

Article

POA-Optimized 1D-CNN with Channel Attention for Power Quality Disturbance Classification Under Strong Noise Conditions

Fulin Gong ^{1,*} , Ruisheng Diao ¹, Chao Cai ² and Jun Han ²

¹ Zhejiang University–University of Illinois Urbana-Champaign Institute (ZJU-UIUC Institute), Zhejiang University, Haining 314400, China

² Economics and Technology Research Institute, State Grid Jiangsu Electric Power Company, Nanjing 210000, China; caic@js.sgcc.com.cn (C.C.); hanj@js.sgcc.com.cn (J.H.)

* Correspondence: fulin.23@intl.zju.edu.cn

Abstract

Power quality disturbance (PQD) classifiers can lose 30–50 percentage points of accuracy when the signal-to-noise ratio (SNR) drops to 5 dB. This study finds that the degradation arises primarily not from architectural limitations but from hyperparameter configurations that fail to adapt across noise levels, within the IEEE 1159 disturbance set and additive-noise conditions studied here. Building on this insight, we propose a one-dimensional convolutional neural network (1D-CNN) with channel attention, whose three key hyperparameters—the learning rate, the first-layer convolutional kernel size, and the attention reduction ratio—are automatically optimized using the Phong Optimization Algorithm (POA). A 12-class synthetic dataset constructed to IEEE 1159-2019, with noise levels from noise-free to 5 dB SNR, is used for training and evaluation. Under the extreme condition of 5 dB SNR, the proposed method achieves 83.4% accuracy, outperforming a support vector machine (SVM) with discrete wavelet transform features (66.30%), a hybrid CNN–long short-term memory network (CNN-LSTM; 43.32%), and plain 1D-CNN (38.39%) under their commonly reported configurations. When every deep-learning baseline receives the same POA hyperparameter optimization under a fair per-SNR protocol, this advantage largely disappears: at 5 dB SNR all POA-optimized deep methods fall within roughly 5 percentage points (82.31–87.13%), and the attention module’s own contribution shrinks to within run-to-run variation (the same proposed model scoring 82.77% with attention vs. 82.48% without), showing that systematic hyperparameter optimization, not architectural novelty, drives the noise robustness. A single-set ablation under the original 10 dB-optimized configuration points the same way (POA optimization alone raising 5 dB accuracy from 41.6% to 76.4%, with channel attention adding a further 7 points). A comparison against classical threshold-index classifiers on the identical dataset shows the same pattern at the level of hand-crafted features: with fixed clean-calibrated thresholds, the index classifier collapses from 95.7% to 17.5% at 5 dB SNR, while per-noise-level re-calibration of the same indices recovers 83.8%. Furthermore, the POA-optimized hyperparameters were validated on an independent public PQD dataset, achieving 90.00% accuracy when training a fresh model from scratch. It also stays robust under more realistic complex noise (80.9% at 5 dB SNR) and, on a two-class real-measured probe, transfers to field signals with only light calibration. These findings suggest that for noise-robust PQD classification, hyperparameter optimization deserves as much attention as the architectural design itself, rather than being treated as a final tuning step.



Academic Editor: José Gabriel Oliveira Pinto

Received: 18 June 2026

Revised: 13 July 2026

Accepted: 23 July 2026

Published: 26 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](https://creativecommons.org/licenses/by/4.0/)

[Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: power quality disturbance classification; 1D convolutional neural network; channel attention; Phong Optimization Algorithm; hyperparameter optimization; noise robustness

1. Introduction

PQD is a prevalent practical problem in modern power system operation [1,2], which can cause serious damage to the equipment and industrial processes. Therefore, accurate classification of PQDs is crucial for power grid monitoring and fault diagnosis [3]. However, signals in real power systems always come with noise, and when the SNR is low, for example 5 dB SNR, the accuracy of current classification methods will drop significantly [4].

Conventional PQD classification works in two steps: first, signal processing methods such as the discrete wavelet transform (DWT), S-transform [5], and the Hilbert–Huang transform [6] are applied to extract time-frequency features from the raw signal, as reviewed in recent comprehensive surveys [7]; these features are then fed into classifiers such as the support vector machine (SVM) [8] or ensemble methods such as random forest [9]. These methods perform well when noise is mild. However, handcrafted features are sensitive to noise—once the SNR drops to a low level, they lose the ability to distinguish different disturbance patterns [10]. Moreover, the choice of transform and the subsequent feature engineering are largely manual and problem-specific: a feature set tuned for one disturbance type or noise level rarely transfers to another, which limits the robustness and scalability of these pipelines and motivates the move toward representations learned directly from the raw signal. This also clarifies why learned classifiers are needed even when classical threshold-based power-quality indices (e.g., RMS deviation, THD, or flicker severity) are readily available: such indices flag well-separated single disturbances under mild noise, but their fixed thresholds lose discriminative power for compound disturbances and at low SNR, where index values from different disturbance types overlap—precisely the strong-noise, compound-disturbance regime targeted in this work. This qualitative motivation is now confirmed numerically in Section 3.6.6: a classifier built on these classical indices with fixed, clean-calibrated thresholds collapses from 95.7% to 17.5% accuracy at 5 dB SNR (12-class chance level: 8.3%), whereas the learned classifier retains 82.8%; the same section also reports, for transparency, the behavior of the indices when their thresholds are re-calibrated per noise level.

In recent years, deep learning methods have been increasingly applied to PQD classification, mainly due to their ability to learn features directly from raw signals rather than relying on hand-designed ones. 1D-CNN performs well in extracting local spatial features from PQD waveforms [11,12]. Recent work has explored several variants of this paradigm: bidirectional long short-term memory (LSTM) with attention mechanisms [13], image-based approaches transforming 1D signals into 2D representations such as Gram angular fields followed by deep CNN architectures [14], 2D wavelet scalograms with deep learning [15], ensemble deep CNN approaches using image-based transfer learning [16], and cross-attention fusion of temporal and spatial features [17]. CNN-LSTM has also been applied to capture temporal dependencies in PQD signals [18,19]. Later, attention mechanisms were introduced—a typical example is SENet, which automatically adjusts the weights of different channel features [20,21]. However, all of these models face a common problem: they are sensitive to hyperparameter settings, and a minor difference in the learning rate or kernel size can lead to a large performance gap. Most existing studies still rely on manual tuning or grid search, which is time-consuming and likely to miss the

optimal solution, especially when the noise level varies [22,23]. A further, often-overlooked limitation is that improvements in this literature are typically attributed to the proposed architecture, yet the architectural change and the accompanying hyperparameter tuning are seldom disentangled. It therefore remains unclear whether reported gains—particularly under strong noise—stem from a genuinely better architecture or simply from more careful optimization. This conflation of architectural and optimization effects is the specific research gap addressed in this work.

Consequently, for hyperparameter optimization, metaheuristic algorithms are adopted in this paper due to their strong ability to search for optimal solutions in complex, non-convex parameter spaces. Metaheuristic algorithms such as particle swarm optimization, genetic algorithms, and the sparrow search algorithm [24] have been widely applied to CNN hyperparameter optimization, including recent demonstrations on 1D-CNN architectures [25] and broader systematic reviews [22,23,26,27]. However, these algorithms have not been extensively explored under strong noise conditions. In addition, these studies generally evaluate the optimizer under standard or low-noise settings and present the specific metaheuristic as the methodological novelty, leaving open the more basic question of whether the performance gains arise from the particular optimizer or simply from performing systematic hyperparameter optimization at all. The Phong Optimization Algorithm (POA) [28] is a recently proposed metaheuristic inspired by a graphics rendering model, structuring its search into three components that shift the balance from broad exploration toward focused exploitation as iterations proceed. As far as we know, POA has not been applied to hyperparameter optimization for PQD classification. We adopt POA as a representative, competitive metaheuristic rather than claiming it is inherently superior to alternatives; as shown in Section 3.6.3, POA, particle swarm optimization, and random search reach statistically indistinguishable accuracy under matched budgets, and POA is retained mainly for its lower run-to-run variance and faster convergence.

Moreover, existing studies rarely validate whether the optimized hyperparameters can generalize across different datasets, leaving the practical transferability of the optimization results unexamined.

In this paper, a POA-optimized 1D-CNN with channel attention mechanism is proposed for robust PQD classification under strong noise conditions. This work makes the following contributions. The contributions are organized into one methodological contribution (item 1), one empirical finding (item 2), and three validation results (items 3–5). The 1D-CNN backbone and the SENet-style channel attention module are deliberately conventional, lightweight components, chosen as a representative testbed so that the central finding is attributable to the optimization process itself rather than to any architectural novelty.

1. We propose a lightweight 1D-CNN with channel attention (17,308 parameters) for 12-class PQD classification including four compound disturbance types, and instantiate hyperparameter optimization using the Phong Optimization Algorithm (POA) over the learning rate, first-layer kernel size, and attention reduction ratio. Unlike most papers that treat the proposed architecture as the main contribution, our ablation study (Section 3.5), optimizer-fair comparison (Section 3.6.3), and architecture-fair comparison (Section 3.6.2) all point to the same thing: what really drives noise-robust PQD classification is the hyperparameter optimization process itself, not architectural novelty or which metaheuristic we use.
2. The proposed method achieves 83.4% accuracy at 5 dB SNR, outperforming SVM+DWT, CNN-LSTM, and plain 1D-CNN under their commonly reported configurations. The fairness analysis pins this down: when the same per-SNR POA optimization is granted to all deep-learning baselines (Section 3.6.2), the 5 dB accuracy values of all POA-optimized deep methods fall within roughly 5 percentage

points (82.31–87.13%), with MobileNet-1D (6920 parameters) reaching 86.35% and 1D-ResNet (46,548 parameters) reaching 87.13% and the attention module's own contribution falling to within run-to-run variation (82.77% vs. 82.48%). The large accuracy gaps reported across much of the PQD literature therefore likely come from unequal hyperparameter tuning rather than from any inherent architectural advantage. The single-set ablation study (Section 3.5) points the same way, with POA optimization alone raising 5 dB accuracy from 41.6% to 76.4% under the transferred 10 dB-optimized configuration. This indicates that for strong-noise PQD classification, hyperparameter selection deserves more attention.

3. We validate the POA-optimized hyperparameters on an independent public dataset, achieving 90.00% accuracy when training a fresh model from scratch. This indicates that the optimized configuration remains effective when the network is retrained on data with a fundamentally different generation procedure—a hyperparameter transferability rarely tested in prior PQD studies; the precise scope of this claim is discussed in Section 3.8.
4. We further evaluate the proposed method under six complex noise scenarios beyond AWGN—including harmonic background, interharmonics, impulsive transients, colored noise, quantization noise, and a composite realistic scenario jointly applying the harmonic, interharmonic, impulsive, AWGN, and quantization mechanisms. The method maintains 80.9% accuracy at 5 dB SNR under the composite condition, confirming that the POA-optimized hyperparameters remain robust under noise conditions more realistic than pure AWGN.
5. We provide a preliminary probe on real measured signals using two real-life datasets recorded on an actual distribution network (real voltage-sag and impulsive-transient recordings, 68 events in total). Although the synthetic-trained classifier head does not transfer zero-shot, the learned feature representation does: a linear probe separates the two available real classes perfectly (100% five-fold cross-validation) and few-shot calibration with eight labeled examples per class reaches 99.8% accuracy, indicating that the domain gap resides in the classifier-head calibration and that only lightweight field calibration is required (Section 3.9).

Scope and applicability. This work focuses on the 12-class disturbance taxonomy defined in IEEE Standard 1159-2019, sampled at $f_s = 2560$ Hz to cover the canonical low-frequency PQDs (sag, swell, harmonics up to the 7th order, interruption, flicker, notch, spike, and their compounds). We acknowledge that emerging power-electronic interfaces introduce additional supraharmmonic phenomena up to 150 kHz that require sampling rates of hundreds of kSa/s in modern PQ analyzers, and that real distribution networks exhibit non-stationary transitions absent from the IEEE 1159 parametric formulation. Extension of the proposed POA-optimization framework to higher-bandwidth supraharmonic regimes and non-stationary event mixtures is outside the scope of this study, but represents a natural direction for future work. The contribution of this paper is methodological—demonstrating that systematic hyperparameter optimization dominates architectural choice for noise-robust classification within the canonical IEEE 1159 disturbance set—rather than dataset-specific.

The rest of this paper is structured in three sections. Section 2 introduces the theoretical background, including the PQD signal model, the complex noise modeling beyond AWGN, the 1D-CNN with channel attention architecture, and the POA algorithm. Section 3 presents the simulation results and discussion. Section 4 draws conclusions and points out future research directions.

2. Theoretical Background

2.1. PQD Signal Mathematical Model

Power quality disturbances can be mathematically modeled as deviations from the ideal sinusoidal voltage waveform. The fundamental pure sinusoidal signal is expressed as

$$v(t) = A \sin(2\pi ft) \quad (1)$$

where A is the amplitude and f is the fundamental frequency (50 Hz in this study). Based on the IEEE 1159-2019 standard [1], 12 types of PQDs are defined in this work, including 8 single disturbances and 4 compound disturbances. The mathematical models are summarized in Table 1.

Table 1. Mathematical models of 12 types of PQDs based on IEEE 1159-2019.

Type	Mathematical Model	Parameters
C1: Normal	$v(t) = \sin(2\pi ft)$	–
C2: Sag	$v(t) = [1 - \alpha \cdot g(t)] \sin(2\pi ft)$	$\alpha \in [0.1, 0.5]$
C3: Swell	$v(t) = [1 + \alpha \cdot g(t)] \sin(2\pi ft)$	$\alpha \in [0.1, 0.5]$
C4: Harmonics	$v(t) = \sin(2\pi ft) + \sum_{h \in \{3,5,7\}} a_h \sin(2\pi hft)$	$a_3 \in [0.05, 0.15], a_5 \in [0.05, 0.12], a_7 \in [0.01, 0.05]$
C5: Interruption	$v(t) = [1 - \alpha \cdot g(t)] \sin(2\pi ft)$	$\alpha \in [0.9, 0.98]$
C6: Flicker	$v(t) = [1 + \beta \sin(2\pi f_f t)] \sin(2\pi ft)$	$\beta \in [0.08, 0.2], f_f \in [5, 15] \text{ Hz}$
C7: Notch	$v(t) = \sin(2\pi ft) - d v(t) $ at zero-crossings	$d \in [0.4, 0.8], \text{width} \in [0.5, 2] \text{ ms}$
C8: Spike	$v(t) = \sin(2\pi ft) + \sum_k A_k \delta_w(t - t_k)$	$A_k \in [1.0, 3.0], n \in [3, 8], \text{width} \in [0.1, 0.5] \text{ ms}$
C9: Sag + Harm.	Combination of C2 and C4	$\alpha \in [0.1, 0.5], a_3, a_5 \text{ as C4}$
C10: Swell + Harm.	Combination of C3 and C4	$\alpha \in [0.1, 0.5], a_3, a_5 \text{ as C4}$
C11: Sag + Flicker	Combination of C2 and C6	$\alpha \in [0.1, 0.5], \beta \in [0.08, 0.2], f_f = 8 \text{ Hz}$
C12: Int. + Harm.	Combination of C5 and C4	$\alpha \in [0.9, 0.98], a_3, a_5 \text{ as C4}$

$g(t) = u(t - t_1) - u(t - t_2)$, where $u(\cdot)$ is the unit step function; t_1, t_2 are randomized disturbance start/end times; $\delta_w(\cdot)$ denotes a rectangular pulse of width w ; $f = 50 \text{ Hz}$, $f_s = 2560 \text{ Hz}$.

For single disturbances, voltage sag is modeled by reducing the amplitude within a time interval $[t_1, t_2]$ by a factor $\alpha \in [0.1, 0.5]$, while voltage swell increases the amplitude by the same range. Harmonics are generated by superimposing the 3rd, 5th, and 7th harmonic components with amplitudes $a_3 \in [0.05, 0.15]$, $a_5 \in [0.05, 0.12]$, and $a_7 \in [0.01, 0.05]$, respectively. Interruption is modeled similarly to sag but with a much larger reduction factor $\alpha \in [0.9, 0.98]$, representing near-complete loss of voltage. Flicker is simulated by amplitude modulation with a low-frequency sinusoidal envelope at frequency $f_{\text{flicker}} \in [5, 15] \text{ Hz}$ and modulation depth $\beta \in [0.08, 0.2]$. Notch disturbances are created by introducing periodic dips at zero-crossing points, while spike disturbances are modeled as randomly positioned high-amplitude transient impulses.

For compound disturbances, each type is constructed by combining two single disturbances: sag with harmonics, swell with harmonics, sag with flicker, and interruption with harmonics. These four combinations were selected because they are the compound disturbances most frequently encountered and most commonly benchmarked in the PQD classification literature: harmonic pollution coexisting with voltage sags, swells, or interruptions reflects nonlinear loads operating during voltage-magnitude events, while the sag-with-flicker pair captures the joint amplitude-modulation and magnitude-reduction case. They also span the three single-disturbance families (magnitude events, harmonic distortion, and amplitude modulation), giving broad coverage of compound behavior without enumerating the full combinatorial space. All disturbance parameters are randomized within the IEEE 1159 specified ranges to enhance data diversity. For completeness,

the harmonic content is summarized as follows. The standalone harmonic class superimposes the 3rd, 5th, and 7th harmonics on the fundamental, with per-unit amplitudes (relative to the fundamental) drawn from $[0.05, 0.15]$, $[0.05, 0.12]$, and $[0.01, 0.05]$, respectively, corresponding to a total harmonic distortion (THD) of approximately 7–20%; the three harmonic-containing compound classes (C9, C10, C12) use the 3rd and 5th harmonics only (THD approximately 7–19%). All harmonic components have constant amplitude over the active interval of each signal, so every modeled disturbance is locally stationary within its active window; non-stationary transitions and time-varying spectra are not modeled in this study and are identified as future work (Section 4). The complete per-class parametric models and their randomized parameter ranges are listed in Table 1.

