

Research Article

Automatic Detection of Power Quality Disturbance Using Convolutional Neural Network Structure with Gated Recurrent Unit

Enes Yiğit ¹, Umut Özkaya ², Şaban Öztürk ³, Dilbag Singh ⁴, and Hassène Gritli ^{5,6}

¹Department of Electrical Electronics Engineering, Uludağ University, Bursa, Turkey

²Department of Electrical and Electronics Engineering, Konya Technical University, Konya, Turkey

³Department of Electrical and Electronics Engineering, Amasya University, Amasya, Turkey

⁴School of Engineering and Applied Sciences, Bennett University, Greater Noida, India

⁵RISC Lab (LR16ES07), National Engineering School of Tunis, University of Tunis El Manar, Tunis, Tunisia

⁶Higher Institute of Information and Communication Technologies, University of Carthage, Tunis, Tunisia

Correspondence should be addressed to Hassène Gritli; grhass@yahoo.fr

Received 25 April 2021; Revised 6 July 2021; Accepted 25 July 2021; Published 2 August 2021

Academic Editor: omar cheikhrouhou

Copyright © 2021 Enes Yiğit et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Power quality disturbance (PQD) is essential for devices consuming electricity and meeting today's energy trends. This study contains an effective artificial intelligence (AI) framework for analyzing single or composite defects in power quality. A convolutional neural network (CNN) architecture, which has an output powered by a gated recurrent unit (GRU), is designed for this purpose. The proposed framework first obtains a matrix using a short-time Fourier transform (STFT) of PQD signals. This matrix contains the representation of the signal in the time and frequency domains, suitable for CNN input. Features are automatically extracted from these matrices using the proposed CNN architecture without preprocessing. These features are classified using the GRU. The performance of the proposed framework is tested using a dataset containing a total of seven single and composite defects. The amount of noise in these examples varies between 20 and 50 dB. The performance of the proposed method is higher than current state-of-the-art methods. The proposed method obtained 98.44% ACC, 98.45% SEN, 99.74% SPE, 98.45% PRE, 98.45% F1-score, 98.19% MCC, and 93.64% kappa metric. A novel power quality disturbance (PQD) system has been proposed, and its application has been represented in our study. The proposed system could be used in the industry and factory.

1. Introduction

Due to the increase in technological developments and easier access to products, many technological devices have started to be used widely. These devices consume energy, and this energy is mostly consumed as electrical energy. All these distributed devices connected to the grid can cause PQD [1]. The leading causes of PQD include nonlinear loads, flexible AC transmission devices, variable frequency drives, arc devices, and converters used in power electronics [2]. Ideally, grid voltages and currents should be in a clean

sinusoidal form. If disturbing components are included in the network system, power losses and various disturbances occur. In this case, almost all electronic devices, from industrial equipment to electronic household appliances, are adversely affected. Besides, energy providers are badly affected in this case [3]. Understanding the causes of these situations enables action to be taken. For this, classifying the problems that arise is one of the most effective solutions [4].

Pattern recognition (PR) applications for PQD detections have critical importance. Due to machine learning (ML) methods and AI applications that have recently

become widespread, this process has become easier [5]. Some studies in the literature divide the PQD classification process into three parts: feature extraction, feature selection, and designing of classifiers [6]. In fact, these three stages are interconnected. In the feature extraction stage, specific situations based on the problem are obtained. Although more effective solutions are used for these stages today, in the past, the feature extraction step was highly dependent on the expert's experience and statistical capacity. Uyar et al. [7] proposed a wavelet entropy-based feature extraction approach for PQD classification. Jayasree et al. [8] presented a PQD classification framework consist of two steps. In the first step, an envelope detector using the Hilbert transform is used. In the second step, an artificial neural network architecture is used to classify the information from the first step. Reaz et al. [9] used discrete wavelet transforms as a feature extractor. Fengzhan and Rengang [10] used S-transform as a feature extractor, and Janik and Lobos [11] used radial basis function (RBF) networks. Lopez-Ramirez et al. [12] used empirical mode decomposition (EMD) for the classification of PQD data. Most of these methods are generally determined according to the properties of a specific dataset. When these methods can be used with another dataset, their performances decrease. In addition, the dimensions of some features are not suitable for classifiers, or classification takes a long time. For this reason, some researchers use feature selection methods to overcome these problems. After the feature selection procedure, a low-dimensional representation of the problem occurs. The representation power of these less parameterized features remains the same. Lee and Shen [13] proposed a feature selection algorithm named as probabilistic neural network-based feature selection (PFS) for PQD data. Panigrahi and Pandi [14] used the genetic algorithm for feature vector the selection to increase PQD classification performance. Singh and Singh [15] used the ant colony optimization technique to select optimal features. Huang et al. [16] presented a feature selection framework for PQD. Their framework includes entropy-importance (EnI)-based random forest (RF) model for selection process. Feature selection algorithms contribute positively to the success of the classification and reduce the time. But the computation of these methods is very complex, and it is model-sensitive.

The purpose of feature extraction and selection in the PQD classification process is to increase the classification performance. Binary and multiclass classifiers are used together with the mentioned hand-crafted features and feature selection methods. Hidden Markov model (HMM) [17], decision tree [18], rule-based systems [19], support vector machine (SVM) [11], probabilistic neural network [20], ANN [21], independent component analysis (ICA) [22], and K-nearest neighbor classifier (kNN) [23] are the most used algorithms for PQD classification. Studies with these classifiers continued intensely until the deep learning approach became popular. Deep learning methods produce striking results in almost all AI problems. Inspired by its high success in other studies, PQD classification studies have been started with deep learning approaches.

CNN, the most popular deep learning technique, is particularly effective in analyzing two-dimensional matrices and images. The robust feature representation power of the CNN architecture in 2D matrices comes from 2D kernels in the convolution layer, in addition to other factors such as other layers of the architecture and activation functions [24]. Among the current PQD classification studies in the literature, those including CNN are briefly reviewed below. Wang and Chen [25] proposed a closed-loop deep-learning method to classify PQD data. Liu et al. [26] used deep CNN and SVM together to classify PQD. Cai et al. [27] combined Wigner–Ville distribution (WVD) with CNN for the PQD dataset. Deng et al. [28] proposed a sequence-to-sequence deep learning model with bidirectional GRU for PQD classification. Shen et al. [29] proposed improved principal component analysis-guided 1D-CNN for PQDs. Rodriguez et al. [30] presented a convolutional auto-encoder compression framework and a stacked long short-term memory (LSTM). Subudhi and Dash [31] proposed the grey wolf optimization-(GWO-) based extreme learning machine (ELM) algorithm to classify PQD signals with limited data. Basha-wyah and Subasi [32] classified five PQDs signals with different machine learning algorithms. Biswal and Dash [33] used the fast dyadic ST algorithm with fuzzy decision tree for power quality disturbance. Khokhar et al. [34] proposed an optimal feature selection algorithm to classify PQDs. Li et al. [35] detected and classified PQDs by using Dag-SVMs with double resolution S-transform. CNN-based studies have superior performance in real samples as well as artificial samples. However, the CNN algorithms are not suitable for datasets that do not contain enough samples[36–38].

