data and frequency domain information by FFT are input respec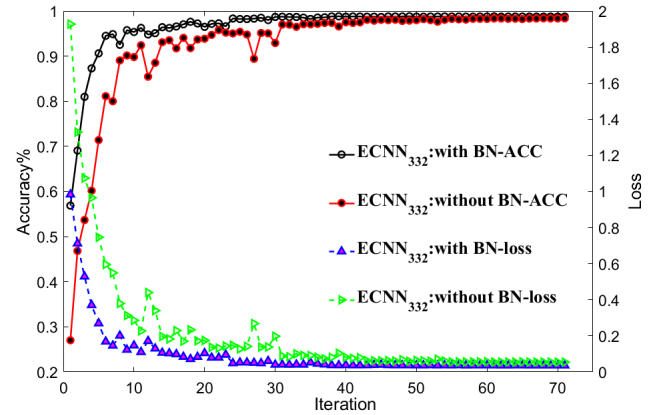tively. After convolution and pooling, the two tensors are connected together. In this model, the CNN depth is 16, and 16 GRU units are connected to the output layer.

3) DCNN: It is composed of three stacking units, each unit superimposes two convolutional layers, and a one-dimensional filter is used in each convolutional layer to extract features composed of one-dimensional convolutional layers, pooling layers, and BN layers. It has a deeper CNN structure with six SAC layers. Therefore, it contains a large number of parameters and takes a long time to train.

4) MFCNN: It contains two submodels, that extract features from time domain and frequency domain information, respectively. The model composes SAC and DEC for convolution, which enhances feature extraction capability. We used the optimal structure MFCNN$3_f3_j$ mentioned in the original literature to carry out the experiment.

For different CNN models, the batch size is set to 64 to ensure a fair comparison of model training costs. And to prevent overfitting, we set the dropout ratio to 0.5. Table 7 summarizes the identification accuracy and total parameters of all models under different noises.

We draw a conclusion that as noise increases, the performance of all methods decreases to varying degrees. Among them, CNN-GRU has the worst anti-noise performance, while ECNN and MFCNN have the best anti-noise performance. In addition, after repeated experiments, it is found that MFCNN and ECNN also have better stability than other methods. However, MFCNN has several times as many parameters as ECNN, which means the training takes longer. Although the number of parameters of CNN-GRU and CNN-LSTM is less than that of the proposed method, their identification results are not as high as ECNN under different noises. Therefore, considering the complexity and classification accuracy of the model, ECNN has the best performance in dealing with complex PQDs.

**TABLE 7.** Performance under different CNN methods.

| Model | clean | 20dB | 30dB | 40dB | Parameters |
|---|---|---|---|---|---|
| CNN-GRU | 98.73±0.26 | 96.21±0.91 | 98.31±0.31 | 98.71±0.32 | 4596 |
| CNN-LSTM | 98.78±1.18 | 97.33±0.83 | 98.41±1.24 | 98.20±1.64 | 32268 |
| DCNN | 99.12±0.63 | 97.22±1.28 | 98.65±0.95 | 98.92±1.57 | 165236 |
| MFCNN | 99.45±0.043 | 97.71±0.18 | 99.62±0.095 | 99.71±0.081 | 229805 |
| ECNN | 99.75±0.12 | 97.79±0.23 | 99.68±0.13 | 99.81±0.075 | 39541 |

### D. COMPARATIVE ANALYSIS WITH TRADITIONAL METHODS

We compared ECNN with the traditional methods, and the comparison results are listed in Table 8. The traditional methods require hand-engineered feature selection, which affects the classification results. For example, the algorithm based on FFT and ANN designed 16 features and achieved 96% accuracy [33]. In contrast, the Hilbert Huang transform (HHT) and the weighted bidi-recursive extreme learning machine (WBELM) achieved similar accuracy, requiring only four manual features [34]. However, the ensemble structure proposed in this paper extracts effective features automatically, which not only simplifies the PQDs identification process and shortens the feature extraction time, but also has more advantages in accuracy than traditional machine learning. Furthermore, the adaptive-context mechanism by ECNN makes it possible to identify complex PQDs with various features.

**TABLE 8.** Comparison of ECNN with traditional methods.

| Method | Types of PQDs | Num of Features | Noise(dB) | Accuracy(%) |
|---|---|---|---|---|
| FFT and ANN | 8 | 16 | 30 | 94.56 |
| HHT and BELM | 16 | 12 | 30 | 96.87 |
| DWT and PNN | 16 | 9 | 30 | 95.25 |
| DWT and SVM | 9 | 2 | 30 | 96.21 |
| VMD and DT | 15 | 4 | 30 | 96.52 |
| ECNN | 20 | Automatic | 30 | 99.41 |

## VI. CONCLUSION

The identification of PQDs is of great significance in researching the influence of power quality on EV charging. To reduce the potential risk of PQDs and provide a strategy for controlling PQDs overlap, an ECNN model is proposed in this paper. The model uses composite convolution to reduce training costs and automatically classifies PQDs by combining time domain and frequency domain information. In addition, the proposed ECNN model realizes an adaptive-context mechanism based on Bayesian optimization, so that the model can be adaptively used to identify PQDs in different regions, which extends the applicability of ECNN. Simulation results and experiments indicated that compared with other CNN methods, ECNN has a faster convergence speed, fewer model parameters, higher performance, and a more compact framework. In future work, we will further learn the

identification effect of ECNN for measured PQDs in different regions to further improve the scalability of the method.