To simulate real-world noise conditions, additive white Gaussian noise (AWGN) is added to each signal:

$$x(t) = v(t) + n(t) \quad (2)$$

where $n(t)$ is Gaussian noise with power determined by the target signal-to-noise ratio (SNR). In this study, the signals are sampled at $f_s = 2560$ Hz with 2560 sampling points per signal, corresponding to one second of recording duration. Five noise levels are considered: noise-free, 30 dB, 20 dB, 10 dB, and 5 dB. Representative waveforms of the 12 disturbance types at 20 dB SNR are shown in Figure 1.

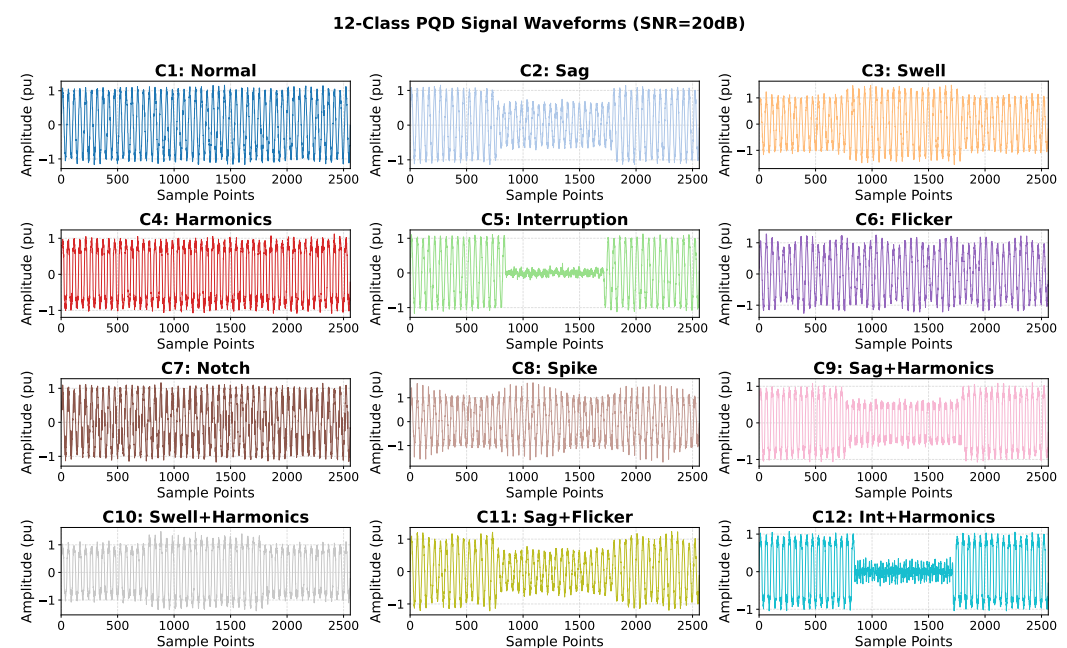


Figure 1. Waveform examples of 12 types of PQDs at 20 dB SNR.

2.2. Complex Noise Modeling

While additive white Gaussian noise (AWGN) is widely used to emulate measurement noise in PQD studies, real distribution networks exhibit far more complex noise characteristics, including persistent harmonic background, interharmonics from power-electronic devices, impulsive transients from switching events, colored noise with non-white power spectral density, and quantization noise introduced by the finite resolution of acquisition hardware. To evaluate the robustness of the proposed method under conditions closer to practical operation, six noise scenarios beyond pure AWGN are additionally modeled and used in Section 3.7.

(1) Harmonic background noise. Low-order harmonics (3rd, 5th, 7th) with randomized phases are superimposed on the clean signal, together with AWGN:

$$x(t) = v(t) + \sum_{h \in \{3,5,7\}} a_h \sin(2\pi h f t + \varphi_h) + n_{\text{awgn}}(t), \quad (3)$$

where $\varphi_h \sim \mathcal{U}(0, 2\pi)$ and a_h are randomized within typical pollution levels reported in field measurements ($a_3 \in [0.03, 0.08]$, $a_5 \in [0.02, 0.06]$, $a_7 \in [0.01, 0.04]$).

(2) Interharmonic noise. A non-integer harmonic component is introduced to emulate interference from power-electronic devices such as variable-frequency drives:

$$x(t) = v(t) + a_{ih} \sin(2\pi f_{ih} t + \varphi) + n_{\text{awgn}}(t), \quad (4)$$

where $f_{ih} = \beta f$ with $\beta \in [2.3, 4.7]$ (non-integer multiple of the fundamental frequency), and $a_{ih} \in [0.03, 0.10]$.

(3) Impulsive noise. Randomly located high-amplitude transient pulses are added to model switching operations and lightning-induced disturbances:

$$x(t) = v(t) + \sum_{k=1}^K A_k \cdot p_w(t - t_k) + n_{\text{awgn}}(t), \quad (5)$$

where p_w denotes a rectangular pulse of random width w , $K \in [5, 15]$ is the number of impulses, $A_k \in [0.5, 1.5]$, and t_k are uniformly distributed within the signal duration.

(4) Colored (pink) noise. Noise with power spectral density proportional to $1/f$ replaces white noise, which better approximates the broadband background of real grids:

$$x(t) = v(t) + n_{\text{pink}}(t), \quad (6)$$

where $n_{\text{pink}}(t)$ is generated by spectrally shaping a white Gaussian sequence using a $1/\sqrt{f}$ filter and scaling its power to match the target SNR.

(5) Quantization noise. The signal is corrupted by AWGN and then quantized to b bits to emulate the finite resolution of analog-to-digital converters (ADCs):

$$x_q(t) = Q_b(v(t) + n_{\text{awgn}}(t)), \quad (7)$$

where $Q_b(\cdot)$ denotes uniform quantization to b bits. In this study, $b = 8$ is used, representing a low-resolution ADC commonly found in low-cost monitoring devices.

(6) Composite realistic noise. Harmonic background, interharmonics, impulsive transients, AWGN, and 8-bit quantization are jointly applied to the clean signal, representing the most demanding, field-like scenario in which multiple noise sources coexist:

$$x_{\text{comp}}(t) = Q_8(v(t) + n_{\text{harm}}(t) + n_{ih}(t) + n_{\text{imp}}(t) + n_{\text{awgn}}(t)). \quad (8)$$

In all six scenarios, the AWGN component is calibrated to a target SNR level (10 dB or 5 dB) as in the baseline simulations, ensuring that the noise power is directly comparable across scenarios. The classification performance under these six conditions is reported in Section 3.7.

We emphasize that the parametric IEEE 1159 generator adopted here does *not* reproduce several signal features characteristic of real low-voltage distribution networks. In particular, the diode-bridge input stages of switched-mode power supplies produce a flat-topped (“clipped sine”) voltage, this distortion is frequently amplitude-modulated by neighboring power-electronic loads, and the combination yields wideband harmonic and interharmonic content that can extend to tens or hundreds of kHz, i.e., supraharmonic

components up to 150 kHz that require sampling rates of hundreds of kSa/s to capture. These mechanisms, together with the non-stationary transitions and overlapping events typical of field signals, are outside the scope of the present idealized generator; they are acknowledged here explicitly as the boundary of applicability, and the synthetic-to-real gap they induce is examined directly on real measured recordings in Section 3.9.

2.3. 1D-CNN with Channel Attention

The proposed classification model consists of three main components: a 1D convolutional feature extractor, a channel attention module, and a global average pooling classifier. The overall architecture is illustrated in Figure 2.

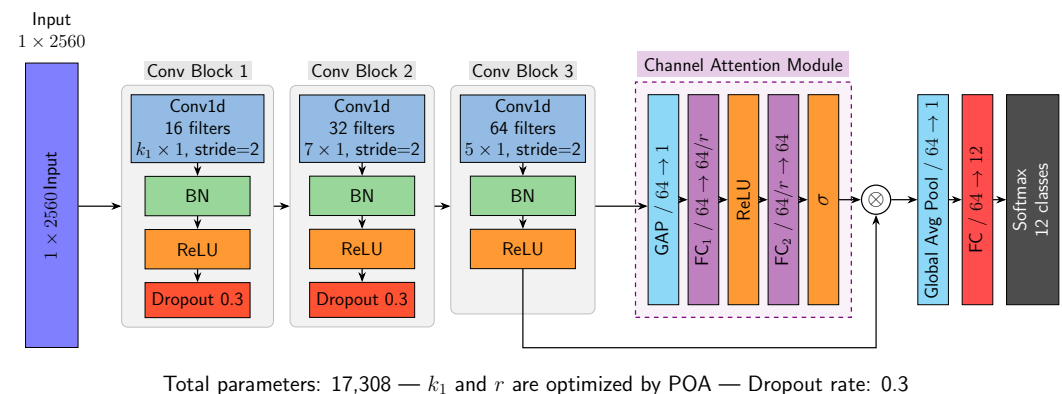


Figure 2. Architecture of the proposed 1D-CNN with channel attention model.

1D Convolutional Feature Extractor. The backbone network comprises three cascaded 1D convolutional layers with increasing filter numbers (16, 32, and 64), which progressively extract hierarchical features from the raw PQD waveforms. Each convolutional layer is followed by batch normalization (BN) [29] and ReLU activation. Dropout [30] with a rate of 0.3 is applied after the first two convolutional layers to mitigate overfitting. The first convolutional layer uses a kernel size k_1 (optimized by POA) with stride 2, while the second and third layers use fixed kernel sizes of 7 and 5, respectively, both with stride 2. This design progressively reduces the temporal resolution while expanding the feature dimensionality. The output of the i -th convolutional layer can be expressed as:

$$F_i = \text{Dropout}(\text{ReLU}(\text{BN}(\text{Conv1D}(F_{i-1})))) \quad (9)$$

where $F_0 = x \in \mathbb{R}^{1 \times 2560}$ is the input signal.

Channel Attention Module. After the three convolutional layers, a channel attention mechanism inspired by the squeeze-and-excitation network (SENet) [20] is applied to adaptively recalibrate the channel-wise feature responses. The module operates as follows. First, global average pooling (GAP) is applied along the temporal dimension to squeeze the feature map $F \in \mathbb{R}^{C \times L}$ into a channel descriptor $z \in \mathbb{R}^C$:

$$z_c = \frac{1}{L} \sum_{l=1}^L F_c(l) \quad (10)$$

Then, a two-layer fully connected network with a bottleneck structure is used to learn the channel-wise attention weights:

$$s = \sigma(W_2 \cdot \text{ReLU}(W_1 \cdot z)) \quad (11)$$

where $W_1 \in \mathbb{R}^{(C/r) \times C}$ and $W_2 \in \mathbb{R}^{C \times (C/r)}$ are the weight matrices of the two fully connected layers, r is the reduction ratio (optimized by POA), and σ denotes the sigmoid

activation function. The final re-calibrated feature map is obtained by channel-wise multiplication:

$$\tilde{F} = s \otimes F \quad (12)$$

where $\otimes$ denotes channel-wise multiplication.

Classification Head. The re-calibrated feature map $\tilde{F}$ is then passed through a global average pooling layer to produce a fixed-length feature vector of dimension $C = 64$, followed by a fully connected layer with softmax activation for 12-class classification. The total number of trainable parameters is 17,308, making the model highly lightweight.

2.4. Phong Optimization Algorithm

The Phong Optimization Algorithm (POA) [28] is a swarm intelligence optimization algorithm. It comes from a computer graphics model named the Phong reflection model [31]. POA treats each candidate solution as a light particle, using light propagation and surface reflection to guide the search. Each light particle's position represents a candidate solution, with its intensity and velocity determining the search direction and step size. When light particles are far from the global optimum, they move at higher velocity to search broadly; as they approach the optimum, the velocity decreases and the search becomes more focused. The complete procedure is summarized in Figure 3.

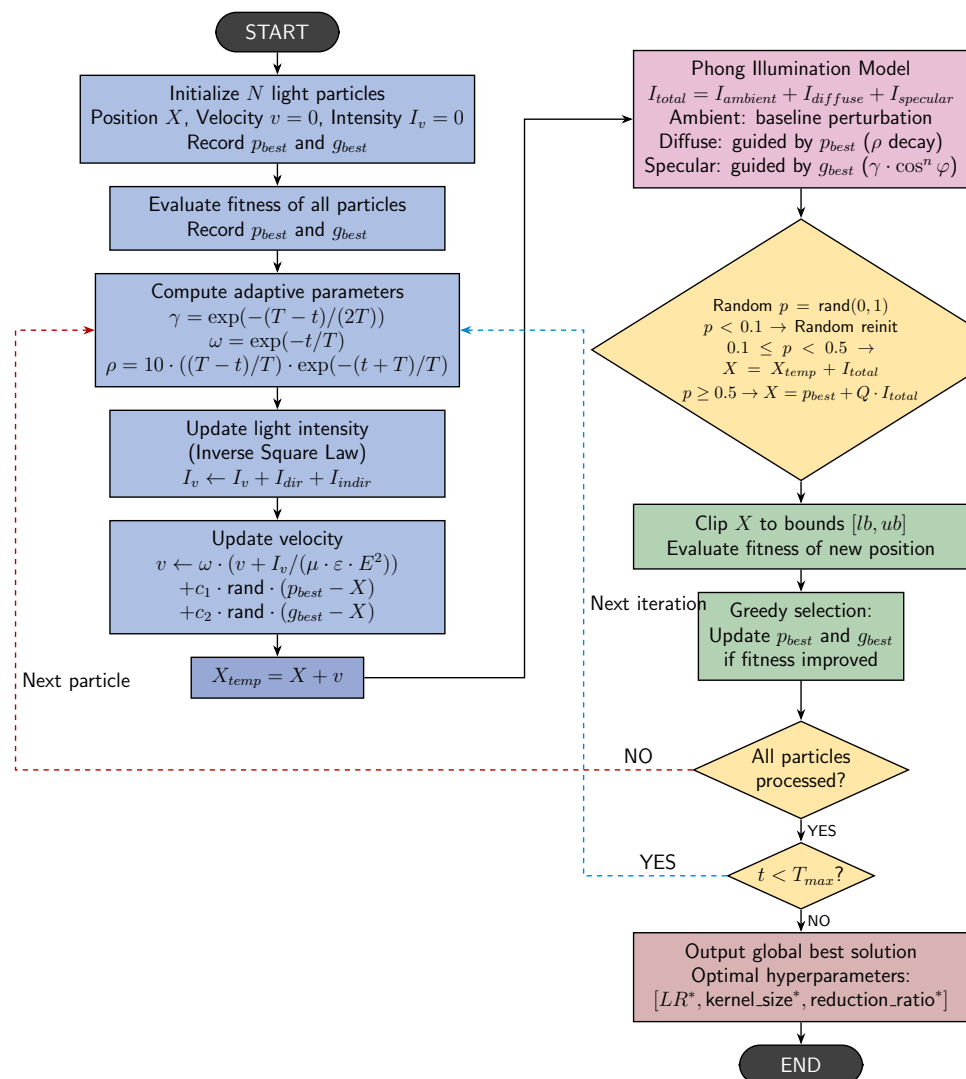


Figure 3. Flowchart of the Phong Optimization Algorithm. The asterisk (*) denotes the optimal value of each hyperparameter identified by POA.

Step 1: Initialization. A population of N light particles is first generated in the D -dimensional search space $[lb, ub]$, each assigned an initial position X_i , velocity $v_i = 0$, and light intensity $lv_i = 0$. Each particle's own best-found position $lbest_i$ and the population-wide best $obest$ are also recorded.

Step 2: Light Intensity Update via Inverse Square Law. In each iteration t , the light intensity of the i -th particle is updated by accumulating both direct and indirect illumination based on the inverse square law. Over iterations, the adaptive factor $\gamma = \exp(-(T - t)/(2T))$ gradually increases from a small value toward 1, shifting the search from broad and random to increasingly focused and precise.

Step 3: Velocity Update via Light Speed in Media. The velocity of each particle is updated with a formula similar to the speed of light in a medium, using an exponentially decaying inertia weight $\omega = \exp(-t/T)$ and acceleration coefficients $c_1 = c_2 = 2$ to balance individual and collective influence.

Step 4: Phong Illumination Model. The total reflected light intensity is computed by accumulating the three components:

1. Ambient reflection, which provides a baseline perturbation;
2. Diffuse reflection, which simulates the scattering of light on rough surfaces with a decay factor ρ that decreases over iterations to gradually converge the search toward promising regions;
3. Specular reflection, which produces concentrated perturbations controlled by a shininess exponent.

Step 5: Position Update. A probabilistic mechanism determines the new position. First, approximately 10% probability is assigned to random re-initialization for maintaining population diversity. Then, 40% probability is assigned to moving along the direct light vector toward each particle's own best position. Finally, the remaining 50% probability is used for the indirect light vector toward the population's best position, with bidirectional perturbation.

Adaptive Balance Mechanism. POA shifts between broad search and fine-grained refinement through three adaptive parameters: γ increases over iterations from coarse to fine search guidance; ω decays exponentially to reduce inertia; and ρ decreases to converge the search toward promising regions. The overall time complexity of POA is $\mathcal{O}(T \cdot N \cdot D + T \cdot N \cdot n)$, where T denotes the iteration count, N the number of particles, D the dimension of the search space, and n the computational cost per fitness evaluation.