In this work, an efficient CNN architecture [39–41] is presented to classify single or composite PQD problems. For this purpose, the dataset is focused on preventing the CNN architecture from falling into the problem of overfitting. As learned from the literature on this subject, using a relatively large and sufficient dataset is essential for training the CNN algorithm. As the second stage, a high-performance CNN architecture designed for PQD sample classification is presented. The proposed architecture is a GRU-supported linear CNN architecture. The proposed CNN-GRU combination improves the performance compared to its counterparts in the literature. It includes a linear CNN architecture based on the proposed deep learning technique. Fully connected layers (FCLs) in this linear CNN architecture have been removed. Instead of these layers, GRU layers have been added. The contributions of the proposed method can be summarized as follows:

- (i) Thanks to the proposed GRU-supported CNN architecture; high classification performance is obtained for PQD datasets containing fewer samples
- (ii) It is faster because it contains fewer parameters than CNN architectures with FCL
- (iii) Suitable for end-to-end training

- (iv) It performs more effectively than current state-of-the-art methods

The rest of this paper is organized as follows: Section 2 includes background information of related algorithms and the proposed method. Section 3 provides experimental details and results. Finally, the conclusion is presented in Section 4.

2. Methodology

2.1. The Background in the Deep Learning Models. The concept of deep learning represents a relatively deep understanding of knowledge. In other words, it aims to be as knowledgeable as an expert in a problem. This is possible using deeper networks and the increase in training samples. Additionally, some mathematical interventions are done in architecture. The most widely used deep learning architecture today is CNN [42]. CNN is ideal for analyzing one-, two-, and three-dimensional matrices or vectors. This section focuses on 2D-CNN. The best way to understand CNN architecture is to understand the CNN layers. For this reason, the operation of CNN layers is covered in this section.

The layer most associated with the CNN architecture is the convolution layer. A CNN architecture contains many convolutional layers of various kernel sizes. These convolution layers learn about the features of the problem. Convolution layers are applied to the image by a convolution operation. One of the most significant advantages of sliding a convolution kernel over the image is parameter sharing. In this way, the total number of parameters in the network is reduced. Another layer in a basic CNN architecture is the pooling layer. This layer is usually used after the convolution layer or after the rectified linear unit (ReLU) layer. The most important task of the pooling layer is downsampling. This reduces the total number of parameters while preserving essential features in the matrix. There are types such as max-pooling, sum-pooling, and average-pooling; max-pooling is used in this study. The other essential layer of a basic CNN architecture is ReLU. Generally, one ReLU is used after almost every convolution layer. The task of the ReLU layer is to disrupt the linear structure of the network and to disaggregate the network. The traditional ReLU layer pulls negative parameters to zero. FCL layer is a kind of artificial neural network structure. It consists of many neurons, and all nodes are interconnected. This section is generally used for the classification of extracted features. A CNN architecture consisting of these three basic layers is calculated as in the following equation:

$$f(l_{\text{next}}) = \text{pool}_{n \times n}(\sigma(w \otimes [D_{\text{in}}] + b)), \quad (1)$$

where l_{next} represents the input of the next layer, pool is the pooling layer, n is the pooling window, σ means ReLU function, w represents convolution kernel, D_{in} represents the input, and b is the bias. The softmax function is generally used at the end of classifier networks. It calculates the probability distribution for m class output using the following equation:

$$P_j = \frac{e^{x_j}}{\sum_1^m e^{x_m}}. \quad (2)$$

In addition to these base layers, many new layers have been proposed and continued to be submitted. However, these layers are sufficient for a clear understanding of the CNN architecture proposed in this study.

2.2. Gated Recurrent Unit (GRU). GRU has emerged to effectively avoid the gradient burst or loss problem in recurrent neural networks (RNN). GRU can be thought of as a simpler version of long short-term memory (LSTM) [43–45]. Two gates are defined in a GRU unit: reset gate and update gate. The reset gate, which we will denote with r , puts the new input in cooperation with the previous memory. The update gate, which we will indicate with z , is responsible for protecting the previous memory value. To calculate the transition functions of a GRU, the following equations can be used:

$$z_t = \sigma(W^z x_t + V^z h_{t-1} + b^z), \quad (3)$$

$$r_t = \sigma(W^r x_t + V^r h_{t-1} + b^r), \quad (4)$$

$$\tilde{h}_t = \tanh(W^c x_t + V^c (r_t \odot h_{t-1})), \quad (5)$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t, \quad (6)$$

where \odot denotes the element-wise product, k represents the dimensionality parameter of hidden vectors, and W, V , and b are the shared parameters.

2.3. The Proposed Method. The proposed method is designed to investigate the effect of GRU layers on CNN architecture. A performance comparison was made between the proposed model with Vgg-16, GoogleNet, and ResNet-50 models. Pretrained CNN networks start learning with a transfer learning approach and are trained shallowly with generated power quality distribution data. These data in the one-dimensional signal format are converted into a two-dimensional matrix format by the STFT method. The obtained data through this transformation contain more features than raw signal data. The feature extraction and classification process have occurred automatically by providing this data input to our proposed GRU-based CNN network. In the training of the network, the number of mini-batches is ten, and the learning rate has been chosen as 0.0001. Dropout is set to 0.2. A high learning rate does not bring success in problem convergence. The learning rate with very low values requires a long training period [42, 46, 47]. A stochastic gradient descent algorithm is used to provide parameter optimization of the proposed CNN model [48].