### REFERENCES

[1] X. Zhao, S. Wang, and X. Wang, "Characteristics and trends of research on new energy vehicle reliability based on the web of science," *Sustainability*, vol. 10, no. 10, p. 3560, Oct. 2018.

[2] M. van der Kam and W. van Sark, "Smart charging of electric vehicles with photovoltaic power and vehicle-to-grid technology in a microgrid; a case study," *Appl. Energy*, vol. 152, pp. 20–30, Aug. 2015, doi: 10.1016/j.apenergy.2015.04.092.

[3] V. S.-P. Cheung, R. S.-C. Yeung, H. S.-H. Chung, A. W.-L. Lo, and W. Wu, "A transformer-less unified power quality conditioner having fast dynamic control," in *Proc. IEEE Energy Convers. Congr. Exposit. (ECCE)*, Cincinnati, OH, USA, vol. 5, Oct. 2017, pp. 2962–2968.

[4] Y. Deng, L. Wang, H. Jia, X. Tong, and F. Li, "A sequence-to-sequence deep learning architecture based on bidirectional GRU for type recognition and time location of combined power quality disturbance," *IEEE Trans. Ind. Informat.*, vol. 15, no. 8, pp. 4481–4493, Aug. 2019.

[5] S. Hui and X. Yonghai, "Study on the impact of voltage sags on different types of electric vehicle chargers," in *Proc. China Int. Conf. Electr. Distribution (CICED)*, Aug. 2016, pp. 1–4.

[6] R. Huang and X. N. Liu, "Simulation results and harmonic analysis of electric vehicle charging mode," *Instrum. Technol.*, 2018, doi: 10.19432/j.cnki.issn1006-2394.2018.02.006.

[7] Y. Luo, K. Li, Y. Li, D. Cai, C. Zhao, and Q. Meng, "Three-layer Bayesian network for classification of complex power quality disturbances," *IEEE Trans. Ind. Inform.*, vol. 14, no. 9, pp. 3997–4006, Sep. 2018.

[8] H. Jianming, J. Hezuo, and L. Xiaoming, "Classification for hybrid power quality disturbance based on STFT and its spectral kurtosis power system technology," *Power Syst. Technol.*, vol. 40, no. 10, pp. 3184–3191, 2016.

[9] Q. Tang, W. Qiu, and Y. Zhou, "Classification of complex power quality disturbances using optimized S-transform and kernel SVM," *IEEE Trans. Ind. Electron.*, vol. 67, no. 11, pp. 9715–9723, Nov. 2020, doi: 10.1109/TIE.2019.2952823.

[10] K. Thirumala, M. S. Prasad, T. Jain, and A. C. Umarikar, "Tunable-Q wavelet transform and dual multiclass SVM for online automatic detection of power quality disturbances," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3018–3028, Jul. 2018.

[11] S. Shukla, S. Mishra, and B. Singh, "Empirical mode decomposition with Hilbert transform for power quality assessment," in *Proc. IEEE PES Gen. Meeting Conf. Exposit.*, Jul. 2014, p. 1.

[12] P. S. Wright, "Short-time Fourier transforms and Wigner-Ville distributions applied to the calibration of power frequency harmonic analyzers," *IEEE Trans. Instrum. Meas.*, vol. 48, no. 2, pp. 475–478, Apr. 1999.

[13] S. Naderian and A. Salemnia, "An implementation of S-transform and type-2 fuzzy kernel based support vector machine algorithm for power quality events classification," *J. Intell. Fuzzy Syst.*, vol. 36, no. 6, pp. 5115–5124, Jun. 2019.

[14] R. Ahila, V. Sadasivam, and K. Manimala, "An integrated PSO for parameter determination and feature selection of ELM and its application in classification of power system disturbances," *Appl. Soft Comput.*, vol. 32, pp. 23–37, Jul. 2015.

[15] B. Biswal, P. K. Dash, and B. K. Panigrahi, "Non-stationary power signal processing for pattern recognition using HS-transform," *Appl. Soft Comput.*, vol. 9, no. 1, pp. 107–117, Jan. 2009.

[16] N. Huang, H. Peng, G. Cai, and D. Xu, "Feature selection and optimal decision tree construction of complex power quality disturbances," *Proc. CSEE*, vol. 37, no. 3, pp. 776–785, 2017.

[17] Z. Liu, Y. Cui, and W. Li, "A classification method for complex power quality disturbances using EEMD and rank wavelet SVM," *IEEE Trans. Smart Grid*, vol. 6, no. 4, pp. 1678–1685, Jul. 2015.