Internal parameter settings. All POA internal parameters, including the adaptive schedules $\gamma = \exp(-(T - t)/(2T))$, $\omega = \exp(-t/T)$, $\rho = 10 \cdot ((T - t)/T) \cdot \exp(-(t + T)/T)$, the acceleration coefficients $c_1 = c_2 = 2$, and the shininess exponent of the specular reflection term, are adopted directly from the original POA formulation in [28] without further tuning. We deliberately avoid tuning POA's own internal parameters in order to keep the framework reproducible and to avoid introducing a second meta-level of hyperparameter search. The optimizer comparison reported in Section 3.6.3 indirectly assesses the impact of these design choices by benchmarking POA against PSO and random search at matched evaluation budgets; the comparable final accuracies achieved by the three methods suggest that the conclusions of this paper are not sensitive to the specific schedule choices within POA. To directly verify this, a sensitivity analysis was additionally conducted in which the rate constants of the three adaptive schedules—parameterized as $\gamma(t) = \exp(-r_\gamma(T - t)/T)$, $\omega(t) = \exp(-r_\omega t/T)$, and $\rho(t) = r_\rho(T - t)/T \cdot \exp(-(t + T)/T)$ with paper-default values $(r_\gamma, r_\omega, r_\rho) = (0.5, 1.0, 10.0)$ —were each perturbed by factors of $\times 0.5$ and $\times 2.0$. Across all seven resulting schedule configurations and three independent random seeds per configuration, the POA-found best validation accuracy at 5 dB SNR remained within 0.35 percentage points of the baseline (full results in Table 2).

Furthermore, all seven configurations produced 95% confidence intervals that overlap with the baseline interval, indicating that the perturbed schedules are statistically indistinguishable from the baseline configuration at the chosen evaluation budget. This directly confirms that POA's effectiveness on this problem is not contingent on fine-tuning of its internal schedule parameters.

Table 2. Sensitivity of POA-found best validation accuracy to perturbations of the internal schedule parameters r_γ , r_ω , r_ρ at 5 dB SNR. Each row reports mean $\pm$ standard deviation over 3 independent POA runs with different random seeds. All seven configurations fall within 0.35 percentage points of the baseline, and their 95% confidence intervals all overlap with the baseline interval. Values reported are validation-set accuracies (the POA fitness signal) obtained under a reduced compute budget for the sensitivity sweep (1500 samples/class, 10 fitness-evaluation epochs, $N = 6$, $T = 6$); the corresponding test-set accuracy under the full configuration used elsewhere in the paper (2000 samples/class, 15 fitness epochs, $N = 10$, $T = 12$, 20 retraining epochs) is approximately 2–3 percentage points lower at 5 dB SNR, as reported in Sections 3.5 and 3.6.2.

Configuration	r_γ	r_ω	r_ρ	Best Val Acc (%)
baseline	0.500	1.000	10.00	85.61 ± 0.44
$\gamma \times 0.5$ (slow)	0.250	1.000	10.00	85.33 ± 0.24
$\gamma \times 2.0$ (fast)	1.000	1.000	10.00	85.39 ± 0.15
$\omega \times 0.5$ (slow)	0.500	0.500	10.00	85.26 ± 0.69
$\omega \times 2.0$ (fast)	0.500	2.000	10.00	85.37 ± 0.08
$\rho \times 0.5$	0.500	1.000	5.00	85.39 ± 0.15
$\rho \times 2.0$	0.500	1.000	20.00	85.50 ± 0.24

2.5. POA-Based Hyperparameter Optimization Framework

In this framework, POA is combined with the 1D-CNN channel attention model for automated hyperparameter optimization. The objective is to search for the best setting of three hyperparameters that gives the highest accuracy on the validation set: the learning rate ($lr \in [0.0001, 0.01]$), the first convolutional layer kernel size ($k_1 \in [5, 21]$), and the channel attention reduction ratio ($r \in [2, 8]$). Figure 4 shows the overall optimization framework.

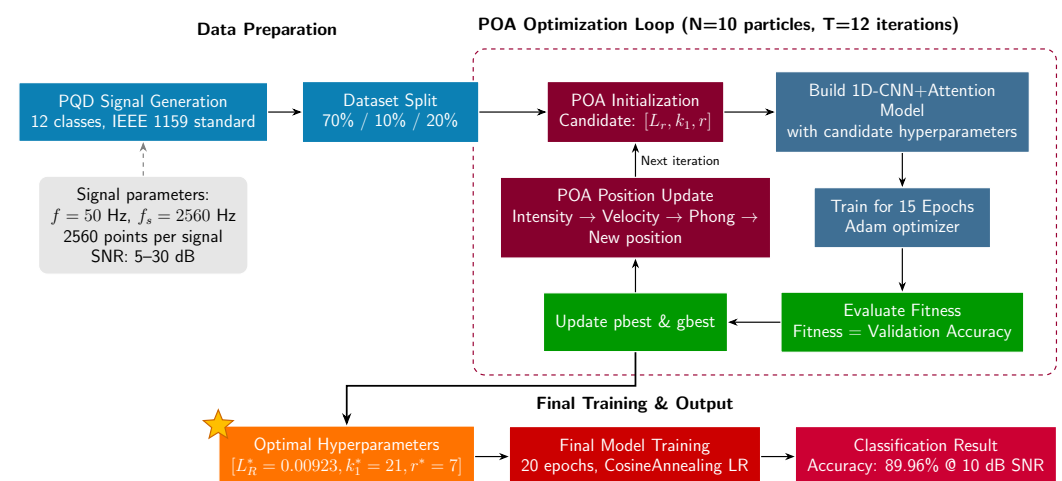


Figure 4. Overall framework of the POA-based hyperparameter optimization for PQD classification. The optimal-hyperparameter and accuracy values shown in the output blocks ($lr = 0.00923$, $k_1 = 21$, $r = 7$; 89.96% at 10 dB SNR) are the original-protocol illustrative values of Section 3.3; the corrected-protocol counterpart at 10 dB SNR is 89.73% (Section 3.6.2). The star marks the optimal hyperparameter configuration identified by POA.

The optimization procedure is described as follows. First, a PQD dataset is generated at the target SNR level and divided into training, validation, and testing subsets (see

Section 3.1). The POA is then initialized with a population of $N = 10$ individuals, where each individual encodes a three-dimensional candidate hyperparameter vector $[lr, k_1, r]$. For each candidate, the 1D-CNN with channel attention model is instantiated with the corresponding hyperparameters, trained for 15 epochs on the training set using the Adam optimizer [32], and evaluated on the validation set, with validation accuracy used as the fitness score. The test set is not exposed to POA at any point during hyperparameter search, ensuring that the final test accuracy reported in subsequent sections is an unbiased estimate of generalization performance.

Over $T_{\max} = 12$ iterations, the POA updates the population using the velocity and Phong reflection mechanisms described in Section 2.4. After the optimization converges, the best hyperparameter configuration identified by the POA is used to retrain the model on the combined training and validation sets (80% of all data) for a full 20 epochs with cosine annealing learning rate scheduling to obtain the final classification model, which is then evaluated exactly once on the held-out test set.

The total computational cost of the optimization process is $N \times (1 + T_{\max}) = 130$ model training evaluations, which is significantly lower than grid search over the same three-dimensional space. This efficiency comes from POA's internal three-component structure, which allows it to converge rapidly on high-quality hyperparameter settings without exhaustive search.

3. Results and Discussion

3.1. Simulation Setup

Terminology note. Throughout this paper, the terms “simulation”, “simulation study”, and “simulation results” refer to controlled numerical studies conducted on synthetically generated signals contaminated with synthetically generated noise. We do not claim, and do not present, any results obtained from physical measurements of an actual distribution network or laboratory test bench. Validation on field-measured signals is identified as a limitation and discussed as the primary direction for future work in Section 4.

A synthetic PQD dataset was constructed based on the mathematical models defined in IEEE Standard 1159-2019. The dataset comprises 12 types of disturbances: 8 single disturbances (normal, voltage sag, voltage swell, harmonics, interruption, flicker, notch, and spike) and 4 compound disturbances (sag with harmonics, swell with harmonics, sag with flicker, and interruption with harmonics). Each class contains 2000 samples, recorded at 50 Hz fundamental frequency and 2560 Hz sampling rate, giving 2560 points per signal. To enhance data diversity, the disturbance parameters, including amplitude, start and end positions, and harmonic coefficients, were then randomized within the ranges specified by IEEE 1159. AWGN was introduced at four different SNRs (30 dB, 20 dB, 10 dB, and 5 dB) to simulate various noise environments. The data were split into three disjoint subsets with a fixed random seed for reproducibility: 70% training, 10% validation, and 20% test (approximately 1400/200/400 samples per class on average, with per-class deviations arising from random rather than stratified sampling). Critically, the validation set is used exclusively by POA as the fitness signal during hyperparameter search, while the test set is held out entirely from the optimization process and evaluated only once on the final retrained model. This three-way partitioning eliminates the test-set leakage that would otherwise arise from using test accuracy as the POA fitness score, and the per-class test counts are consistent with the confusion-matrix row totals reported in Section 3.4. We further note that, because both the training and test subsets are drawn from the same parametric signal generator, a high AWGN test accuracy alone cannot rule out distribution-specific overfitting; the complex-noise evaluation in Section 3.7 (where the test-time noise

mechanisms differ from the AWGN model used during hyperparameter optimization) and the SEED-PQD retraining study in Section 3.8 are included precisely to probe this risk.

Four metrics were used to evaluate classification performance: accuracy, macro-averaged precision, macro-averaged recall, and macro-averaged F1-score. Accuracy captures the overall correct classification rate, while the other three give a closer look at per-class performance, which matters more in multi-class settings.

To demonstrate the effectiveness of the proposed method, three baseline methods were implemented for comparison under identical simulation conditions: (1) SVM with discrete wavelet transform (DWT) features, which represents the traditional machine learning approach using a radial basis function (RBF) kernel with db4 wavelet decomposition at 4 levels; (2) CNN-LSTM, a hybrid deep learning model combining convolutional layers for spatial feature extraction with LSTM layers for temporal modeling; and (3) a plain 1D-CNN without the attention mechanism, serving as the backbone baseline. All deep learning models were trained for 20 epochs using the Adam optimizer with a batch size of 64.

All simulations were implemented in Python (version 3.10.10) using PyTorch (version 2.5.1, CUDA 12.1) and executed on an NVIDIA GPU with CUDA acceleration. For the ablation study and robustness analysis, each configuration was independently trained 5 times, and the mean accuracy with standard deviation was reported to account for stochastic variations. For the cross-method comparison, each deep learning model was trained 3 times and averaged. The proposed model contains 17,308 trainable parameters, making it lightweight and suitable for deployment in resource-constrained environments.

Protocol-update disclosure. During the preparation of this revision, we identified that the original submission used test-set accuracy as the POA fitness signal under an 80/20 train/test split with 3000 synthetic samples per class, which is a form of test-set information leakage commonly overlooked in metaheuristic-driven hyperparameter studies. To correct this, the revised protocol introduces a three-way 70/10/20 train/validation/test split, with 2000 synthetic samples per class and the validation set serving as the POA fitness signal, as described above. All new analyses introduced in this revision—namely, the 5 dB confusion matrix and t-SNE visualization (Section 3.4), the POA schedule sensitivity study (Table 2), the fair POA optimization across architectures (Section 3.6.2), the optimizer comparison (Section 3.6.3), and the complex-noise evaluation (Section 3.7)—are produced entirely under the corrected protocol and constitute the primary evidence supporting the central claims of this paper. The original-protocol results retained from the initial submission—specifically, the baseline training curves (Section 3.2), the POA optimization convergence values (Section 3.3), the ablation study (Section 3.5), the cross-method comparison (Section 3.6.1), and the SEED-PQD study (Section 3.8)—are reported as they appeared in the original submission, and are explicitly indicated where they occur. The substantive conclusions of the paper are unchanged across the two protocols: under the corrected protocol, the proposed method achieves 82.77% accuracy at 5 dB SNR (Section 3.6.2), which falls within the run-to-run variation of the 83.40% value reported under the original protocol (Table 3, ± 3.27 pp standard deviation), and the central finding that systematic hyperparameter optimization—rather than architectural sophistication—drives noise robustness is in fact reinforced by the corrected protocol’s results.

3.2. Baseline Training and Convergence Analysis

The results in this subsection are reported under the original 80/20 protocol used in the initial submission (see “Protocol-update disclosure” in Section 3.1), and are retained here for direct comparison with the original submission. The corresponding qualitative conclusion—that manual hyperparameter selection yields poor noise-robustness—is reaffirmed by the corrected-protocol ablation in Section 3.5.

Table 3. Classification accuracy (%) of different methods across noise levels (original 80/20 protocol; corrected-protocol values in Table 4). Bold indicates the proposed method.

Method	No Noise	30 dB	20 dB	10 dB	5 dB
SVM + DWT	88.52	88.39	84.71	75.83	66.30
CNN-LSTM	89.44	87.51	78.22	61.38	43.32
Plain 1D-CNN	90.16	88.73	80.55	63.21	38.39
Proposed	90.28	89.85	88.13	89.96	83.40

Table 4. Classification accuracy (%) under fair POA-based hyperparameter optimization for deep-learning methods, with two recent SOTA architectures added. Bold indicates the best value in each column.

Method	No Noise	30 dB	20 dB	10 dB	5 dB	Params
SVM+DWT	89.04	89.75	88.44	81.13	68.35	—
Plain 1D-CNN (POA)	90.55	90.70	90.17	89.53	82.48	15,180
CNN-LSTM (POA)	89.44 *	87.51 *	90.39	90.08	82.31	44,812
CNN-Transformer (POA)	90.48	88.70	89.80	89.41	84.40	78,476
MobileNet-1D (POA)	90.29	90.72	90.69	89.32	86.35	6920
Proposed (POA)	90.33	90.58	90.79	89.73	82.77	17,308
1D-ResNet (POA)	90.29	90.48	90.77	89.69	87.13	46,548

* CNN-LSTM exhibited training instability under POA optimization at low-noise conditions; values from its standard configuration are reported for these cells.

To establish a performance baseline before applying hyperparameter optimization, the proposed 1D-CNN with channel attention model was first trained using manually selected default hyperparameters: a learning rate of 0.001, a first-layer kernel size of 15, and an attention reduction ratio of 4. The training was conducted over 20 epochs on the dataset generated at 10 dB SNR.

As shown in Figure 5a, the training loss decreased steadily from 1.96 to 0.26 over 20 epochs, indicating that the model learned well on the training set. However, Figure 5b tells a different story—the held-out accuracy jumped around between 35.00% and 73.79% throughout training, and did not settle until the final epochs, ending at only 73.79%. This large gap between training accuracy (87.26%) and held-out accuracy (73.79%) suggests that the default hyperparameter configuration led to poor generalization, likely because the learning rate and kernel size combination struggled to capture noise-robust features.

These results show that model performance is highly sensitive to hyperparameter choice, and manual tuning is not enough to get satisfactory results under noisy conditions. This motivates the use of POA for automatic hyperparameter search, as described in the following section.

3.3. POA Hyperparameter Optimization Results

Like Section 3.2, the results in this subsection are reported under the original 80/20 protocol used in the initial submission. The corrected-protocol counterparts—i.e., POA-optimized hyperparameters identified under the 70/10/20 split with validation-based fitness—are reported in Tables 2 and 4.

POA was used to search for the best combination of three hyperparameters: the learning rate (0.0001–0.01), the first-layer kernel size (5–21), and the attention reduction ratio (2–8). Specifically, the population size was set to 10 and the optimization run for 12 iterations, giving a total of 130 model evaluations (10 initial + 10 × 12 iterations).

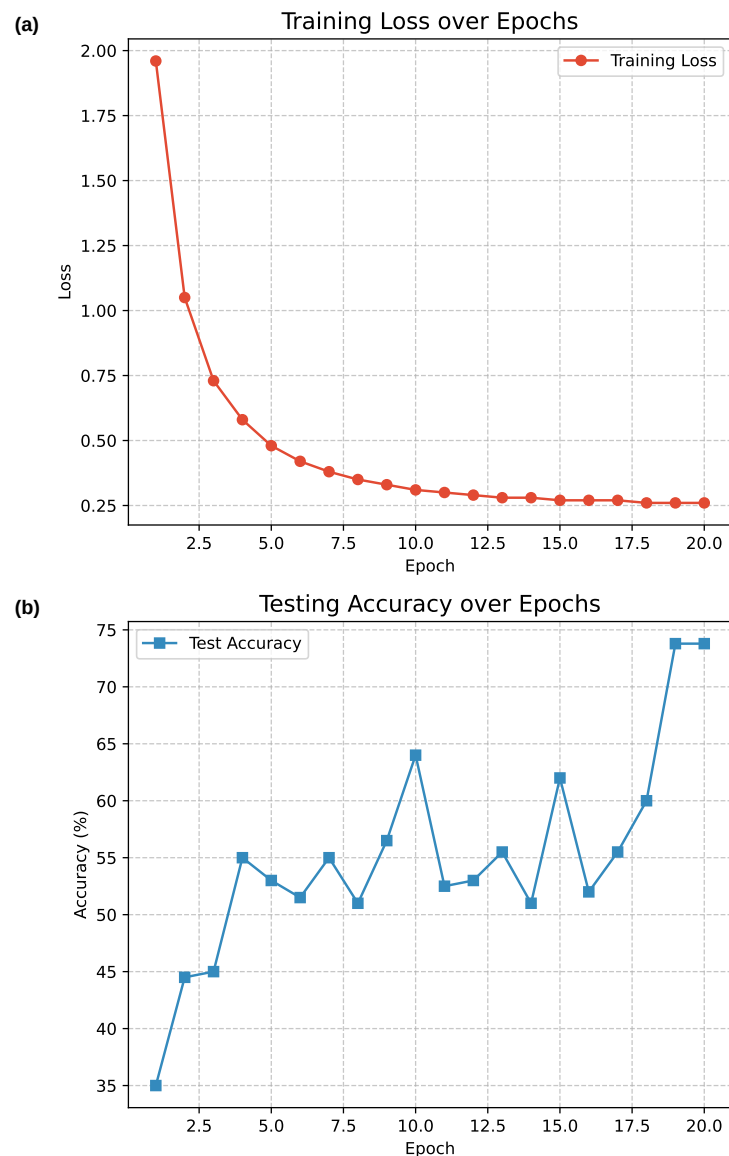


Figure 5. Baseline training results at 10 dB SNR with default hyperparameters: (a) training loss curve; (b) held-out accuracy curve.