In the proposed CNN architecture, the GRU structure has replaced fully connected layers. Two-level GRU blocks are included in the proposed CNN architecture. The first GRU block consists of 200 hidden neurons, and the latter GRU block consists of 100 hidden neurons. A dropout

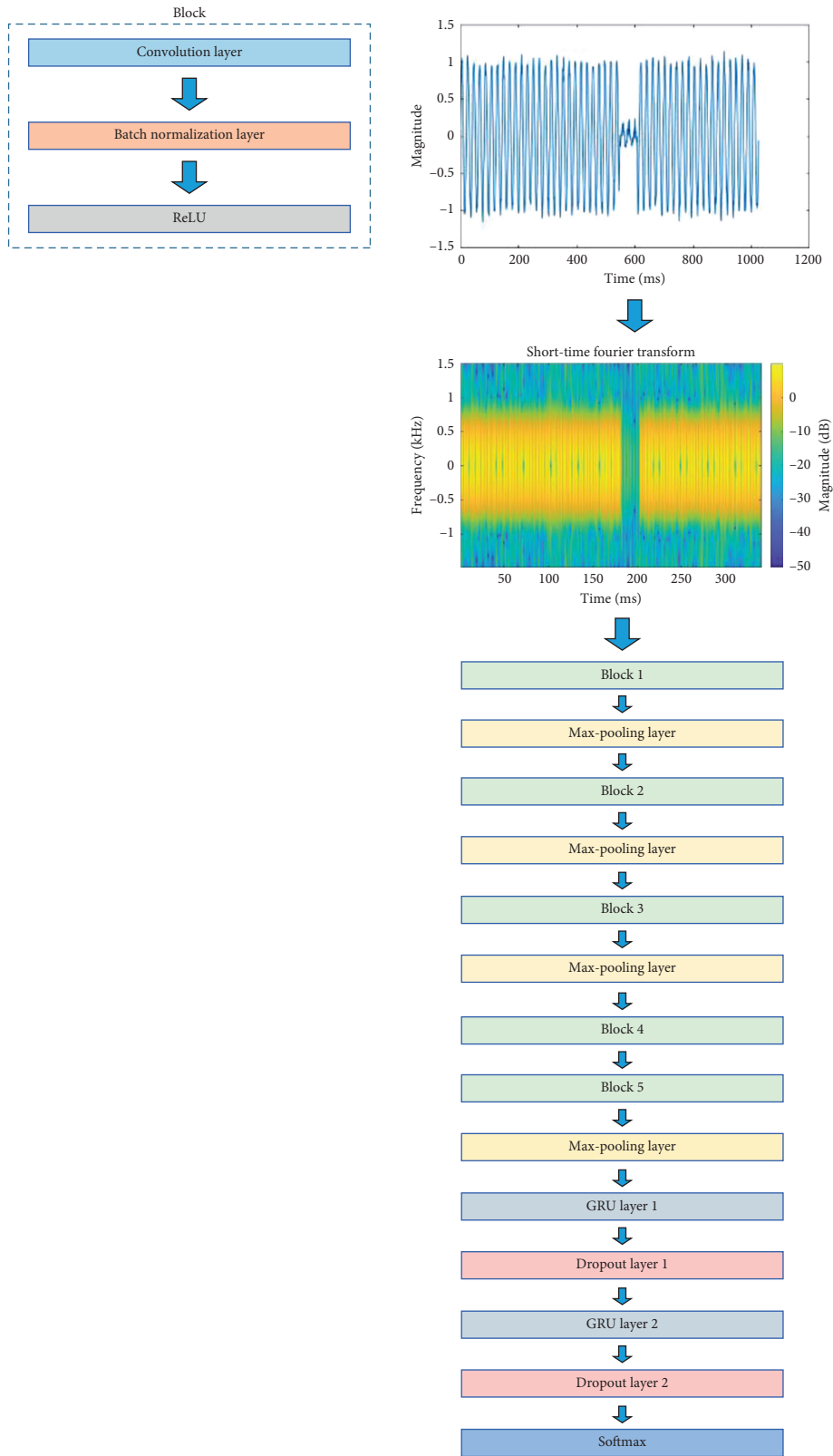


FIGURE 1: The overall structure of the proposed method.

layer has been added between GRU blocks to avoid overfitting problems. The proposed method is shown in Figure 1.

The basic architecture of the proposed method is the structure called a block. The block structure consists of one convolution, one batch normalization, and one rectifier

linear unit (ReLU) layer, respectively. There are a total of five block structures in the proposed architecture. There are maximum pooling layers between the first four block structures. In the proposed architecture, there are two GRU layers instead of fully connected layers. By means of dropout layers between the GRU layers, the overfitting problem is avoided.

3. Results and Discussion

3.1. Dataset Description. The dataset used within the study's scope was generated in the simulation environment, and noise levels of 20–50 dB were added. The dataset consists of 7 classes in total and includes 12336 signals. 75% of these data is reserved for training and 25% for testing. The dataset comprises singular and composite power quality defects. While singular defects are in sag, swell, oscillatory transient, flicker, and harmonics, composite defects consist of sag + harmonics and swell + harmonics classes. The presence of different and high levels of noise in the signals made it more realistic and challenging. In Table 1, the number of seven PQD signals containing pure sine waves is given to evaluate the performance of the GRU-based CNN network. Parameter changes were made in accordance with the IEEE-1159 standard [33]. Examples of the PQD signal are shown in Figure 2.

The amount of PQD signals classically is very important. This situation, which is called as data balance, is vital for training and testing CNN networks. The generated dataset contains 7 PQD signal classes in total. There are 1800 pieces of sag, 2000 pieces of swell, 1736 pieces of oscillatory transient, 1600 pieces of flicker, 2000 pieces of harmonic, 1600 pieces of sag+harmonic, and 1600 pieces of swell + harmonic PQD data. Table 1 includes the class distribution of these data.

3.2. Evaluation Metric. Seven classification metrics were used to evaluate the proposed CNN architecture and pretrained CNN networks. These metrics are accuracy (ACC), sensitivity (SEN), specificity (SPE), precision (PRE), $F1$ -score, Matthews correlation coefficient (MCC), and kappa, respectively. In addition, there are terms used in the calculation of metrics. These terms are true positive (TP), true negative (TN), false positive (FP), and false negative (FN). The computations of these metrics are given in the following equations:

$$\text{accuracy} = \frac{TP + TN}{TP + FN + TN + FP}, \quad (7)$$

$$\text{sensitivity} = \frac{TP}{TP + FN}, \quad (8)$$

$$\text{specificity} = \frac{TN}{TN + FP}, \quad (9)$$

TABLE 1: Number of PQD signal data in each class.