[18] W. Huihui, P. Wang, T. Liu, and W. Zhang, "Power quality disturbance classification based on growing and pruning optimal RBF neural network," *Power Syst. Technol.*, vol. 42, pp. 2408–2415, Aug. 2018.

[19] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–44, May 2015.

[20] R. Kumar, B. Singh, D. T. Shahani, A. Chandra, and K. Al-Haddad, "Recognition of power-quality disturbances using S-transform-based ANN classifier and rule-based decision tree," *IEEE Trans. Ind. Appl.*, vol. 51, no. 2, pp. 1249–1258, Mar./Apr. 2015.

[21] E. Balouji and O. Salor, "Classification of power quality events using deep learning on event images," in *Proc. 3rd Int. Conf. Pattern Recognit. Image Anal. (IPRIA)*, 2017, pp. 21–216.

[22] N. Mohan, K. P. Soman, and R. Vinayakumar, "Deep power: Deep learning architectures for power quality disturbances classification," in *Proc. Int. Conf. Technol. Advancements Power Energy (TAP Energy)*, Dec. 2017, pp. 1–6.

[23] W. Qiu, Q. Tang, J. Liu, and W. Yao, "An automatic identification framework for complex power quality disturbances based on multifusion convolutional neural network," *IEEE Trans. Ind. Informat.*, vol. 16, no. 5, pp. 3233–3241, May 2020.

[24] S. Wang and H. Chen, "A novel deep learning method for the classification of power quality disturbances using deep convolutional neural network," *Appl. Energy*, vol. 235, pp. 1126–1140, Feb. 2019.

[25] X. Xiao and K. Li, "Multi-label classification for power quality disturbances by integrated deep learning," *IEEE Access*, vol. 9, pp. 152250–152260, 2021, doi: 10.1109/ACCESS.2021.3124511.

[26] P. Ke, M. Cai, H. Wang, and J. Chen, "A novel face recognition algorithm based on the combination of LBP and CNN," in *Proc. 14th IEEE Int. Conf. Signal Process. (ICSP)*, Aug. 2018, pp. 539–543.

[27] F. Yang, G. Wang, X. Peng, J. Wen, and Z. Li, "Partial discharge pattern recognition of high-voltage cables based on convolutional neural network," *Electr. Power Automat. Equip.*, 2018, doi: 10.16081/j.issn.1006-6047.2018.05.018.

[28] J. Motskus and L. Motskus, "Bayesian approach to global optimization and applications," *J. Quality Technol.*, 1987, doi: 10.12968/coan.2013.18.4.177.

[29] 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," 2017, *arXiv:1704.04861*.

[30] *IEEE Recommended Practice for Monitoring Electric Power Quality*, IEEE Power and Energy Society, IEEE Standard 1159-2009, 2009.

[31] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[32] F. Chollet. (2015). *Keras: Deep Learning Library for Theano and Tensorflow*. [Online]. Available: https://keras.io/

[33] F. A. S. Borges, R. A. S. Fernandes, I. N. Silva, and C. B. S. Silva, "Feature extraction and power quality disturbances classification using smart meters signals," *IEEE Trans. Ind. Informat.*, vol. 12, no. 2, pp. 824–833, Apr. 2016.

[34] M. Sahani and P. K. Dash, "Automatic power quality events recognition based on Hilbert Huang transform and weighted bidirectional extreme learning machine," *IEEE Trans. Ind. Informat.*, vol. 14, no. 9, pp. 3849–3858, Jan. 2018.
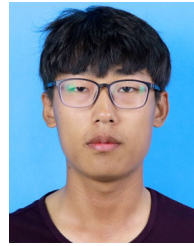
**MINGHAO WANG** was born in Jinan, China, in 1999. He received the B.Sc. degree in software engineering from Northeastern University, Shenyang, China, in 2021, where he is currently pursuing the M.Sc. degree. His research interests include power quality monitoring, power system measurement, and signal processing.
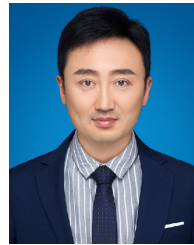


**ZHUOFU DENG** received the B.E. and M.E. degrees in software engineering from Northeastern University, China, in 2007 and 2009, respectively, where he is currently an Associate Professor with the Software College. His main research interests include computer vision, artificial intelligence, and medical image analysis.



**YUWEI ZHANG** was born in Heilongjiang, China, in 1991. He is currently pursuing the Ph.D. degree in software engineering under the joint training with Northeastern University, Shenyang, China, and BMW Brilliance Automotive. His research interests include power quality detection and influence analysis of electric vehicle charging.



**ZHILIANG ZHU** received the Ph.D. degree in computer science from Northeastern University, China. His main research interests include information integrate, complexity software systems, network coding and communication security, chaos-based digital communications, and the applications of complex-network theories. He is a Fellow of the China Institute of Communications and a member of the Software Engineering Education Instruction Committee of Education Ministry Expert of SOA Standards Working Group of Ministry of Industry and Information Technology.