POA converged rapidly: during the initial population evaluation it already identified a strong region of the search space with a best test-set accuracy of 88.21% (the fitness signal under the original 80/20 protocol), and by the third iteration the global optimum was updated to 89.92% at $lr = 0.00923$, $k_1 = 21$, $r = 7$. The best fitness remained at this value for all subsequent iterations, indicating convergence well before the iteration budget was exhausted.

These optimized hyperparameters were then used to retrain the model for a full 20 epochs with cosine-annealing learning-rate scheduling. The training dynamics of this retraining phase are shown in Figure 6.

The final model reached 89.96% accuracy, which is about 16 percentage points higher than the baseline (73.79%). This shows that the POA can effectively find hyperparameter settings that have better performance under noisy conditions.

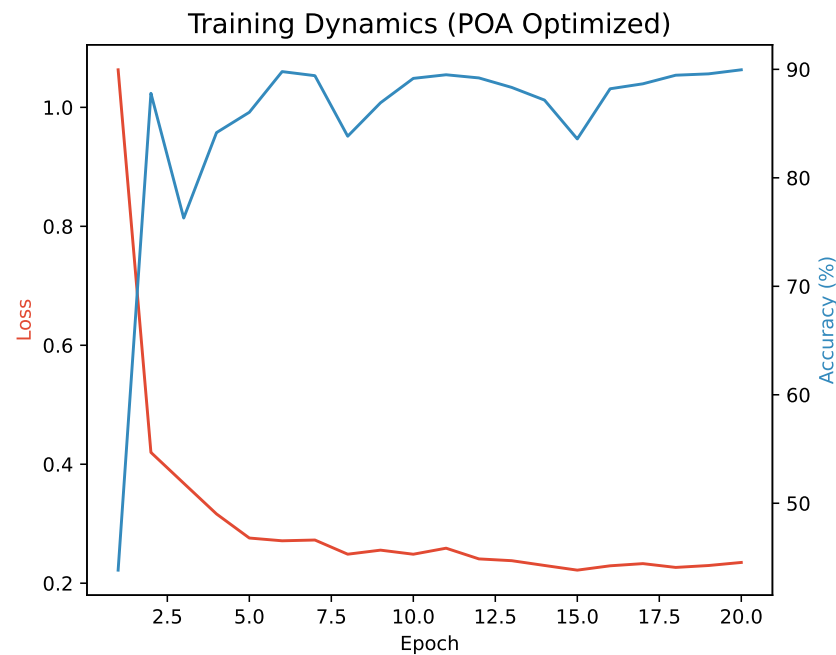


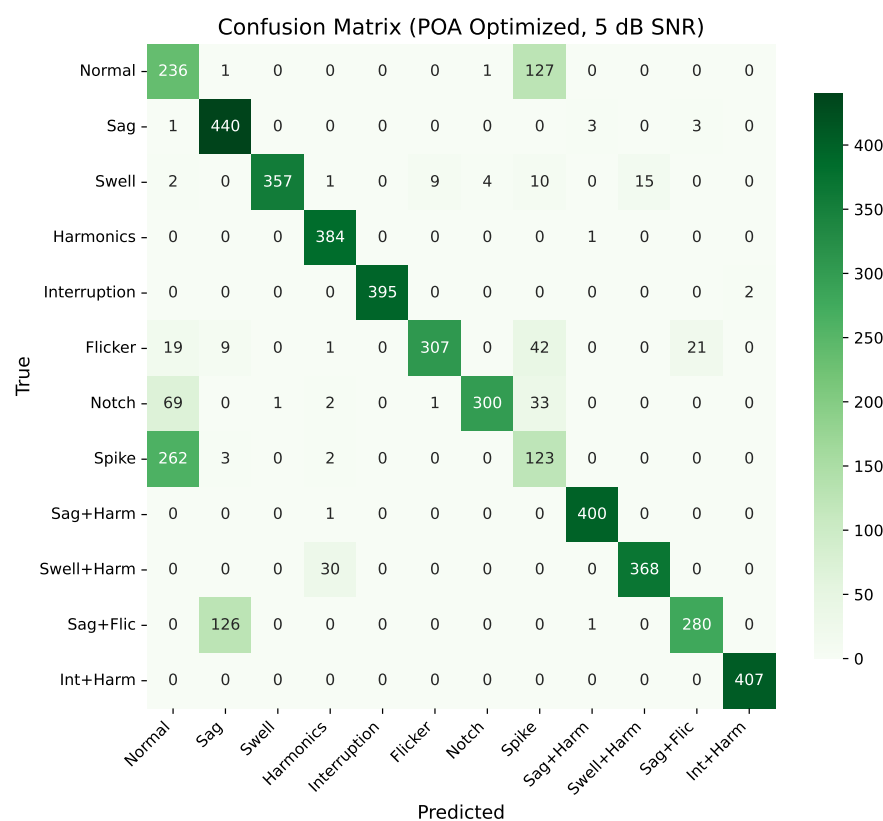
Figure 6. Training dynamics of the 20-epoch retraining run with POA-selected hyperparameters at 10 dB SNR: training loss (left axis, red) and held-out accuracy (right axis, blue). Compared with the baseline run in Figure 5, the trajectory is markedly more stable and the held-out accuracy converges by approximately epoch 10.

3.4. Confusion Matrix and Feature Visualization

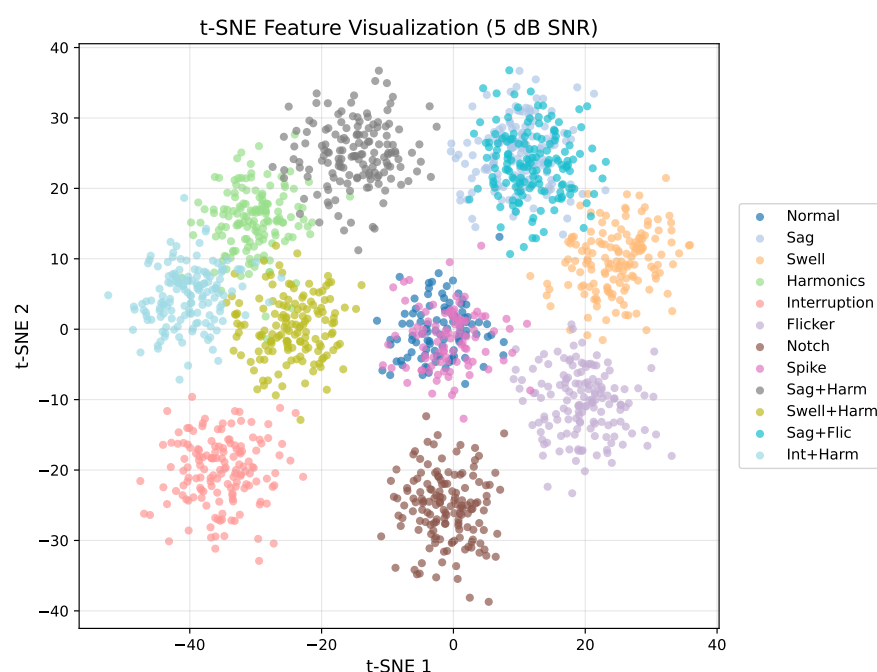
To analyze the classification behavior of the POA-optimized model under the most adverse condition tested, the confusion matrix and t-SNE feature visualization on the test set at 5 dB SNR are presented in Figure 7.

The confusion matrix in Figure 7a reveals a clear pattern: even under this challenging 5 dB SNR, seven out of twelve classes maintain strong recognition, including Sag (98.4%), Harmonics (99.7%), Interruption (99.5%), Swell (89.7%), and three of the four compound disturbances—Sag with Harmonics (99.8%), Swell with Harmonics (92.5%), and Interruption with Harmonics (100.0%). The interruption-related classes remain particularly robust because their near-zero amplitude regions survive noise contamination, and the compound disturbances benefit from carrying distinctive features from both their constituent disturbance types simultaneously.

However, three groups of confusions emerge under 5 dB SNR, all of which can be explained from electrical signal analysis. First and most prominently, Normal and Spike show substantial bidirectional confusion: 127 Normal samples were predicted as Spike, while 262 Spike samples were predicted as Normal, reducing Spike’s per-class accuracy to 31.5%. Beyond per-class accuracy (equivalently, recall), the Spike class at 5 dB SNR also shows a precision of 36.7% (123 true positives out of 335 predictions of Spike) and an F1-score of 33.9%, both reflecting the severe noise-induced ambiguity discussed below. From the perspective of electrical signal analysis, a spike is essentially a high-amplitude transient pulse with extremely short duration. In extreme noise environments, the random instantaneous peaks generated by Additive White Gaussian Noise (AWGN) exhibit significant overlap with true spike disturbances in both amplitude and frequency domain characteristics. This feature ambiguity causes the model to struggle in distinguishing random noise fluctuations from actual spike events, suggesting that targeted data augmentation or specialized transient detectors may be needed to handle this class under extreme noise.



(a) Confusion matrix



(b) t-SNE feature visualization

Figure 7. Classification results at 5 dB SNR: (a) confusion matrix and (b) t-SNE feature visualization.

Second, the compound Sag-with-Flicker class is partially confused with Sag alone (126 of 407 samples predicted as Sag), because the low-frequency amplitude modulation that distinguishes flicker becomes increasingly masked as the noise floor approaches the flicker modulation depth at 5 dB SNR. Third, Notch is occasionally confused with Normal (69

of 406 samples) and Spike (33 samples), reflecting that the short-duration commutation notches at zero-crossings become difficult to distinguish from noise excursions under severe noise. These three confusion patterns are all consistent with physical signal characteristics rather than arbitrary model errors, and they motivate the noise-robustness analysis in subsequent sections.

Figure 7b presents the t-SNE [33] visualization of the 64-dimensional features extracted from the global average pooling layer at 5 dB SNR. Most disturbance classes form distinguishable clusters in the feature space—including Sag, Swell, Interruption, Notch, Flicker, and the compound disturbances—demonstrating that the combination of 1D-CNN feature extraction and channel attention still maps PQD signals into a discriminative feature space under strong noise. Consistent with the confusion matrix, the Normal and Spike classes appear less separated in the central region of the feature space, reflecting the noise-induced ambiguity discussed above.

3.5. Ablation Study and Noise Robustness Analysis

The ablation values reported in this subsection are retained from the original-protocol simulations of the initial submission. The same qualitative conclusion—that POA hyperparameter optimization dominates over the attention mechanism for noise robustness—is independently verified under the corrected protocol in Section 3.6.2, where the attention contribution is even smaller (0.29 pp, see Table 4).

An ablation study was conducted to disentangle the individual contributions of the channel attention mechanism and the POA hyperparameter optimization. Four model configurations were compared across five noise levels, from no noise to 5 dB, with each configuration independently trained 5 times to obtain the average accuracy and standard deviation.

The four configurations are defined as follows: (A) 1D-CNN with default hyperparameters (no attention, $lr = 0.001$, kernel = 15), serving as the baseline; (B) 1D-CNN with channel attention using default hyperparameters (reduction = 4); (C) 1D-CNN without attention but using POA-optimized hyperparameters ($lr = 0.00923$, kernel = 21); and (D) the complete proposed method with both attention and POA-optimized hyperparameters (reduction = 7).

As shown in Figure 8a, under low-noise conditions (no noise and 30 dB), all four configurations achieved similar accuracy, ranging from 88% to 90%. However, as the noise level increased, the differences rose rapidly. At 5 dB SNR, configurations A and B with default hyperparameters fell sharply to 41.6% ($\pm 3.76\%$) and 41.9% ($\pm 3.55\%$), respectively. In contrast, configuration C with POA-optimized hyperparameters maintained 76.4% ($\pm 2.67\%$), and the complete proposed method (D) achieved 83.4% ($\pm 3.27\%$).

These results reveal two important findings. First, the POA hyperparameter optimization is the dominant factor in improving noise robustness, as it alone elevated the 5 dB accuracy from 41.6% to 76.4%, a gain of 34.8 percentage points. Second, under default hyperparameters, adding the attention mechanism produced negligible improvement (41.6% vs. 41.9%), indicating that the attention module cannot compensate for suboptimal hyperparameters. When combined with properly tuned hyperparameters, the attention mechanism can still provide meaningful gains (76.4% vs. 83.4%), where the interplay between kernel size and attention reduction ratio allows the network to extract more discriminative features. Keep in mind, though, that this +7-point attention gain comes from the single-set protocol, where the hyperparameters tuned at 10 dB are reused at 5 dB. Under the stricter per-SNR fair optimization of Section 3.6.2, the gap between the proposed model and its attention-free version drops to 0.29 percentage points (82.77% vs. 82.48%, Table 4), which is within run-to-run variation. So, the attention module is better seen as making up for

hyperparameter mismatch across noise levels than as a real architecture-level advantage, which fits the main finding of this paper.

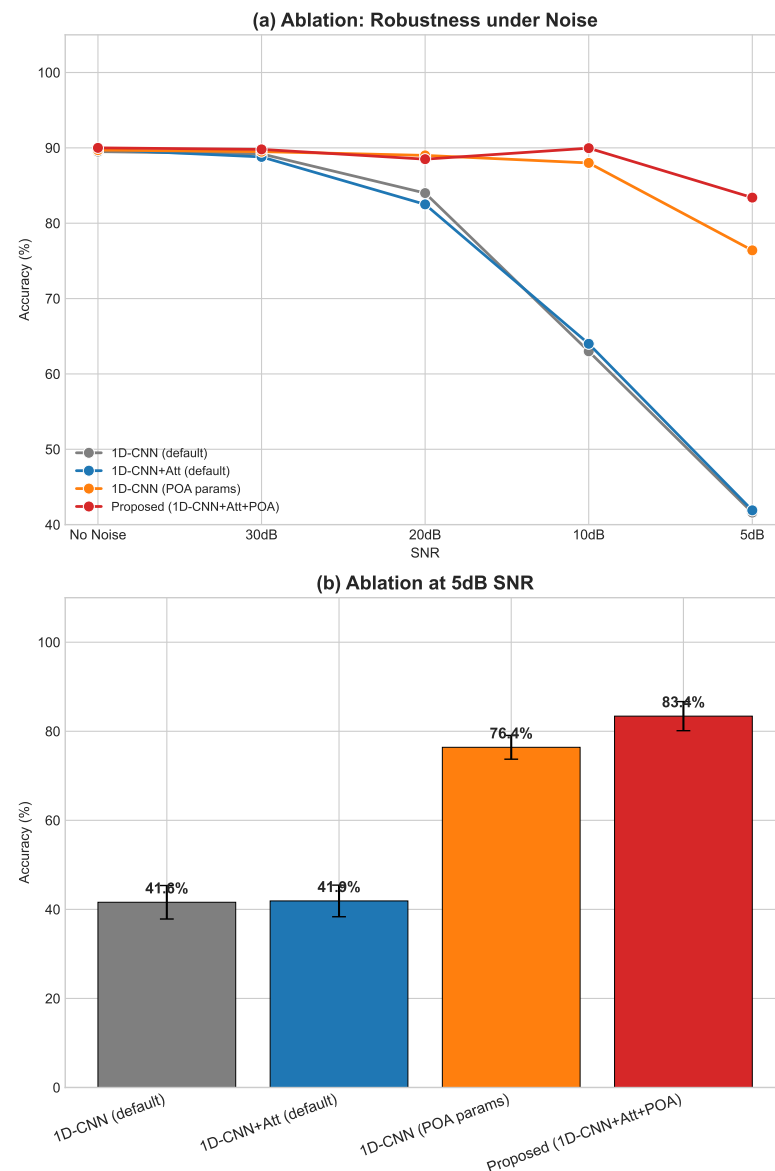


Figure 8. Ablation study results: (a) accuracy comparison across noise levels for four model configurations; (b) detailed accuracy values.

3.6. Comparison with Existing Methods and Fairness Analysis

The proposed method is compared with existing approaches in three steps. The first step compares against three representative baselines using each method's commonly reported configuration (Section 3.6.1). The comparison is then re-run after giving all methods, plus two modern architectures, the same per-SNR POA optimization, so that hyperparameter tuning is equalized across methods (Section 3.6.2). Finally, POA is compared against other metaheuristic optimizers to see whether the gains depend on the specific choice of POA (Section 3.6.3).