Class	Number of data
Sag	1800
Swell	2000
Oscillatory transient	1736
Flicker	1600
Harmonic	2000
Sag + harmonic	1600
Swell + harmonic	1600
Total	12336

$$\text{precision} = \frac{TP}{TP + FP}, \quad (10)$$

$$F1 - \text{score} = \frac{2 \times TP}{2 \times TP + FN + FP}, \quad (11)$$

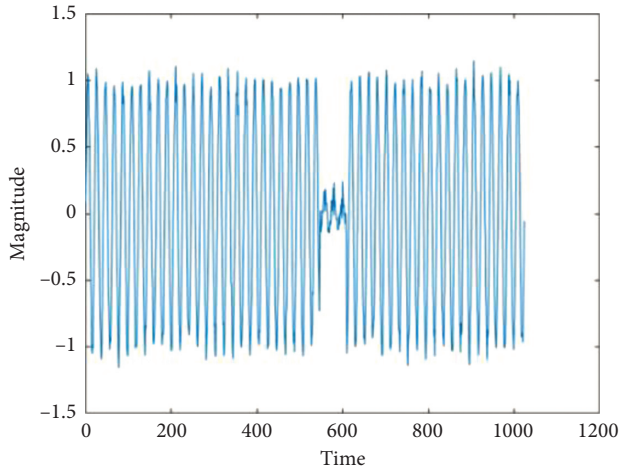
$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}, \quad (12)$$

$$\text{kappa} = \frac{\text{total accuracy} - \text{random accuracy}}{1 - \text{random accuracy}}, \quad (13)$$

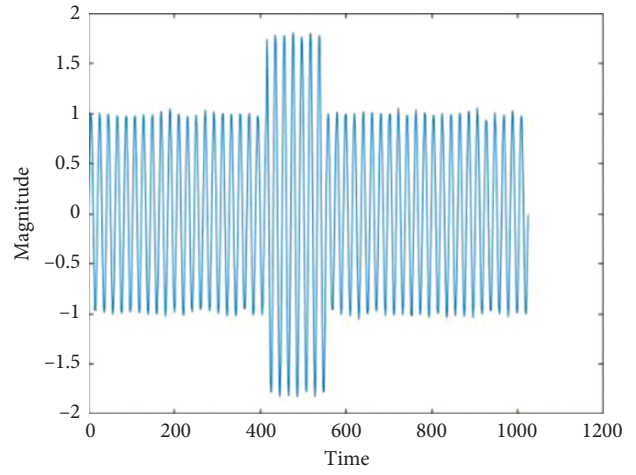
3.3. Results. The proposed method was performed with a computer with Intel Core i7-7700K CPU (4.2 GHz), 32 GB DDR4 RAM, and NVIDIA GeForce GTX 1080 graphic card. To compare the performance of the proposed method, a comparison was made with a pretrained CNN network in three different architectures. The importance of this process is to show the performance of the proposed CNN architecture with scratch structure when supported by GRU modules. Also, training charts of each CNN network are shown in Figures 3–6. Blue lines on training curves show training accuracy, and black lines show test accuracy. Red lines on the loss curve represent education; black lines represent test loss. As can be seen from the graphs, GRU-based CNN architecture showed higher performance. However, the overfitting problem cannot occur in the proposed CNN + GRU architecture. The proposed method quickly begins to converge on the problem during the training process. The availability of a sufficient amount of training data led to high performance in all CNN networks.

According to Figures 4–6, pretrained CNN networks using transfer learning and the proposed CNN architecture showed very high performance. The main reason for this situation is that there is enough data for training and testing. A comparative analysis is presented for specified CNN models in Table 2.

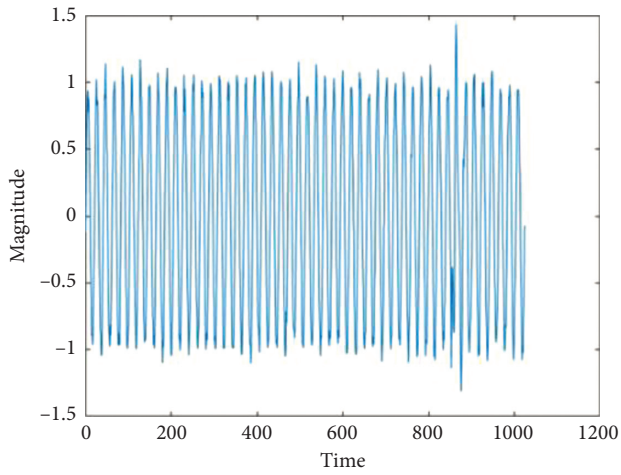
Table 3 contains the number of parameters of CNN architectures. The number of parameters of the proposed method is very low.



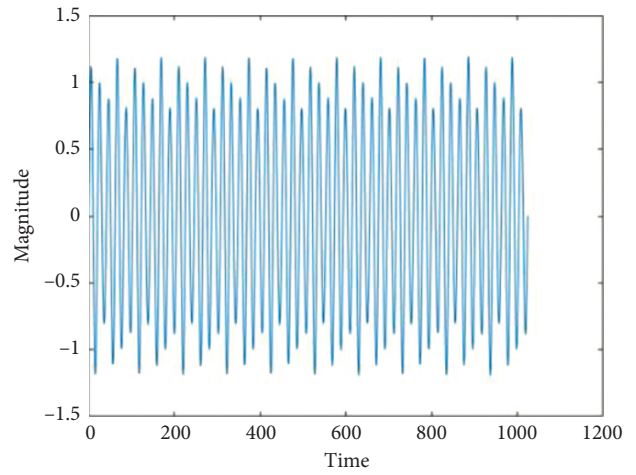
(a)



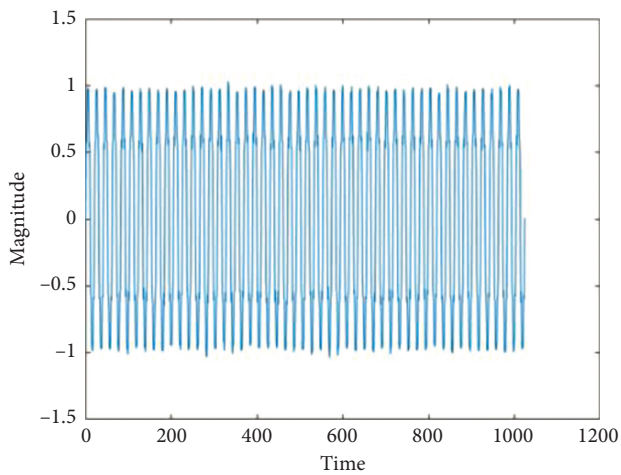
(b)



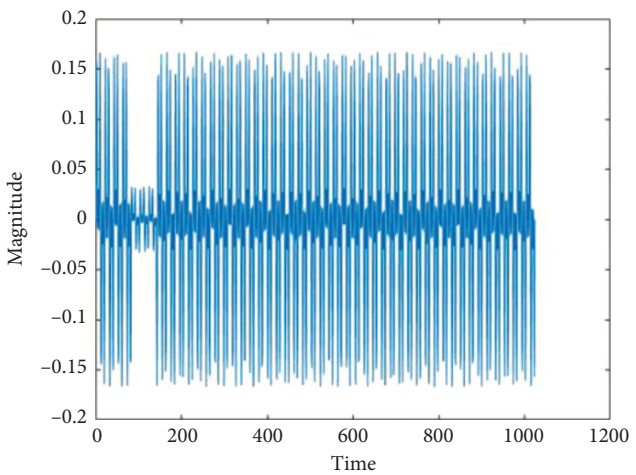
(c)



(d)



(e)



(f)

FIGURE 2: Continued.