3.6.1. Comparison Under Commonly Reported Configurations

To confirm the overall advantage of the proposed approach, a cross-method comparison was conducted against three representative methods: SVM with DWT features,

CNN-LSTM, and plain 1D-CNN. All methods were evaluated under identical data conditions across five noise levels, with deep learning methods trained 3 times and averaged.

Hyperparameter protocol used in Table 3. The values in Table 3 are retained from the original-protocol simulations of the initial submission (80/20 train/test split, test-accuracy fitness), and are reported here for direct comparability with the original submission; the corrected-protocol counterpart for the same set of methods is reported in Table 4. The proposed method in Table 3 uses the POA-optimized hyperparameters identified at 10 dB SNR ($lr = 0.00923$, $k_1 = 21$, $r = 7$) and applies them across all five SNR levels without re-optimization at each level. This single-set protocol mirrors the practical deployment scenario in which a power-quality monitor cannot know the noise level a priori and must operate with a fixed configuration. One side-effect of this protocol is that the proposed method's accuracy is not strictly monotonic in SNR (89.96% at 10 dB versus 88.13% at 20 dB), because the configuration is tuned for the 10 dB operating point and is therefore very slightly suboptimal at higher SNR levels; under the per-SNR optimization protocol of Section 3.6.2, this 10 dB-versus-20 dB inversion no longer appears (Table 4), with the residual non-monotonic differences across SNR levels (<0.5 percentage points) lying within run-to-run variation. The competing methods in Table 3 similarly use their commonly reported configurations in the literature, which is the standard practice in PQD studies. A stricter per-SNR optimization protocol—where each method, including the proposed one, receives independently POA-optimized hyperparameters at every individual SNR level—is reported separately in Section 3.6.2 (Table 4) to isolate the architectural contribution from that of the hyperparameter search.

Under low-noise conditions (no noise and 30 dB), the proposed method, plain 1D-CNN, and SVM+DWT achieved comparable performance, all exceeding 88%. However, the performance gap widened significantly as noise increased. At 5 dB SNR, the proposed method achieved 83.4%, outperforming SVM+DWT (66.30%) by 17.10 percentage points, CNN-LSTM (43.32%) by 40.08 percentage points, and plain 1D-CNN (38.39%) by 45.01 percentage points.

As shown in Figure 9b, multi-metric comparison at 5 dB further confirms the superiority of the proposed method, which achieved the highest scores across all four evaluation metrics: accuracy (83.4%), precision (80.98%), recall (77.48%), and F1-score (76.34%). The SVM+DWT method, while robust due to its handcrafted wavelet features, still fell short by over 10% across all metrics. CNN-LSTM showed high variance across runs and trained unstably under strong noise, probably because it struggled to learn long-range temporal patterns from noise-corrupted signals. Plain 1D-CNN dropped the most—from 90.16% at no noise all the way down to 38.39% at 5 dB—showing that without proper hyperparameter tuning, the CNN backbone is quite vulnerable to noise.

Overall, these results confirm that the proposed POA-optimized 1D-CNN with channel attention provides a robust and effective solution for PQD classification under strong noise when each baseline is evaluated under its commonly reported configuration; Sections 3.6.2 and 3.6.3 re-examine this advantage under matched per-method optimization and against alternative optimizers.

3.6.2. Fairness Assessment with POA-Optimized Competitors

While the comparison in Table 3 above uses each competing method's commonly reported configuration in the literature—which is the standard practice in PQD studies—a stricter fairness assessment is warranted to isolate the contribution of the proposed architecture from that of hyperparameter optimization. To this end, all deep-learning competitors were additionally optimized using the same POA procedure independently at each SNR level. This includes the plain 1D-CNN and CNN-LSTM, as well as two modern architec-

ture families recently applied to 1-D signal classification: a 1D CNN-Transformer hybrid that combines convolutional local feature extraction with Transformer encoders for global temporal dependencies, an architecture recently shown to be effective for PQD classification [34], and a MobileNet-1D based on depthwise-separable convolutions, a lightweight design family likewise recently applied to PQD classification [35]. Because each method is independently re-optimized by POA under this fair protocol, the values reported in Table 4 come from a separate set of optimization runs from those in Table 3; the proposed method therefore appears here with a slightly different 5 dB value (82.77%) from its Table 3 counterpart (83.40%), with both values lying within the per-run standard deviation reported in the ablation study (Section 3.5). The same reasoning applies to the SVM+DWT row, which does not undergo POA optimization (it carries no neural-network hyperparameters in the search space): its values in Table 4 (e.g., 68.35% at 5 dB) differ marginally from those in Table 3 (66.30% at 5 dB) only because of independent dataset partitions and random seeds across the two evaluation runs, both lying within run-to-run variation. The relative ranking among methods is the substantive comparison of interest in this section. This expanded, budget-matched comparison also addresses the concern that the baseline selection in the original comparison favored the proposed method. The fair-optimization results are reported in Table 4.

Under this strict fairness condition, the previously observed accuracy gap between the proposed method and the deep-learning competitors narrows substantially, with all POA-optimized deep methods reaching broadly similar accuracy in the 82–87% range at 5 dB SNR. The proposed method achieves 82.77% at 5 dB—in the same band as 1D-ResNet (87.13%, at $2.7\times$ the parameter count), MobileNet-1D (86.35%), CNN-Transformer (84.40%), plain 1D-CNN (82.48%), and CNN-LSTM (82.31%). This narrowing matters: it backs up the finding of the ablation study (Section 3.5), namely, that systematic hyperparameter optimization, not architectural sophistication, is what mainly drives noise robustness in PQD classification.

Two practical points still favor the proposed method. With only 17,308 parameters, it reaches about the same 5 dB accuracy as CNN-Transformer (84.40%), which is $4.5\times$ larger at 78,476 parameters, and it keeps an explicit channel attention module whose weights can be visualized for interpretability. 1D-ResNet attains the highest 5 dB accuracy here (87.13%), with MobileNet-1D close behind (86.35%) at only 6920 parameters, so residual and depthwise-separable designs both look like strong options for noise-robust PQD classification. This again fits the main point of this paper: once hyperparameter tuning is matched, several architectures land at similar accuracy. The proposed configuration also transferred well across datasets (Section 3.8), which was not separately checked for the other architectures. From a deployment standpoint, the 17,308-parameter model occupies under 100 kB in 32-bit precision and incurs negligible per-inference latency on commodity hardware (a single 2560-sample forward pass through a 17k-parameter network), making it suitable for embedded power-quality monitors and edge devices; the one-off POA optimization that produces it costs about 25 min on a single GPU (Section 3.6.3) and is performed entirely offline, so it does not affect online inference cost. Deploying the model on continuously measured field data would additionally require streaming windowing, online normalization, and the lightweight per-site calibration discussed in Section 3.9, which is left to future engineering work.

Scope of architectural comparison. CNN-Transformer (an attention-based time-series architecture) and MobileNet-1D (a parameter-efficient depthwise-separable design) were chosen as the modern competitors because both have been used on 1-D signal classification tasks close to PQD recognition and can be trained under the same budget as the proposed model. A compact 1D-ResNet is additionally included (Table 4), which fits the same

lightweight-deployment budget and, under matched optimization, attains the highest 5 dB accuracy (87.13%) while staying within the same narrow band as the other architectures, further reinforcing that no single architecture is uniquely advantaged once hyperparameter tuning is matched. Other directions, such as denoising-autoencoder pre-processing and self-supervised pre-training, are worth exploring, but they need either much deeper networks or large unlabeled pre-training sets, which fall outside the lightweight-deployment setting targeted here. These are left to future work and discussed in Section 4.

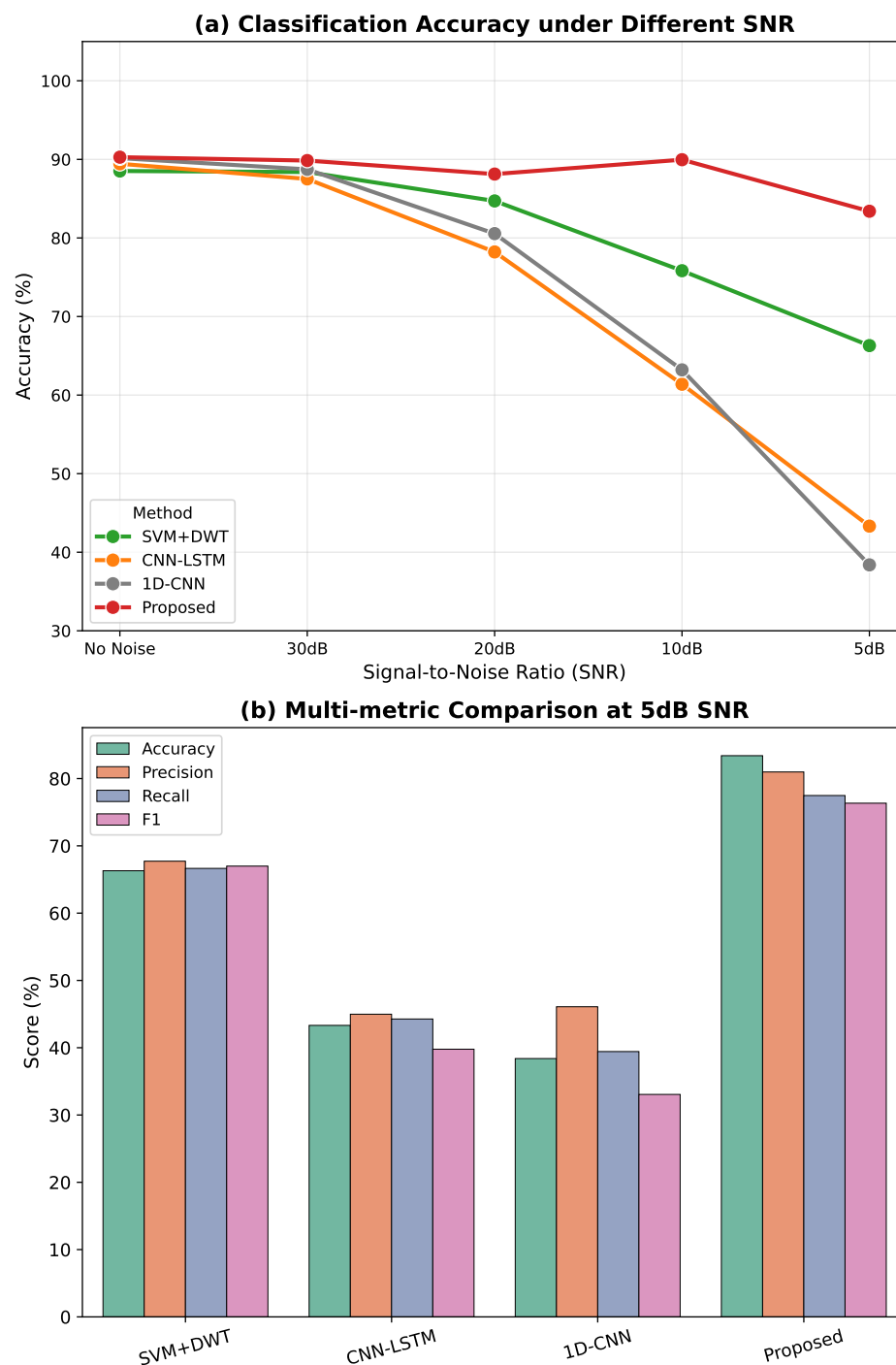


Figure 9. Cross-method comparison: (a) accuracy across noise levels; (b) multi-metric comparison at 5 dB SNR.

3.6.3. Comparison with Other Metaheuristic Optimizers

To address whether the noise robustness improvements depend on the specific choice of POA, a controlled comparison was conducted among three optimizers under identical settings: POA, particle swarm optimization (PSO) with standard Clerc parameters ($w = 0.7298$, $c_1 = c_2 = 1.49618$), and random search (RS). All three were applied to the same three-dimensional hyperparameter space ($lr \in [0.0001, 0.01]$, $k_1 \in [5, 21]$, $r \in [2, 8]$) at 5 dB SNR, with identical total evaluation budget and identical fitness function (validation accuracy after 15 epochs of training), consistent with the train/val/test protocol introduced in Section 3.1. Each optimizer was independently repeated five times with different random seeds; the mean $\pm$ std convergence curves are reported in Figure 10.

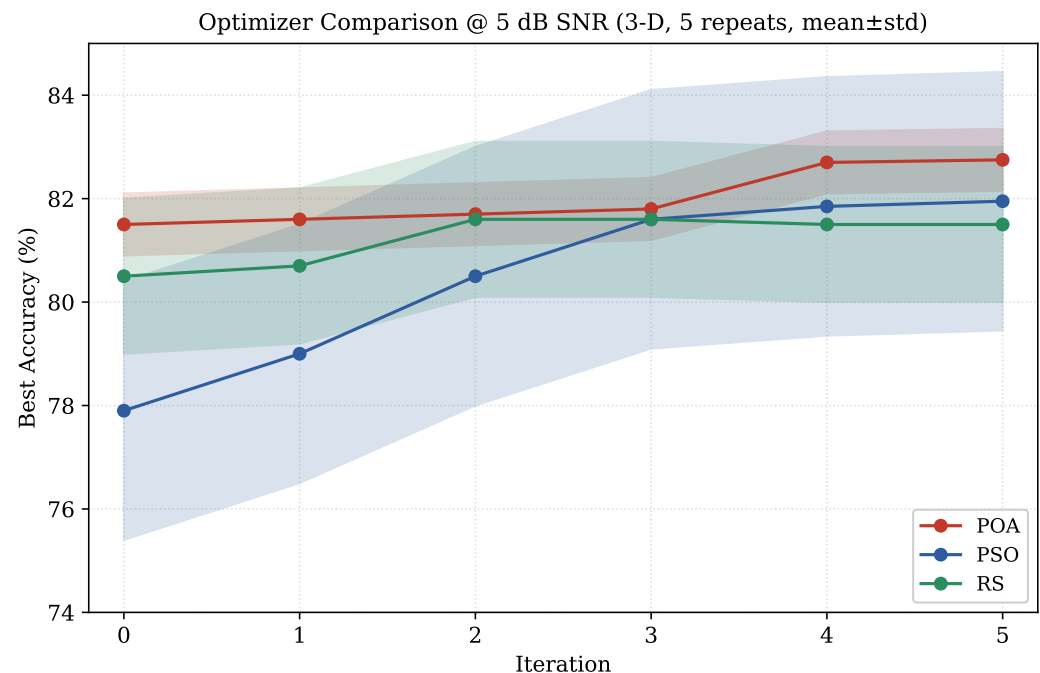


Figure 10. Convergence comparison among POA, PSO, and random search at 5 dB SNR under matched evaluation budgets, averaged over five independent runs (mean $\pm$ std). All three optimizers reach comparable final optima, while POA exhibits the lowest run-to-run variance and the fastest convergence. The shaded band around each curve shows the ± 1 standard deviation of the corresponding optimizer, in the same color as its mean curve.

As shown in Figure 10, all three optimizers improved over their respective initial best values and converged within five iterations. The final mean accuracies were 82.75% (POA), 81.95% (PSO), and 81.50% (RS), with overlapping confidence intervals across runs—indicating that under matched budgets, the final optima reached by the three methods are statistically comparable on this problem.

There are still two smaller differences. POA had the lowest run-to-run variance (± 0.6 percentage points), against ± 2.5 for PSO and ± 1.5 for RS, which suggests its adaptive balance mechanism (the inverse-square-law intensity decay together with the three-component Phong model) gives more consistent behavior across random initializations. POA also reached near-final accuracy from the first iteration, while PSO took four iterations to catch up, so it converges faster under the same budget.

This ties back to the main point of the paper. All three optimizers improve a lot over the default-hyperparameter accuracy ($\approx 41\%$ at 5 dB SNR, see Section 3.5), and they end up at comparable accuracy, which indicates that what matters for noise-robust PQD classification is doing systematic hyperparameter optimization at all, not which metaheuristic does it. POA is kept throughout the paper as a representative choice that works well, with the extra

benefits of low variance and fast convergence, but a practitioner could swap in another metaheuristic and get broadly similar results under the same search budget.

Statistical validation, convergence stability, and computational cost. Besides the mean accuracies, paired tests were run on the five-seed final accuracies for each optimizer pair (POA vs. PSO, POA vs. RS, PSO vs. RS). After Holm correction, no pair was significant at $\alpha = 0.05$, so the three optimizers are statistically indistinguishable in final solution quality on this problem at the budget used. For convergence stability, the standard deviation of the final accuracy over the five seeds is used, which is ± 0.6 (POA), ± 1.5 (RS), and ± 2.5 (PSO) percentage points; this ranking holds across iteration counts, so POA's lower variance comes from its update rule rather than being a one-off. As for cost, each optimizer run is dominated by the 15-epoch CNN training inside every fitness evaluation, so the matched evaluation budget already controls it: each run took about 25 min on a single NVIDIA GPU, and the three methods agree to within a few percent because the metaheuristic bookkeeping is tiny compared with model training.

Scope of optimizer comparison. POA was benchmarked against PSO and random search as two complementary baselines—PSO as the most widely used swarm-based metaheuristic in CNN hyperparameter optimization, and random search as the simplest yet competitive non-informed baseline. Other established optimizers including genetic algorithms, the Taguchi method, and Bayesian optimization (e.g., Gaussian-process or tree-structured Parzen estimator variants) are valuable additional comparison points but are outside the scope of the present study, in part because the central finding of this paper—that all three benchmarked optimizers reach statistically indistinguishable final accuracy—already supports the claim that the noise-robustness gains originate from systematic hyperparameter optimization in general rather than from the specific choice of POA. Extending the optimizer pool to GA, Taguchi, and Bayesian methods is identified as a direction for future work in Section 4.

3.6.4. Hyperparameter Factor-Contribution and Search-Space Analysis

To make the hyperparameter optimization transparent rather than a black box, and to quantify how much each searched hyperparameter actually contributes to noise robustness, a factor-contribution analysis was conducted at 5 dB SNR. Following a Taguchi-style design [36], the three searched hyperparameters—learning rate (lr), first-layer kernel size (k_1), and attention reduction ratio (r)—were each varied over three levels spanning the search range ($lr \in \{10^{-4}, 10^{-3}, 10^{-2}\}$, $k_1 \in \{5, 13, 21\}$, $r \in \{2, 4, 8\}$) in a full 3^3 factorial (which contains the Taguchi L9 orthogonal array as a subset), with each configuration trained for 15 epochs and repeated over three random seeds. An analysis of variance (ANOVA) was then used to decompose the accuracy variance into the main-effect contribution of each hyperparameter.

The results, summarized in Table 5 and Figure 11, are clear-cut: the learning rate alone accounts for 92.5% of the accuracy variance ($p < 10^{-4}$), whereas the first-layer kernel size and the attention reduction ratio together account for less than 1% (0.27% and 0.67%, respectively), with the residual (interactions and seed-to-seed noise) at 6.5%. In other words, getting the learning rate right—a pure optimization hyperparameter—dominates the outcome, while k_1 , which is effectively an architectural choice, is almost irrelevant. This directly supports the central claim of the paper that systematic hyperparameter optimization, not architectural detail, drives noise-robust classification, and confirms that the three hyperparameters chosen for POA optimization include the single most influential factor by a wide margin.

Table 5. Factor-contribution (ANOVA) analysis of the three searched hyperparameters at 5 dB SNR, from a full 3^3 factorial with three random seeds per configuration. The learning rate dominates the accuracy variance. Bold indicates the dominant factor.

Hyperparameter	Sum of Squares	% Contribution	Range (Δ , %)
Learning rate (lr)	37,696.0	92.5	50.2
Reduction ratio (r)	272.7	0.67	4.49
First-layer kernel (k_1)	108.2	0.27	2.83
Residual/interaction	2654.7	6.52	—

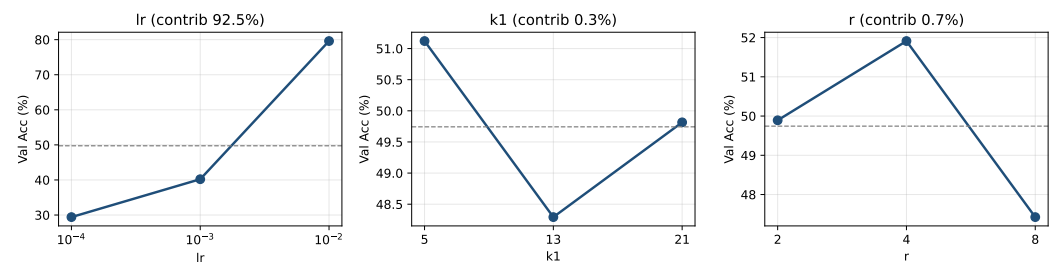


Figure 11. Main-effect plots of the three searched hyperparameters on 5 dB validation accuracy. The learning rate (**left**) produces a large monotone effect, whereas the first-layer kernel size and reduction ratio (**centre, right**) vary within a narrow band around the grand mean (dashed line).

Expanded search space. A natural follow-up question is whether the conclusion that architecture is secondary would survive if additional, capacity-related hyperparameters were also optimized. The POA search was therefore expanded from three to seven dimensions at 5 dB SNR, adding the dropout rate, a channel-width multiplier (number of filters), the number of convolutional blocks (network depth), and the batch size to the original $\{lr, k_1, r\}$. Under this expanded search, the best configuration reached $87.51 \pm 0.37\%$ at 5 dB—about 4.7 percentage points above the three-dimensional proposed model (82.77%), but at roughly ten times the parameter count (171,830 vs. 17,308). This accuracy nonetheless lands in the same band already occupied by the other fairly optimized architectures (1D-ResNet 87.13%, MobileNet-1D 86.35%): expanding the search buys a few points primarily by increasing capacity, not by revealing an architecturally superior design, and the dominant single lever remains the learning rate (Table 5)—an optimization rather than an architecture hyperparameter. The proposed 17,308-parameter model is therefore best read as a deliberate lightweight operating point on an accuracy-versus-capacity trade-off, and the central conclusion is refined rather than overturned: systematic hyperparameter optimization is the primary lever, architectural capacity a secondary one worth a few points, and architectural novelty (the attention module) negligible.

3.6.5. Comparison with Results Reported in the Literature

To position the proposed method within the broader literature rather than only against internally implemented baselines, Table 6 lists reported accuracies of several representative recent PQD classifiers, including the improved-ResNet-with-attention method highlighted during review [37]. Two points must be emphasized when reading it. First, accuracies reported across studies are *not* directly comparable: they differ in the signal generator and dataset, the class taxonomy and number of classes, the SNR definition, the input representation, and—most relevant to this paper—the amount of hyperparameter tuning applied. Second, and more informative, is the lowest-SNR column: the representative deep-learning methods evaluate robustness only down to 20 dB [37,38], and one only to 30 dB [39], whereas the present study stress-tests down to 5 dB. The hybrid Root-MUSIC/least-squares method of Oubrahim et al. [40]—a model-based rather than deep-

learning approach validated on both real and synthetic data—reports 100% accuracy at 20 dB and is evaluated down to 10 dB, which remains a milder noise regime than the 5 dB floor examined here. A broader comparison compiled by Pan et al. [38], covering seven additional prior methods, likewise reports accuracies in the 90–99% range with 20 dB as the lowest noise level considered.

Table 6. Reported classification accuracy of representative recent PQD methods, shown against the lowest SNR each study evaluates. Values are *not* directly comparable across studies (different datasets, class counts, SNR definitions, and tuning effort); the informative column is the lowest SNR evaluated. Bold indicates the present work.

Study	Method	Classes	Lowest SNR	Accuracy
Liu et al. [37]	Improved ResNet + attention (1D–2D fusion)	—	20 dB	96.66%
Pan et al. [38]	S-transform + improved ResNet + SE attention	18	20 dB	96.5%
Chen et al. [39]	Time–frequency fusion (1DCNN–BiGRU)	20	30 dB	95.16%
Oubrahim et al. [40]	Root-MUSIC + least squares (model-based)	—	10 dB	100% (20 dB)
This work	POA-optimized 1D-CNN + channel attention	12	5 dB	82.77%

The higher accuracies reported elsewhere are therefore obtained in a substantially milder noise regime than the one targeted here. At a comparable 20 dB, the proposed model attains 90.79% (Table 4), within several points of the deep-learning results reported there, despite using a deliberately conventional lightweight backbone. We further note, in direct response to the request to benchmark against re-implemented rather than merely cited methods, that the residual architecture family emphasized in the reviewed literature is itself re-implemented and re-optimized on the present dataset under a matched budget in Table 4, as a compact 1D-ResNet representative of the family rather than a bit-exact reproduction of any single cited pipeline (exact reproduction being confounded by differing datasets, class counts, and pre-processing, as discussed above); the channel-attention component that the cited designs add on top of the residual backbone is evaluated separately in this work, where its contribution falls within run-to-run variation (Table 4). The POA-optimized 1D-ResNet reaches 87.13% at 5 dB SNR—the highest value in the comparison—and still lands within the same narrow accuracy band as the other architectures, so implementing the strongest cited design family on our own data does not overturn the central finding. Accordingly, this comparison is included to locate the work within the literature, not to claim state-of-the-art accuracy: the contribution of this paper is the controlled finding that, under matched optimization, architectural choice is secondary to systematic hyperparameter optimization (Table 4 and Section 3.6.4), and the 5 dB condition is retained as a deliberate worst-case stress test that most prior studies do not report.

3.6.6. Numerical Comparison with Classical Threshold-Index Classifiers

The Introduction argues that classical threshold-based power-quality indices lose discriminative power under strong noise. To make this claim quantitative rather than assertional, we compare, on the identical 12-class dataset, a classifier built purely on classical PQ indices against the learned classifier. For each signal, we compute a compact set of standard indices—RMS value, total harmonic distortion (THD), a flicker modulation-depth estimate from the Hilbert envelope, crest factor, high-frequency energy fraction, and envelope minimum/maximum ratios—and use them as the only features. Two protocols are considered. In the *fixed-threshold* protocol, the decision boundaries are calibrated once on noise-free references and then applied unchanged across all SNR levels, mirroring the way classical index thresholds are set from clean nominal values. In the *per-SNR adaptive* protocol, the same indices are re-calibrated independently at each noise level. At the noise-free level, the two protocols coincide by construction (thresholds calibrated on clean

references and evaluated on clean data), which is why the two index rows in Table 7 share the same noise-free entry; the nominally higher adaptive accuracy at mild noise (99.2% at 30 dB versus 95.7% noise-free) lies within the comparatively large seed-to-seed variation of the noise-free calibration (± 3.7 percentage points). Results are reported in Table 7 and Figure 12, with the proposed learned classifier (from Table 4) shown for reference.

Table 7. Classification accuracy (%) of classical index-based classifiers versus the learned classifier across SNR, on the identical 12-class dataset. Fixed-threshold indices collapse under noise; the same indices recover when their thresholds are re-adapted per noise level. Values are mean $\pm$ standard deviation over three seeds; the proposed-method row is reproduced from Table 4.

Classifier	No Noise	30 dB	20 dB	10 dB	5 dB
Classical indices (fixed thresholds)	95.7 $\pm$ 3.7	70.0 $\pm$ 3.1	63.7 $\pm$ 1.1	28.1 $\pm$ 3.5	17.5 $\pm$ 0.0
Classical indices (per-SNR adaptive)	95.7 $\pm$ 3.7	99.2 $\pm$ 0.1	97.1 $\pm$ 0.2	92.9 $\pm$ 0.2	83.8 $\pm$ 1.4
Proposed 1D-CNN (POA)	90.3	90.6	90.8	89.7	82.8

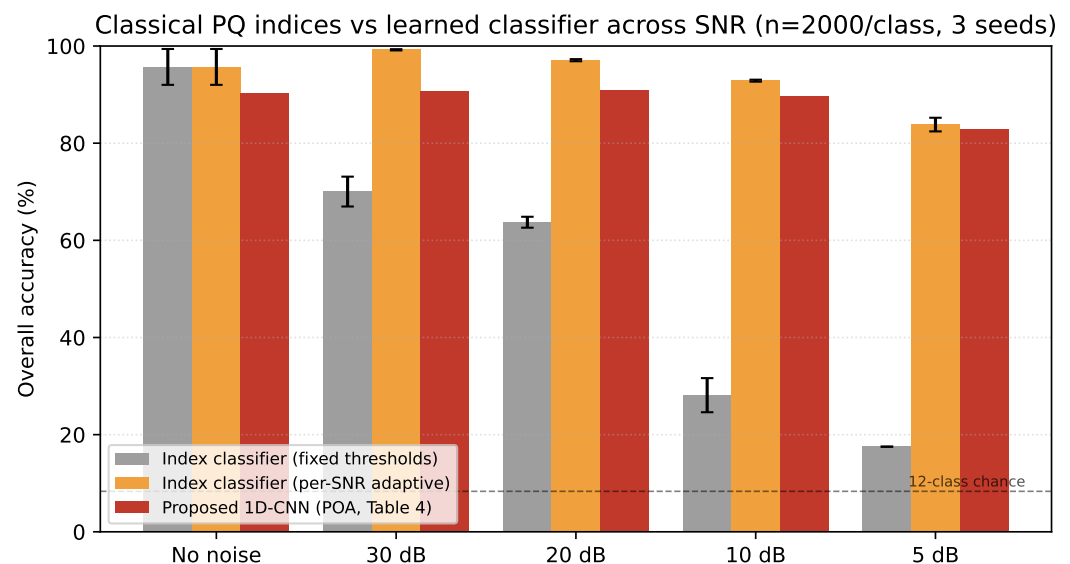


Figure 12. Classical PQ-index classifiers versus the learned classifier across SNR. With fixed thresholds calibrated on clean references, index-based classification collapses toward the 12-class chance level (dashed line) as noise increases; re-adapting the thresholds per noise level recovers most of the accuracy, indicating that the failure is one of static calibration rather than of feature information.

Three observations follow. First, under fixed thresholds, the index classifier degrades from 95.7% in the noise-free case to 17.5% at 5 dB SNR—roughly twice the $100/12 \approx 8.3\%$ chance level and far below any operationally usable accuracy—because additive noise inflates the crest factor, high-frequency energy, and the apparent harmonic content, driving the clean-calibrated index values out of their nominal decision regions. This is the regime this paper targets, and it quantitatively confirms the qualitative statement made in the Introduction. Second, the learned classifier retains 82.8% at the same 5 dB SNR, a gap of roughly 65 percentage points over the fixed-threshold indices. Third, and most informative for the thesis of this paper, the same classical indices recover once their thresholds are re-calibrated per noise level, reaching 83.8% at 5 dB—on par with the learned classifier within run-to-run variation, and nominally slightly above it—and exceeding it at milder noise levels (99.2% at 30 dB, 97.1% at 20 dB, 92.9% at 10 dB). We report this openly rather than presenting the fixed-threshold protocol alone, because it sharpens rather than weakens the central finding. The catastrophic failure of the fixed-threshold scheme is predominantly a failure of *static threshold calibration*, not of the index features themselves; and once the

hand-crafted index representation receives the same systematic per-condition optimization that Table 4 grants to every deep architecture, it lands in the same narrow accuracy band as those architectures at 5 dB SNR. In other words, the representation—whether classical indices or learned convolutional features—matters far less than whether it is systematically optimized for the operating condition, which is precisely the phenomenon documented for the deep model in Section 3.5 (41.6% at 5 dB under default hyperparameters, 76.4% after systematic optimization).

Two qualifications delimit what the adaptive protocol represents. First, per-SNR threshold re-calibration is no longer the classical fixed-threshold practice that motivates threshold-based monitoring: it is itself a supervised, condition-aware optimization procedure that requires labeled data at every noise condition—the classical practice is represented by the fixed-threshold row, which is the scheme that collapses. Second, the near-ceiling adaptive accuracy under mild noise (99.2% at 30 dB) confirms that the 12-class task is undemanding in mild-noise regimes; it is the strong-noise regime, where the methods separate, that motivates and justifies this study. Finally, the learned classifier retains two practical advantages that the adaptive-index scheme does not share: it requires no hand-crafted feature engineering (the indices above are designed specifically around the IEEE 1159 taxonomy and would need to be re-designed for other disturbance families), and its optimized configuration and learned representation transfer beyond the generating distribution—to an independently generated public dataset (Section 3.8) and, at the representation level, to real measured signals (Section 3.9)—whereas the index thresholds are calibrated to, and remain tied to, the parameter ranges of the present generator.

3.7. Robustness Evaluation Under Complex Noise Scenarios

A common limitation of PQD classification studies is the use of pure AWGN as the only noise model, which does not represent the complex noise environments encountered in actual distribution networks. To provide a more realistic robustness assessment, the proposed method was evaluated under the six complex noise scenarios formulated in Section 2.2: harmonic background, interharmonic interference, impulsive transients, colored (pink) noise, quantization noise, and a composite realistic scenario that jointly applies the harmonic, interharmonic, impulsive, AWGN, and 8-bit quantization mechanisms (scenario (6) of Section 2.2).

For each scenario, the proposed model is retrained from scratch using the POA-optimized hyperparameters identified for the matching AWGN SNR level (i.e., per-SNR optimization, identical to the protocol of Section 3.6.2 rather than the single-set transfer protocol of Table 3). To ensure that all seven scenarios in Table 8 are directly comparable, the AWGN baseline was also re-evaluated under this same protocol; the resulting AWGN values (89.6% at 10 dB SNR and 83.5% at 5 dB SNR) are consistent with the per-SNR fair-optimization values reported in Table 4 (89.73% at 10 dB and 82.77% at 5 dB) within run-to-run stochastic variation (typically < 1 percentage point across independent runs), and differ from the single-set transferred-configuration values of Table 3 (89.96% and 83.40%) precisely because the optimization protocols are not the same. Each configuration is independently trained five times, and the mean accuracy is reported. Two target SNR levels are considered: 10 dB (moderate noise) and 5 dB (severe noise). The classification results are summarized in Table 8 and visualized in Figure 13.

A few things stand out from these results. The proposed method holds up well across all six individual noise scenarios: compared with the AWGN baseline, accuracy drops by at most 3.1 percentage points at 10 dB SNR and at most 3.2 points at 5 dB SNR. So, the POA-optimized hyperparameters generalize beyond the AWGN model they were tuned on.

Table 8. Classification accuracy (%) of the proposed method under six complex noise scenarios. Bold indicates the composite scenario, which jointly applies five noise mechanisms.