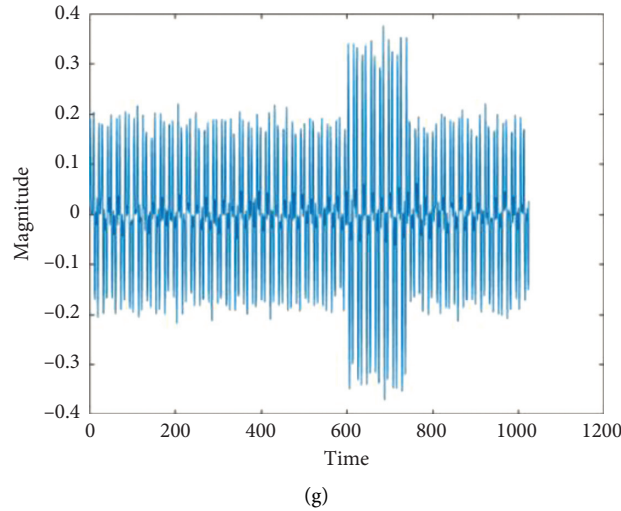


FIGURE 2: Examples of PQD signals. (a) Sag. (b) Swell. (c) Oscillatory transient. (d) Flicker. (e) Harmonics. (f) Sag + harmonics. (g) Swell + harmonics.

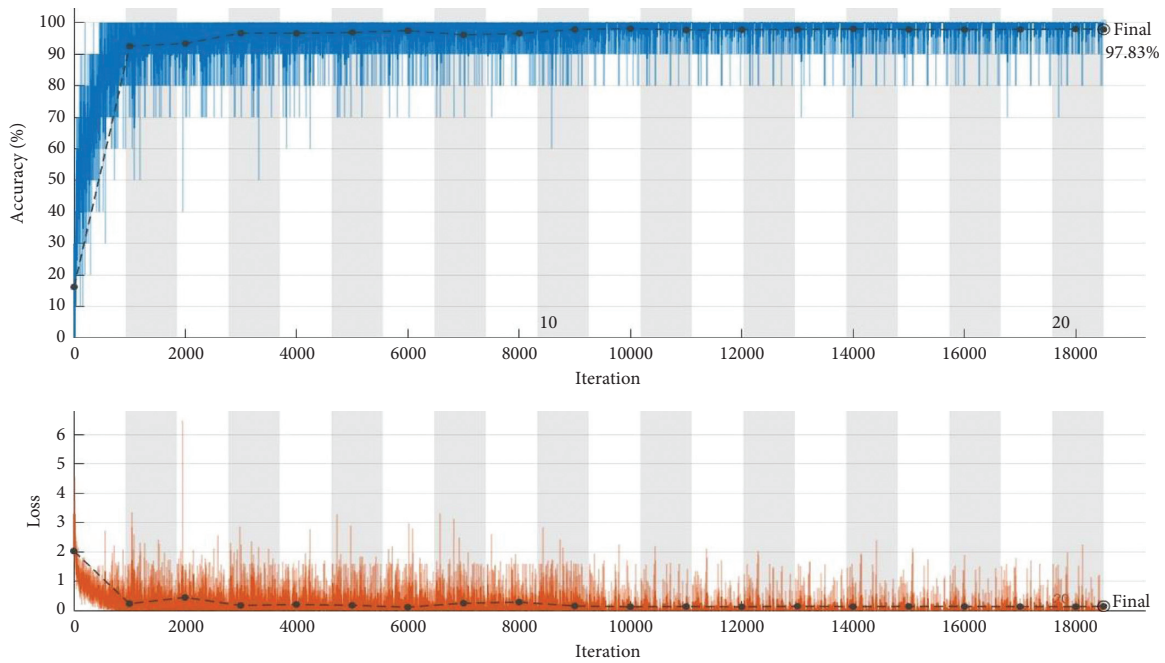


FIGURE 3: Training and testing process of the VGG-16 model.

3.4. Discussion. Considering the performance of the proposed CNN architecture and the number of parameters, it stands out more than other pretrained CNN networks. According to the results in Table 2, GoogleNet showed the lowest performance. Performance of this CNN architecture is 96.73% ACC, 96.68% SEN, 99.45% SPE, 96.76% PRE, 96.71% F1-score, 96.17% MCC, and 86.63% kappa. The performance of the VGG-16 and ResNet-50 models is very close to each other. However, the VGG-16 model exhibited higher performance. The metric performance of the VGG-16 model provided 97.83% ACC, 97.80% SEN, 99.64% SPE, 97.83% PRE, 97.81% F1-score, 97.45% MCC, and 91.13%

kappa. The proposed CNN model has the highest performance. It achieved better results than other pretrained CNN models in 7 classification metrics. The performance of this model is 98.44% ACC, 98.45% SEN, 99.74% SPE, 98.45% PRE, 98.45% F1-score, 98.19% MCC, and 93.64% kappa. On the contrary, when the number of parameters is evaluated, the proposed method shows a higher performance with fewer parameters. The VGG-16 model has the highest number of parameters, with 138 million. Among the pretrained CNN models, the minimum number of parameters belongs to the GoogleNet model with 7 million. The proposed method includes a very low number of parameters