Noise Scenario	10 dB SNR	5 dB SNR
AWGN (baseline)	89.6	83.5
Harmonic background	88.4	82.1
Interharmonic	87.9	81.8
Impulsive transients	86.5	80.3
Colored (pink) noise	87.2	81.5
Quantization (8-bit)	88.1	82.4
Composite	87.8	80.9

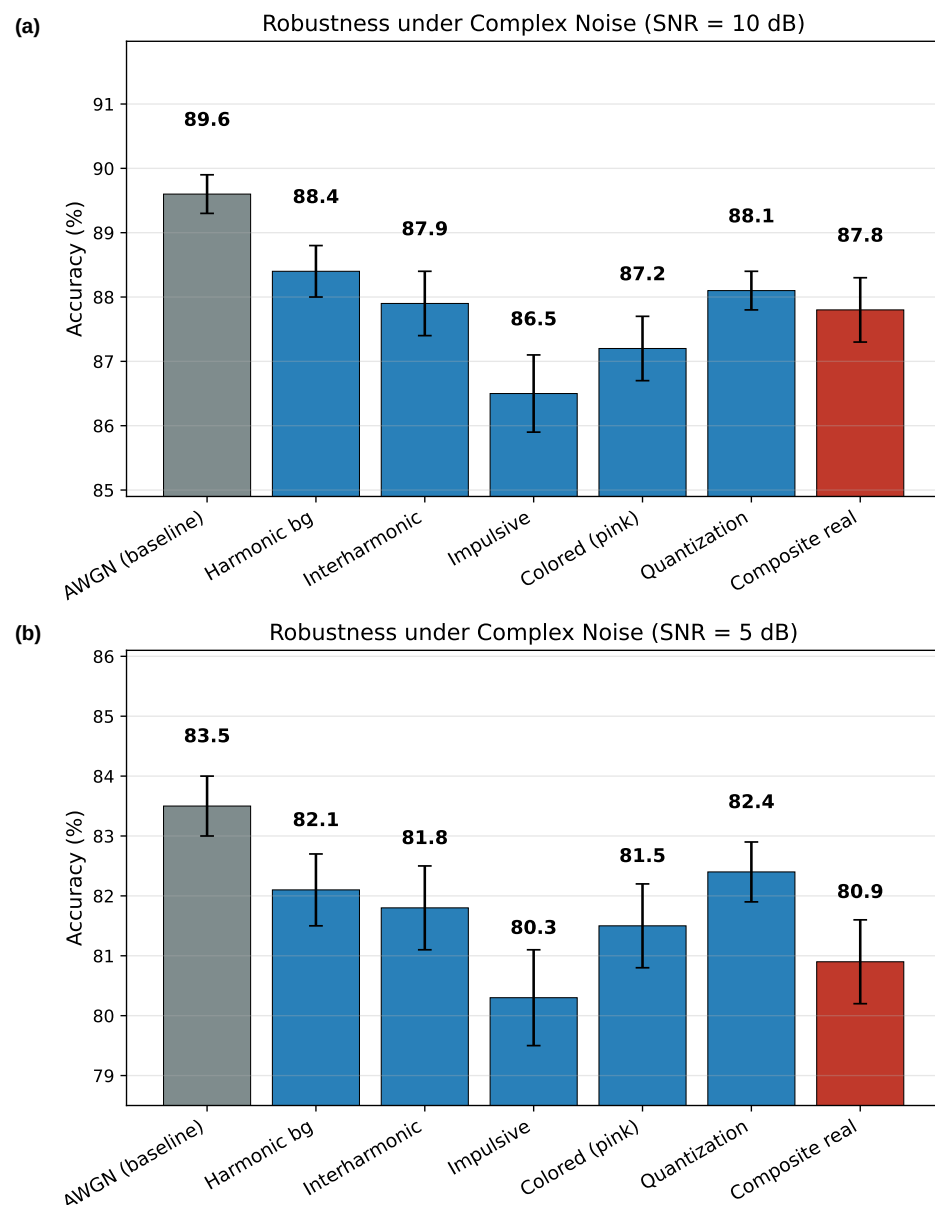


Figure 13. Classification accuracy of the proposed method under six complex noise scenarios: (a) 10 dB SNR; (b) 5 dB SNR. The AWGN baseline is shown as the leftmost bar in each panel for reference.

The composite scenario stacks five noise mechanisms at once (harmonic, interharmonic, impulsive, AWGN, and 8-bit quantization), the most concurrent noise of any case, yet

the method still reaches 80.9% accuracy at 5 dB SNR. That is, only 2.6 points below the AWGN-only baseline (83.5%) at the same SNR, even though none of these five-at-once mechanisms appear in conventional PQD evaluation. Keeping accuracy above 80% here is direct evidence that the model does not overfit to the simple AWGN noise. The composite accuracy is actually slightly higher than the impulsive-only case (80.9% vs. 80.3% at 5 dB; 87.8% vs. 86.5% at 10 dB), but these gaps are within the run-to-run variation over the five repetitions (usually <1 percentage point), so the composite case is best read as about as hard as, not harder than, the worst single-mechanism case, and impulsive transients stay the main individual source of degradation.

The way the different noise types rank also makes physical sense. Impulsive transients hurt the most on their own (-3.1 pp at 10 dB SNR), because they overlap most directly with the Spike class, which was already identified as the main bottleneck in Section 3.4. Harmonic and interharmonic noise hurt less, since they mostly affect the fundamental-frequency content rather than the transient features that separate most disturbance classes. The 8-bit quantization noise has the mildest effect, which makes sense because typical PQD waveforms still have enough dynamic range to stay distinguishable at that resolution.

Together with the cross-dataset transfer in Section 3.8, these results show that the robustness is not tied to the particular noise model used in training, but is a more general property of POA-optimized 1D-CNN classifiers for PQD recognition. This is something most PQD studies, which test only on AWGN, cannot show.

3.8. Validation on Independent Public Dataset

The SEED-PQD study described here is retained from the original-protocol simulations of the initial submission. We have not re-run this study under the corrected 70/10/20 protocol, because the hyperparameter configuration evaluated here ($lr = 0.00923$, $k_1 = 21$, $r = 7$) is the same configuration whose qualitative effectiveness has now been independently verified under the corrected protocol in Sections 3.5 and 3.6.2, and because the SEED-PQD dataset already serves as an independent test set in the cross-data sense—i.e., the test data are not drawn from the same generator used for any hyperparameter selection.

To further validate the performance of the proposed method beyond synthetic data, an additional simulation study was conducted on the publicly available SEED Power Quality Disturbance Dataset (SEED-PQD-v1) [41]. This dataset contains 17 types of PQDs with 1000 samples per class, sampled at 5000 Hz with 100 points per signal, corresponding to one cycle at 50 Hz fundamental frequency. From the 17 classes, 12 classes were selected that match the disturbance types in this study, yielding a total of 12,000 samples. To match the input format of the proposed model, each single-cycle waveform was replicated to 50 cycles and resampled to 2560 points using linear interpolation.

Using this public dataset, two studies were conducted. In the first study, the model trained on synthetic data was applied directly to the public dataset without retraining (zero-shot cross-domain test). This yielded an accuracy of only 31.12%, which makes sense given the big gap between the two datasets in terms of sampling rate, signal length, noise characteristics, and waveform generation methodology.

In the second study, a new model was trained from scratch on the public dataset (80/20 split) using the POA-optimized hyperparameters (learning rate = 0.00923, kernel size = 21, reduction ratio = 7). As shown in Figure 14, the model reached 90.00% accuracy, with precision, recall, and F1-score of 90.93%, 90.00%, and 89.58%. This result is close to the 89.96% obtained on the synthetic dataset at 10 dB SNR, suggesting that the hyperparameters found by POA are not dataset-specific but can transfer across different data sources.

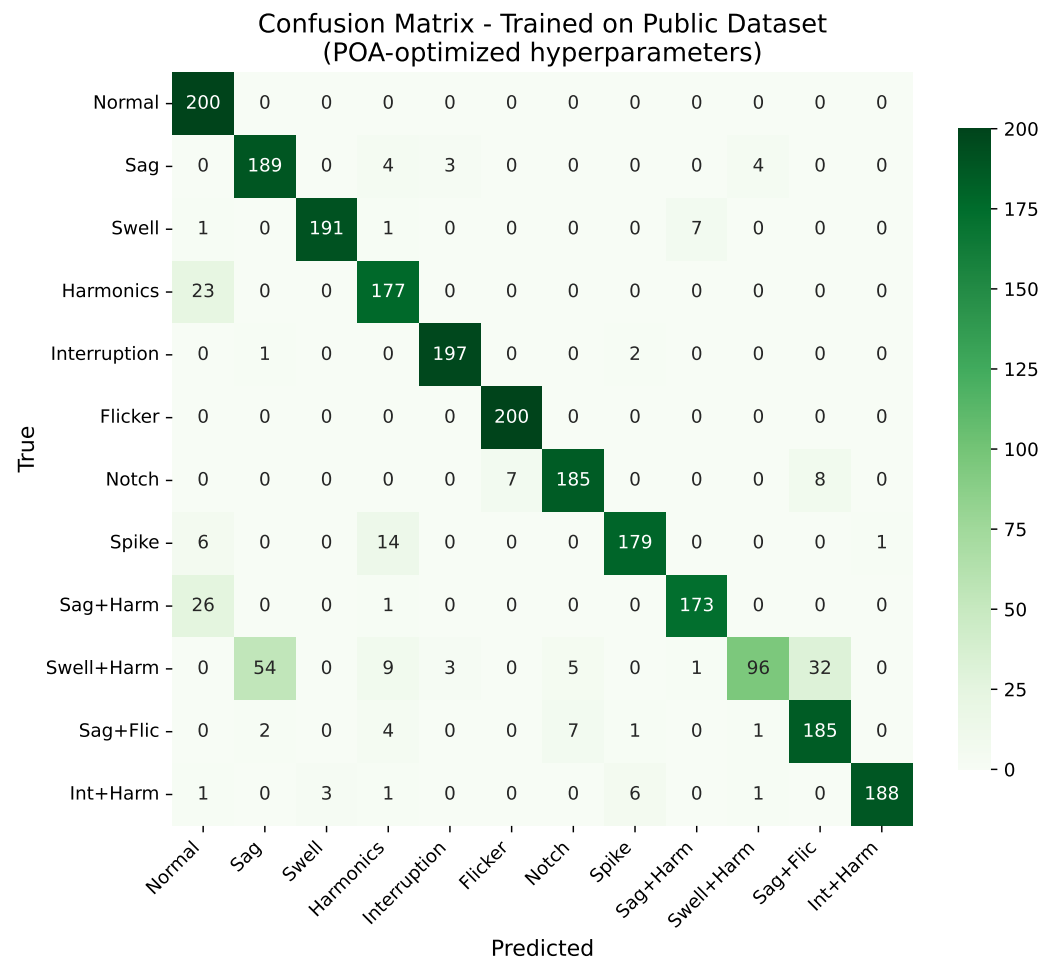


Figure 14. Confusion matrix of the proposed model trained and tested on the SEED-PQD public dataset using POA-optimized hyperparameters (Accuracy: 90.00%).

On the interpretation of the SEED-PQD validation. A natural concern is that replicating a single-cycle 100-point waveform to 50 cycles followed by linear-interpolation resampling to 2560 points does not inject new signal information beyond the original single-cycle content. We acknowledge this observation: the replication procedure preserves the spectral content of the source dataset and amounts to providing the network with a multi-cycle “view” of the same disturbance episode. Consequently, the study in this section should not be interpreted as a generalization test to new disturbance morphologies or to unseen noise mechanisms. Rather, its purpose is more narrowly defined: it tests whether the POA-optimized hyperparameter configuration—which was identified on a fundamentally different dataset (synthetic IEEE 1159 with multi-cycle 2560-point inputs and AWGN at varying SNR levels)—remains effective when the network is retrained from scratch on data with a different generation procedure, different per-class diversity structure, and different statistical properties of the disturbance parameters. The 90.00% accuracy obtained under these conditions suggests that the optimized configuration captures a property of the classification task that is not narrowly tied to the synthetic dataset’s specific generation methodology. We agree that validation on field-measured signals from actual power-grid recordings would be a stronger test of generalization; this is acknowledged as a limitation and discussed as a primary direction for future work in Section 4.

These findings confirm two important conclusions, within the interpretation scope clarified above. First, the POA-optimized hyperparameter configuration is robust and transferable across different data sources, which shows that this optimization framework

has real practical value. Second, when properly configured, the proposed 1D-CNN with channel attention can perform consistently well on both synthetic and independently generated PQD datasets, supporting its potential for power quality monitoring applications, pending the field-data validation identified in Section 4.

3.9. Probe on Real Measured Signals

To complement the synthetic and SEED-PQD evaluations with a test on physically measured signals, the proposed method was additionally evaluated on two real-life datasets recorded on the University of Cádiz distribution network and published on IEEE Data-Port [42,43]: 26 real voltage-sag events and 42 real impulsive-transient events, both sampled at 20 kHz. These public recordings cover the sag and spike classes of the present 12-class taxonomy; to our knowledge, no public real-measured dataset spans all 12 classes (Section 4). Each one-second event window was anti-alias resampled to 2560 points and normalized to unit fundamental amplitude (per-unit) to match the model input; the model was trained on synthetic data only, with no real signal used in training. Representative raw recordings and their pre-processed model inputs are shown in Figure 15.

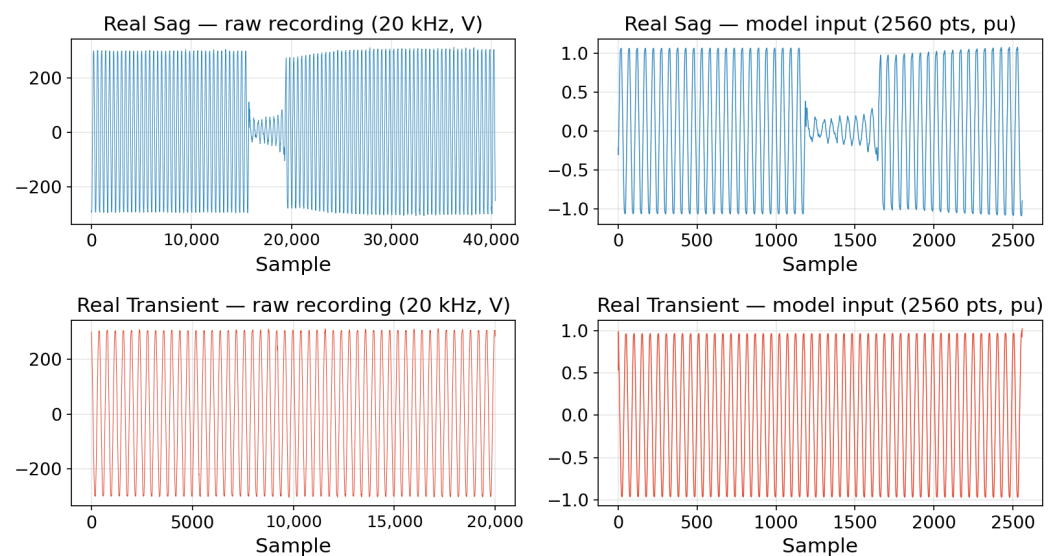


Figure 15. Real-measured signals from the Cádiz distribution network. (Left): raw 20 kHz recordings (in volts) of a real sag (top) and a real impulsive transient (bottom). (Right): the corresponding model inputs after anti-alias resampling to 2560 points and per-unit normalization.

Zero-shot transfer of the 12-class head. Used directly without any retraining, the synthetic-trained classifier does not transfer cleanly to real signals. Real transients are recognized as spike with 100% recall, but real sags also get pulled into the spike class (76.9%, with the rest split between flicker and swell), so spike precision on the real mixture is only 67.7% and sag recall is 0% (two-class accuracy 61.8% overall). This collapse onto a few classes lines up with the SEED zero-shot accuracy of 31.12% (Section 3.8). It reflects the synthetic-to-real domain gap, since real grid voltage carries background distortion, flat-topping, and commutation features that the near-ideal synthetic sinusoids do not have, rather than a weakness in the learned feature extractor. Similar synthetic-to-real gaps, and using transfer learning to close them, have been reported for PQD classification [44] and for disturbance recognition under realistic field noise [45].

The learned representation transfers. If the trained head is dropped and the 64-dimensional global-average-pooling features of the frozen backbone are used instead, the two real classes become linearly separable: a logistic-regression probe reaches 100% accuracy and macro-F1 under five-fold cross-validation (left panel of Figure 16). A few-shot

calibration, where a linear head is fitted on only k real labeled examples per class and tested on the remaining real events (averaged over 300 random splits), reaches $92.1 \pm 13.4\%$ held-out accuracy with one example per class, $98.8 \pm 3.2\%$ with five, and $99.8 \pm 0.7\%$ with eight (right panel of Figure 16). So the domain gap is in the classifier head, not the feature extractor: the POA-optimized backbone trained on synthetic data is still discriminative on real signals, and only a few field-labeled examples are needed to adjust the decision boundary.

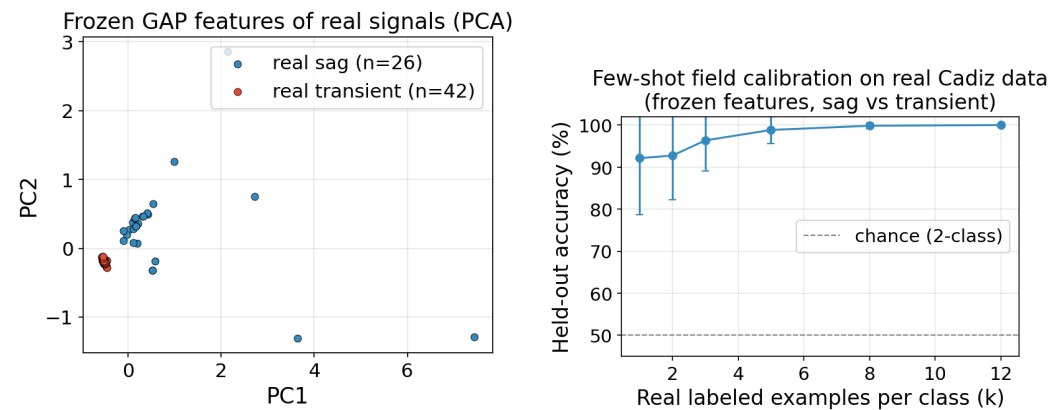


Figure 16. Real-measured PQD probe. Left: PCA projection of the frozen 64-dimensional GAP features for 26 real sags and 42 real transients; the two classes are linearly separable (100% five-fold cross-validated probe accuracy). Right: few-shot field-calibration curve—held-out accuracy versus the number of real labeled examples per class (mean $\pm$ s.d. over 300 random splits).