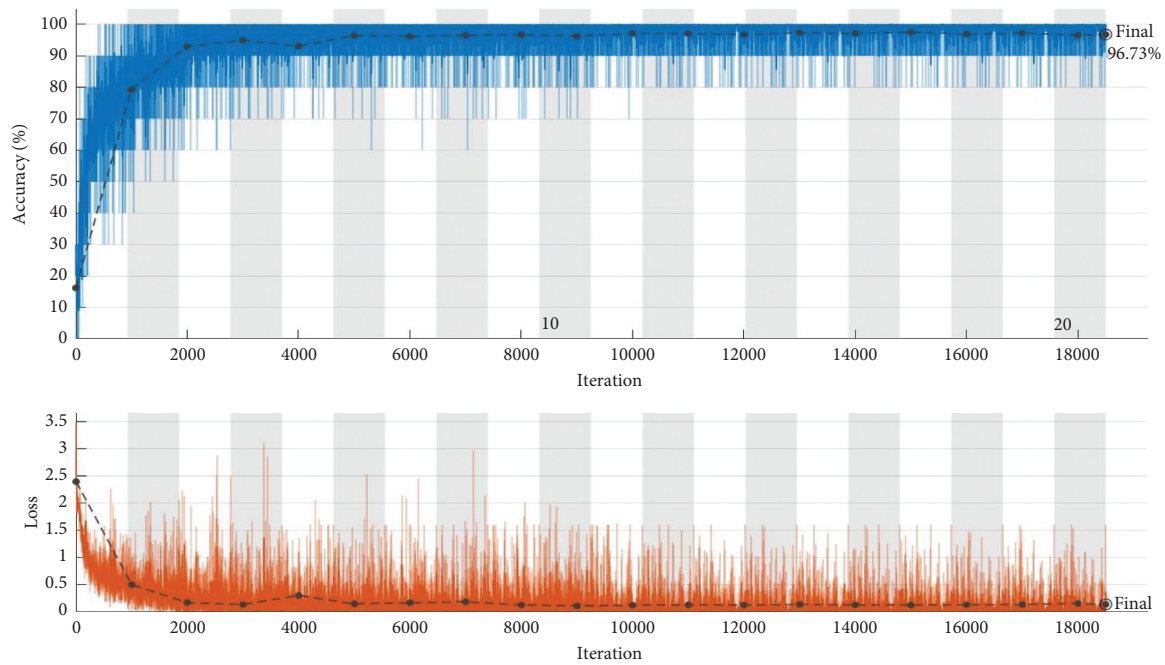


FIGURE 4: Training and testing process of the GoogleNet model.

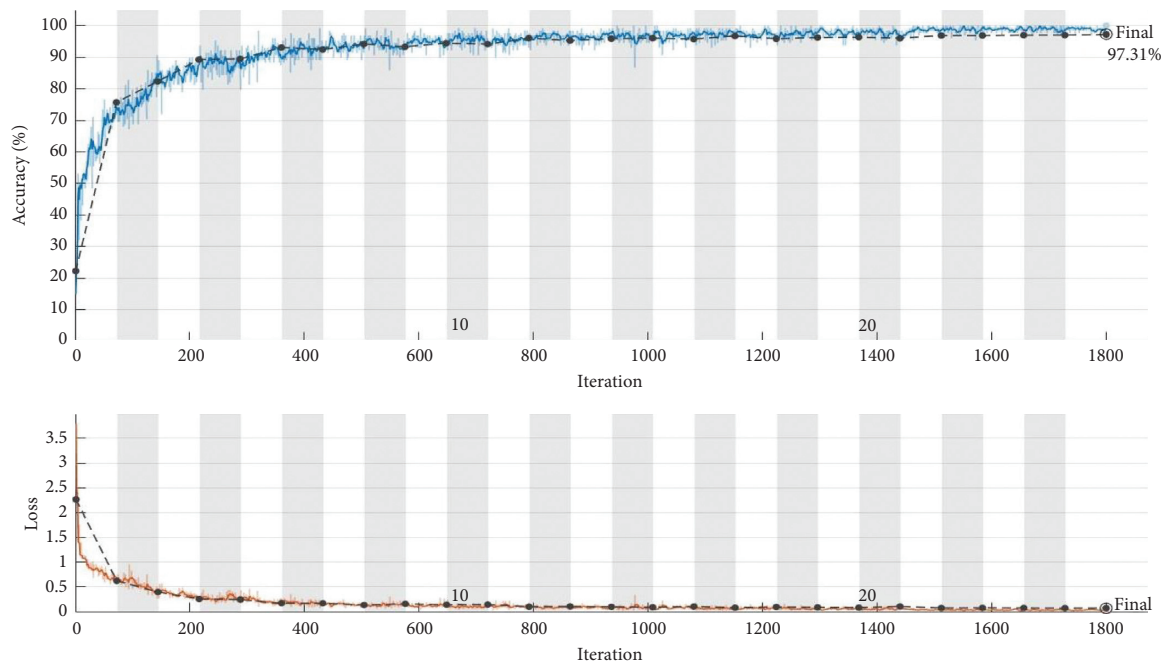


FIGURE 5: Training and testing process of the ResNet-50 model.

with 245284. The number of updated parameters is a factor that directly affects the training and test processes of the model.

Table 4 includes the results of the state-of-the-art. The lowest performance belongs to Khokhar et al. [34] with 86.86%. The highest performance is obtained with 97.94% accuracy by the FST + fuzzy DT method [33]. These datasets

do not contain any noise on PQDs. In our work, PQD datasets include 20–50 dB noise levels. This makes it a challenging problem for AI algorithms. On the contrary, the number of samples is too much than other previous studies. In conclusion, the proposed CNN + GRU algorithm has succeeded the highest performance with 98.44% within Table 4.

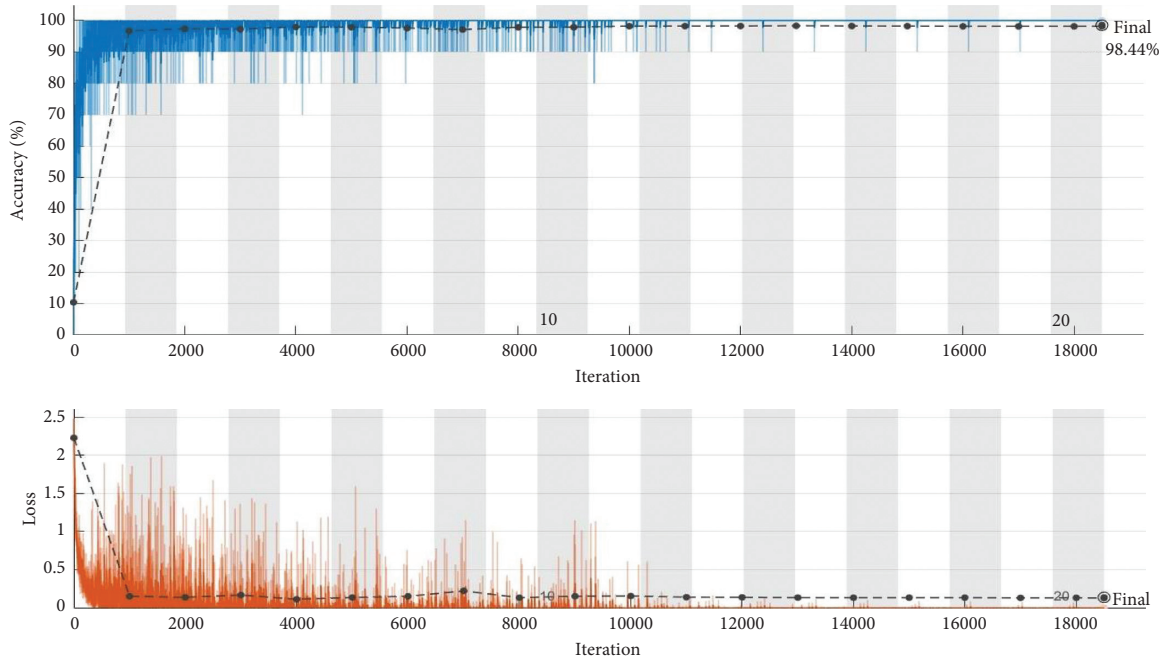


FIGURE 6: Training and testing process of the proposed CNN architecture.

TABLE 2: The performance comparison of deep CNN models in the detection of PQD.

Methods	Evaluation metrics (%)						
	ACC	SEN	SPE	PRE	F1-score	MCC	Kappa
VGG-16	97.83	97.80	99.64	97.83	97.81	97.45	91.13
GoogleNet	96.73	96.68	99.45	96.76	96.71	96.17	86.63
ResNet-50	97.31	97.25	99.55	97.26	97.25	96.81	89.01
Proposed method	98.44	98.45	99.74	98.45	98.45	98.19	93.64

Bold values indicate the better performance over the other approaches.

TABLE 3: Number of parameters for CNN models.

Methods	Parameters
ResNet-50	25.6 millions
VGG-16	138 millions
GoogleNet	7 millions
Proposed method	245284

TABLE 4: Performance comparison of different methods.

Literature	Method	Number of PQDs	Accuracy (%)
[32]	Bagging + rotation forest	5	95.33
[33]	FST + fuzzy DT	13	97.94
[34]	Wavelet + PNN	14	86.86
[35]	ST + DAG-SVM	9	97.77
Proposed method	CNN + GRU	7	98.44

4. Conclusions

In this study, a GRU-based CNN architecture is proposed for the identification of PQD defects. A CNN model has been designed that can analyze individual and composite disorders within PQD. PQD signals are formed into a

two-dimensional form by applying STFT. Despite the high level of noise in the PQD signals, a significant classification performance has been achieved. In this way, the proposed CNN model maintains its feasibility even in noisy environments and exhibits an adaptive feature. Of course, there are some shortcomings in the algorithm and the study, and it

is recommended that researchers who continue to work on this issue should analyze more PQD error classes. It can also measure error times.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] M. Mishra, "Power quality disturbance detection and classification using signal processing and soft computing techniques: a comprehensive review," *International Transactions on Electrical Energy Systems*, vol. 29, no. 8, 2019.
- [2] P. Khetarpal and M. M. Tripathi, "A critical and comprehensive review on power quality disturbance detection and classification," *Sustainable Computing: Informatics and Systems*, vol. 28, 2020.
- [3] C. I. Garcia, F. Grasso, A. Luchetta, M. C. Piccirilli, L. Paolucci, and G. Talluri, "A comparison of power quality disturbance detection and classification methods using CNN, LSTM and CNN-LSTM," *Applied Sciences*, vol. 10, no. 19, 2020.
- [4] B. Kiruthiga, R. Narmatha Banu, and D. Devaraj, "Detection and classification of power quality disturbances or events by adaptive NFS classifier," *Soft Computing*, vol. 24, no. 14, pp. 10351–10362, 2019.
- [5] M. Nour, H. Sindi, E. Abozinadah, Ş. Öztürk, and K. Polat, "A healthcare evaluation system based on automated weighted indicators with cross-indicators based learning approach in terms of energy management and cybersecurity," *International Journal of Medical Informatics*, vol. 144, Article ID 104300, 2020.
- [6] J. Wang, Z. Xu, and Y. Che, "Power quality disturbance classification based on DWT and multilayer perceptron extreme learning machine," *Applied Sciences*, vol. 9, no. 11, 2019.
- [7] M. Uyar, S. Yildirim, and M. T. Gencoglu, "An effective wavelet-based feature extraction method for classification of power quality disturbance signals," *Electric Power Systems Research*, vol. 78, no. 10, pp. 1747–1755, 2008.
- [8] T. Jayasree, D. Devaraj, and R. Sukanesh, "Power quality disturbance classification using Hilbert transform and RBF networks," *Neurocomputing*, vol. 73, no. 7–9, pp. 1451–1456, 2010.
- [9] M. B. I. Reaz, F. Choong, M. S. Sulaiman, F. Mohd-Yasin, and M. Kamada, "Expert system for power quality disturbance classifier," *IEEE Transactions on Power Delivery*, vol. 22, no. 3, pp. 1979–1988, 2007.
- [10] Z. Fengzhan and Y. Rengang, "Power quality disturbance recognition using S-transform," in *Proceedings of the 2006 IEEE Power Engineering Society General Meeting*, Montreal, Canada, 2006.
- [11] P. Janik and T. Lobos, "Automated classification of power-quality disturbances using SVM and RBF networks," *IEEE Transactions on Power Delivery*, vol. 21, no. 3, pp. 1663–1669, 2006.
- [12] M. Lopez-Ramirez, L. Ledesma-Carrillo, E. Cabal-Yepez, C. Rodriguez-Donate, H. Miranda-Vidales, and A. Garcia-Perez, "EMD-based feature extraction for power quality disturbance classification using moments," *Energies*, vol. 9, no. 7, 2016.
- [13] C.-Y. Lee and Y.-X. Shen, "Optimal feature selection for power-quality disturbances classification," *IEEE Transactions on Power Delivery*, vol. 26, no. 4, pp. 2342–2351, 2011.
- [14] B. K. Panigrahi and V. R. Pandi, "Optimal feature selection for classification of power quality disturbances using wavelet packet-based fuzzy k-nearest neighbour algorithm," *IET Generation, Transmission & Distribution*, vol. 3, no. 3, pp. 296–306, 2009.
- [15] U. Singh and S. N. Singh, "A new optimal feature selection scheme for classification of power quality disturbances based on ant colony framework," *Applied Soft Computing*, vol. 74, pp. 216–225, 2019.
- [16] N. Huang, G. Lu, G. Cai et al., "Feature selection of power quality disturbance signals with an entropy-importance-based random forest," *Entropy*, vol. 18, no. 2, 2016.
- [17] H. Dehghani, B. Vahidi, R. A. Naghizadeh, and S. H. Hosseinian, "Power quality disturbance classification using a statistical and wavelet-based Hidden Markov model with Dempster-Shafer algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 47, pp. 368–377, 2013.
- [18] T. K. Abdel-Galil, M. Kamel, A. M. Youssef, E. F. El-Saadany, and M. M. A. Salama, "Power quality disturbance classification using the inductive inference approach," *IEEE Transactions on Power Delivery*, vol. 19, no. 4, pp. 1812–1818, 2004.
- [19] Z. Gang, S. Mei-Xiang, L. Ding, Y. Jian, and M. Zhu-Mei, "Power quality disturbance classification based on rule-based and wavelet-multi-resolution decomposition," in *Proceedings of the International Conference on Machine Learning and Cybernetics*, pp. 2137–2141, Beijing, China, 2002.
- [20] H. Wang, P. Wang, and T. Liu, "Power quality disturbance classification using the S-transform and probabilistic neural network," *Energies*, vol. 10, no. 1, 2017.
- [21] C. A. Naik and P. Kundu, "Power quality disturbance classification employing S-transform and three-module artificial neural network," *International Transactions on Electrical Energy Systems*, vol. 24, no. 9, pp. 1301–1322, 2014.
- [22] D. D. Ferreira, J. M. d. Seixas, and A. S. Cerqueira, "A method based on independent component analysis for single and multiple power quality disturbance classification," *Electric Power Systems Research*, vol. 119, pp. 425–431, 2015.
- [23] N. M. Khoa, D. T. Viet, and N. H. Hieu, "Classification of power quality disturbances using wavelet transform and K-nearest neighbor classifier," in *Proceedings of the 2013 IEEE International Symposium on Industrial Electronics*, Taipei, Taiwan, 2013.
- [24] Ş. Öztürk and U. Özkaya, "Gastrointestinal tract classification using improved LSTM based CNN," *Multimedia Tools and Applications*, vol. 79, no. 39–40, pp. 28825–28840, 2020.
- [25] S. Wang and H. Chen, "A novel deep learning method for the classification of power quality disturbances using deep convolutional neural network," *Applied Energy*, vol. 235, pp. 1126–1140, 2019.
- [26] H. Liu, F. Hussain, Y. Shen, S. Arif, A. Nazir, and M. Abubakar, "Complex power quality disturbances classification via curvelet transform and deep learning," *Electric Power Systems Research*, vol. 163, pp. 1–9, 2018.
- [27] K. Cai, W. Cao, L. Aarniovuori, H. Pang, Y. Lin, and G. Li, "Classification of power quality disturbances using Wigner-Ville distribution and deep convolutional neural networks," *IEEE Access*, vol. 7, pp. 119099–119109, 2019.

- [28] 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 Transactions on Industrial Informatics*, vol. 15, no. 8, pp. 4481–4493, 2019.
- [29] Y. Shen, M. Abubakar, H. Liu, and F. Hussain, "Power quality disturbance monitoring and classification based on improved PCA and convolution neural network for wind-grid distribution systems," *Energies*, vol. 12, no. 7, 2019.
- [30] M. A. Rodriguez, J. Felipe Sotomonte, J. Cifuentes, and M. Bueno-Lopez, "Power quality disturbance classification via deep convolutional auto-encoders and stacked LSTM recurrent neural networks," in *Proceedings of the 2020 International Conference on Smart Energy Systems and Technologies (SEST)*, Istanbul, Turkey, 2020.
- [31] U. Subudhi and S. Dash, "Detection and classification of power quality disturbances using GWO ELM," *Journal of Industrial Information Integration*, vol. 22, Article ID 100204, 2021.
- [32] D. A. Bashawyah and A. Subasi, "Power quality event detection using FAWT and bagging ensemble classifier," in *Proceedings of the 2019 IEEE International Conference on Environment and Electrical Engineering and 2019 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)*, Genova, Italy, 2019.
- [33] M. Biswal and P. K. Dash, "Measurement and classification of simultaneous power signal patterns with an S-transform variant and fuzzy decision tree," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 4, pp. 1819–1827, 2013.
- [34] S. Khokhar, A. A. Mohd Zin, A. P. Memon, and A. S. Mokhtar, "A new optimal feature selection algorithm for classification of power quality disturbances using discrete wavelet transform and probabilistic neural network," *Measurement*, vol. 95, pp. 246–259, 2017.
- [35] J. Li, Z. Teng, Q. Tang, and J. Song, "Detection and classification of power quality disturbances using double resolution S-transform and DAG-SVMs," *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 10, pp. 2302–2312, 2016.
- [36] Y. Xu and T. T. Qiu, "Human activity recognition and embedded application based on convolutional neural network," *Journal of Artificial Intelligence and Technology*, vol. 1, no. 1, pp. 51–60, 2021.
- [37] D. Singh, V. Kumar, V. Yadav, and M. Kaur, "Deep neural network-based screening model for COVID-19-infected patients using chest X-ray images," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 35, no. 3, Article ID 2151004, 2021.
- [38] D. Jiang, G. Hu, G. Qi, and N. Mazur, "A fully convolutional neural network-based regression approach for effective chemical composition analysis using near-infrared spectroscopy in cloud," *Journal of Artificial Intelligence and Technology*, vol. 1, no. 1, pp. 74–82, 2021.
- [39] M. Kaur and D. Singh, "Multi-modality medical image fusion technique using multi-objective differential evolution based deep neural networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 2, pp. 2483–2493, 2021.
- [40] S. Ghosh, P. Shivakumara, P. Roy, U. Pal, and T. Lu, "Graphology based handwritten character analysis for human behaviour identification," *CAAI Transactions on Intelligence Technology*, vol. 5, no. 1, pp. 55–65, 2020.
- [41] M. Kaur and D. Singh, "Multiobjective evolutionary optimization techniques based hyperchaotic map and their applications in image encryption," *Multidimensional Systems and Signal Processing*, vol. 32, no. 1, pp. 281–301, 2021.
- [42] Ş. Öztürk, "Stacked auto-encoder based tagging with deep features for content-based medical image retrieval," *Expert Systems with Applications*, vol. 161, Article ID 113693, 2020.
- [43] R. Zhao, D. Wang, R. Yan, K. Mao, F. Shen, and J. Wang, "Machine health monitoring using local feature-based gated recurrent unit networks," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 2, pp. 1539–1548, 2018.
- [44] B. Gupta, M. Tiwari, and S. Lamba, "Visibility improvement and mass segmentation of mammogram images using quantile separated histogram equalisation with local contrast enhancement," *CAAI Transactions on Intelligence Technology*, vol. 4, no. 2, pp. 73–79, 2019.
- [45] M. Kaur, D. Singh, and V. Kumar, "Color image encryption using minimax differential evolution-based 7D hyper-chaotic map," *Applied Physics B*, vol. 126, no. 9, pp. 1–19, 2020.
- [46] H. S. Basavegowda and G. Dagnev, "Deep learning approach for microarray cancer data classification," *CAAI Transactions on Intelligence Technology*, vol. 5, no. 1, pp. 22–33, 2020.
- [47] G. Hu, S.-H. K. Chen, and N. Mazur, "Deep neural network-based speaker-aware information logging for augmentative and alternative communication," *Journal of Artificial Intelligence and Technology*, vol. 1, no. 2, pp. 138–143, 2021.
- [48] N. Gianchandani, A. Jaiswal, D. Singh, V. Kumar, and M. Kaur, "Rapid COVID-19 diagnosis using ensemble deep transfer learning models from chest radiographic images," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–13, 2020.