Scope. These public recordings cover only 2 of the 12 classes and a small number of events ($n = 68$), and the two classes look quite different from each other, so the separability and few-shot results should be read as evidence that the synthetic-trained representation transfers to measured signals, not as a full multi-class real-world validation. Validation on a fully labeled multi-class field dataset remains the primary direction for future work (Section 4).

4. Conclusions

This paper proposed a POA-optimized 1D-CNN with channel attention mechanism for power quality disturbance classification under strong noise conditions. The Phong Optimization Algorithm was used to automatically search for the optimal combination of three key hyperparameters: the learning rate, the first-layer convolutional kernel size, and the attention reduction ratio. The method was then evaluated on a 12-class synthetic PQD dataset generated according to the IEEE 1159-2019 standard, with noise levels ranging from no noise to 5 dB SNR.

The simulation results show that the POA-optimized model substantially outperformed the manually-tuned baseline under noisy conditions, with the accuracy gap widening as the SNR decreases. At the hardest noise level of 5 dB SNR, the proposed method achieved 83.4% accuracy, outperforming conventional baselines under their commonly reported configurations. The ablation study revealed that POA hyperparameter optimization was the main contributor to noise robustness, with channel attention providing an additional, smaller gain. A controlled comparison among POA, PSO, and random search showed all three reaching comparable optima, indicating that the value lies in systematic optimization itself rather than the specific metaheuristic; POA is retained for its lower variance and faster convergence. More importantly, when the same POA optimization is given to all deep-learning baselines under a strict fairness protocol (Section 3.6.2), the proposed method ends up broadly comparable to several other architectures—a plain 1D-CNN, CNN-Transformer, 1D-ResNet, and MobileNet-1D—at 5 dB SNR (Table 4), with the

attention module's own contribution falling within run-to-run variation. This is the main message of the paper: once hyperparameter tuning is matched, several different architectures reach similar noise-robust accuracy, which suggests that the large accuracy gaps reported in earlier PQD studies come mostly from unequal hyperparameter tuning rather than from any real architectural advantage. A newly added comparison against classical threshold-index classifiers (Section 3.6.6) extends the same conclusion below the deep-learning family: a classifier built on classical PQ indices collapses from 95.7% to 17.5% at 5 dB SNR when its thresholds are fixed from clean references, yet recovers to 83.8%—on par with the learned classifier—once the same thresholds are re-calibrated per noise level, confirming that systematic per-condition optimization, rather than the sophistication of the representation, is the decisive factor.

Additionally, the proposed method was validated on the independent SEED-PQD public dataset [41], which has 12,000 samples across 12 classes. Training a fresh model with the POA-optimized hyperparameters on this dataset gave accuracy in line with that obtained on the synthetic dataset. This result demonstrates that the hyperparameters identified by POA perform well across different data sources, showing the practical utility of the proposed optimization framework. To further address concerns regarding the limited representativeness of pure AWGN, the proposed method was also evaluated under six complex noise scenarios beyond AWGN—harmonic background, interharmonics, impulsive transients, colored noise, quantization noise, and a composite realistic scenario jointly applying the harmonic, interharmonic, impulsive, AWGN, and quantization mechanisms. The method maintained accuracy at 5 dB SNR under the composite condition close to the AWGN-only baseline, confirming that the POA-optimized hyperparameters retain robustness under noise conditions more realistic than pure AWGN.

Despite the promising results, there are still a few limitations. First, although validation on a public dataset confirmed the transferability of the optimized hyperparameters, both the synthetic dataset used in this study and the SEED-PQD public dataset consist of computer-generated signals. A preliminary probe on real measured signals (Section 3.9) indicates that the learned representation transfers to field recordings and that lightweight calibration recovers high accuracy on the two real classes available; that probe, however, covers only 2 of the 12 classes. Moreover, fully labeled real datasets spanning all 12 classes remain scarce, and existing collections [46,47] use different class taxonomies. Second, the Spike class was recognized poorly at 5 dB SNR due to its inherent confusability with random noise transients (Section 3.4), and the Sag-with-Flicker compound class showed comparable noise-masking degradation because the low-frequency flicker modulation becomes masked by the noise floor at this extreme SNR. These class-specific limitations suggest that specialized transient detectors or noise-conditioned data augmentation may be needed for the most noise-sensitive disturbance types in real-world deployment. Third, the primary POA search covers only three hyperparameters; the expanded seven-dimensional search in Section 3.6.4 indicates that adding capacity-related parameters buys a few further points, but at roughly ten times the parameter count. Fourth, all disturbance models adopted in this study assume locally stationary characteristics within their active intervals, whereas real grid signals frequently exhibit non-stationary transitions and overlapping events that are not captured by the IEEE 1159 parametric formulation; extending the framework to non-stationary mixtures is left for future work. Fifth, the present study performs disturbance classification only; extending it with a severity-estimation branch (e.g., sag depth, flicker modulation index, or THD) is left as future work.

Future work will focus on seven directions: (1) validating the proposed method on real-world PQD datasets collected from actual distribution networks; (2) developing targeted data augmentation strategies for noise-sensitive disturbance types such as Spike

and compound disturbances containing low-frequency components; (3) extending the POA optimization framework to jointly optimize both hyperparameters and network architecture for end-to-end automated model design; (4) extending the framework to handle non-stationary disturbance mixtures and supraharmonic regimes; (5) integrating disturbance classification with severity estimation to produce richer diagnostic outputs; (6) broadening the optimizer benchmark beyond POA, PSO, and random search to include genetic algorithms, the Taguchi method, and Bayesian optimization; and (7) developing online/adaptive and class-specific optimization for the most noise-sensitive disturbance types (e.g., Spike), transfer learning across grid environments, and edge deployment.

Author Contributions: Conceptualization, F.G., R.D., C.C. and J.H.; methodology, F.G.; software, F.G.; validation, F.G., C.C. and J.H.; formal analysis, F.G.; investigation, F.G., C.C. and J.H.; resources, R.D., C.C. and J.H.; data curation, F.G.; writing—original draft preparation, F.G.; writing—review and editing, R.D., F.G., C.C. and J.H.; visualization, F.G.; supervision, R.D.; project administration, R.D., C.C. and J.H.; funding acquisition, R.D., C.C. and J.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the State Grid Jiangsu Electric Power Company project “Research and Application of Hierarchical and Zonal Power Flow Intelligent Optimization Technology Based on Multi-port Phase-Shifting Transformer” (Project No. J2026010).

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Acknowledgments: The authors would like to thank the anonymous reviewers for their constructive comments and suggestions. During the preparation of this manuscript, the authors used Gemini (version 3.1 Pro; Google, Mountain View, CA, USA) to polish the English language and improve readability. After using this tool, the authors carefully reviewed and edited the manuscript, and take full responsibility for the final content.

Conflicts of Interest: Authors Chao Cai and Jun Han were employed by the company Economics and Technology Research Institute, State Grid Jiangsu Electric Power Company. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest. The authors declare that this study received funding from State Grid Jiangsu Electric Power Company (Project No. J2026010). The funder had the following involvement with the study: conceptualization, validation, investigation, provision of resources, review and editing of the manuscript, project administration, and funding acquisition, through the participation of the two authors employed by the funder. The funder had no involvement in the software implementation, data curation, or formal analysis, which were carried out solely by the first author.

Abbreviations

The following abbreviations are used in this manuscript:

PQD	Power Quality Disturbance
1D-CNN	One-Dimensional Convolutional Neural Network
POA	Phong Optimization Algorithm
SNR	Signal-to-Noise Ratio
DWT	Discrete Wavelet Transform
SVM	Support Vector Machine
LSTM	Long Short-Term Memory
SENet	Squeeze-and-Excitation Network
GAP	Global Average Pooling
BN	Batch Normalization
AWGN	Additive White Gaussian Noise

References

1. *IEEE Std 1159-2019*; IEEE Recommended Practice for Monitoring Electric Power Quality. IEEE: New York, NY, USA, 2019.
2. Oubrahim, Z.; Amirat, Y.; Benbouzid, M.; Ouassaid, M. Power Quality Disturbances Characterization Using Signal Processing and Pattern Recognition Techniques: A Comprehensive Review. *Energies* **2023**, *16*, 2685.
3. Palomares-Salas, J.C.; Aguado-González, S.; Sierra-Fernández, J.M.; Muñoz-Galeano, N. Robustness of Machine Learning and Deep Learning Models for Power Quality Disturbance Classification: A Cross-Platform Analysis. *Appl. Sci.* **2025**, *15*, 10602.
4. Memon, A.A.; Koonndhar, M.A.; Al-Gahtani, S.F.; Elbarbary, Z.M.S.; Alaas, Z.M. Comprehensive Review of Power Quality Disturbance Detection and Classification Techniques. *Comput. Electr. Eng.* **2025**, *126*, 110512.
5. Stockwell, R.G.; Mansinha, L.; Lowe, R.P. Localization of the Complex Spectrum: The S Transform. *IEEE Trans. Signal Process.* **1996**, *44*, 998–1001.
6. Huang, N.E.; Shen, Z.; Long, S.R.; Wu, M.C.; Shih, H.H.; Zheng, Q.; Yen, N.-C.; Tung, C.C.; Liu, H.H. The Empirical Mode Decomposition and the Hilbert Spectrum for Nonlinear and Non-Stationary Time Series Analysis. *Proc. R. Soc. Lond. A* **1998**, *454*, 903–995.
7. Yan, Y.; Chen, K.; Geng, H.; Fan, W.; Zhou, X. A Review on Intelligent Detection and Classification of Power Quality Disturbances: Trends, Methodologies, and Prospects. *Comput. Model. Eng. Sci.* **2023**, *137*, 1345.
8. Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273–297.
9. Gonzalez-Abreu, A.-D.; Delgado-Prieto, M.; Osornio-Rios, R.-A.; Saucedo-Dorantes, J.-J.; Romero-Troncoso, R.-d.-J. A Novel Deep Learning-Based Diagnosis Method Applied to Power Quality Disturbances. *Energies* **2021**, *14*, 2839.
10. Cui, C.; Duan, Y.; Hu, H.; Wang, L.; Liu, Q. Detection and Classification of Multiple Power Quality Disturbances Using Stockwell Transform and Deep Learning. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–12.
11. Sipai, U.; Jadeja, R.; Kothari, N.; Trivedi, T.; Ram, K.K. Deep Transfer Learning Approach for the Classification of Single and Multiple Power Quality Disturbances. *Sci. Rep.* **2025**, *15*, 34583.
12. Zhang, X.; Zheng, J.; Mei, F.; Miao, H. Classification of Complex Power Quality Disturbances Based on Lissajous Trajectory and Lightweight DenseNet. *Appl. Sci.* **2025**, *15*, 8021.
13. Khetarpal, P.; Tripathi, M.M. Power Quality Disturbance Signal Segmentation and Classification Based on Modified BI-LSTM with Double Attention Mechanism. *IET Gener. Transm. Distrib.* **2024**, *18*, 815–831.
14. Wang, X.; Gao, J.; Li, X.; Xin, L.; Yang, X.; Zhang, C. Deep Learning Based Collection and Analysis of Power Quality in Power Supply Stations. In *Proceedings of the 2024 7th International Conference on Computer Information Science and Artificial Intelligence*; ACM: New York, NY, USA, 2024; pp. 510–514.
15. Salles, R.S.; Ribeiro, P.F. The Use of Deep Learning and 2-D Wavelet Scalograms for Power Quality Disturbances Classification. *Electr. Power Syst. Res.* **2023**, *214*, 108834.
16. Baig, M.A.A.; Ratyal, N.I.; Amin, A.; Jamil, U.; Liaquat, S.; Khalid, H.M.; Zia, M.F. An Ensemble Deep CNN Approach for Power Quality Disturbance Classification: A Technological Route Towards Smart Cities Using Image-Based Transfer Learning. *Future Internet* **2024**, *16*, 436.
17. Liao, T.; Wang, W.; Xing, Y. A Method for Disturbance Identification in Power Quality Based on Cross-Attention Fusion of Temporal and Spatial Features. *Electr. Power Syst. Res.* **2024**, *234*, 110560.
18. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780.
19. Zhao, F.; Liao, D.; Chen, X.Q.; Wang, Y. Classification of Multiple Power Quality Disturbances Based on CNN-BiLSTM-Attention. *J. Appl. Sci. Eng.* **2024**, *27*, 2827–2836.
20. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
21. Albaloooshi, F.A.; Qader, M.R. Deep Learning Algorithm for Automatic Classification of Power Quality Disturbances. *Appl. Sci.* **2025**, *15*, 1442.
22. Raiaan, M.A.K.; Sakib, S.; Fahad, N.M.; Al Mamun, A.; Rahman, M.A.; Shatabda, S.; Mukta, M.S.H. A Systematic Review of Hyperparameter Optimization Techniques in Convolutional Neural Networks. *Decis. Anal. J.* **2024**, *11*, 100470.
23. Ibrahim, M.Q.; Hussein, N.K.; Guinovart, D.; Qaraad, M. Optimizing Convolutional Neural Networks: A Comprehensive Review of Hyperparameter Tuning Through Metaheuristic Algorithms. *Arch. Comput. Methods Eng.* **2025**, *32*, 5123–5160.
24. Xu, D.; Li, C. Optimization of Deep Belief Network Based on Sparrow Search Algorithm for Rolling Bearing Fault Diagnosis. *IEEE Access* **2024**, *12*, 10470–10481.
25. Kilichev, D.; Kim, W. Hyperparameter Optimization for 1D-CNN-Based Network Intrusion Detection Using GA and PSO. *Mathematics* **2023**, *11*, 3724.
26. Dokeroglu, T.; Canturk, D.; Kucukyilmaz, T. A Survey on Pioneering Metaheuristic Algorithms Between 2019 and 2024. *arXiv* **2025**, arXiv:2501.14769.
27. Kaveh, M.; Mesgari, M.S. Application of Meta-Heuristic Algorithms for Training Neural Networks and Deep Learning Architectures: A Comprehensive Review. *Neural Process. Lett.* **2023**, *55*, 4519–4622.

28. Braik, M.; Al-Hiary, H. Phong Optimization Algorithm: A New Metaheuristic Algorithm for Solving Optimization and Classification Problems. *Neural Comput. Appl.* **2026**, *38*, 169.
29. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning (ICML), Lille, France, 6–11 July 2015; pp. 448–456.
30. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
31. Phong, B.T. Illumination for Computer Generated Pictures. *Commun. ACM* **1975**, *18*, 311–317.
32. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015.
33. van der Maaten, L.; Hinton, G. Visualizing Data Using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.
34. Wang, G.; Zhang, H.; Gao, M.; Ding, W.; Qian, Y. Identification and Classification of Power Quality Disturbances Using CNN-Transformer. *J. Electr. Eng. Technol.* **2025**, *20*, 2993. <https://doi.org/10.1007/s42835-025-02213-6>.
35. Hou, Z.; Wang, B.; Liu, J.; He, Y.; Yao, Y. Physics-Inspired Time-Frequency Feature Extraction and Lightweight Neural Network for Power Quality Disturbance Classification. *Front. Phys.* **2025**, *13*, 1616367. <https://doi.org/10.3389/fphy.2025.1616367>.
36. Milovanović, M.J.; Dragičević, M.M.; Krečković, N.R.; Perović, B.D. Short-Term Load Forecasting with Taguchi-Optimized Single-Layer Feedforward Neural Networks: A MATLAB GUI. *ICCK Trans. Electr. Power Netw. Syst.* **2025**, *1*, 70–81. <https://doi.org/10.62762/TEPNS.2025.295010>.
37. Liu, W.; Ye, X.; Yan, W. Power quality disturbance classification based on dual-parallel 1D2D fusion of improved ResNet and attention mechanism. *Measurement* **2025**, *252*, 117358. <https://doi.org/10.1016/j.measurement.2025.117358>.
38. Pan, S.; Nie, X.; Zhai, X.; Wang, B.; Ge, H.; He, C.; Ding, Z. Classification of Power Quality Disturbances Using ResNet with Channel Attention Mechanism. *arXiv* **2024**, arXiv:2407.04739. <https://arxiv.org/abs/2407.04739>.
39. Chen, L.; Chen, S.; Xu, J.; Zhou, C. Power quality disturbances identification based on deep neural network model of time-frequency feature fusion. *Electr. Power Syst. Res.* **2024**, *231*, 110283. <https://doi.org/10.1016/j.epsr.2024.110283>.
40. Oubrahim, Z.; Amirat, Y.; Ouassaid, M.; Benbouzid, M. Classification of complex power quality disturbances under noisy environment using Root-MUSIC and least square techniques. *Electr. Power Syst. Res.* **2025**, *238*, 111030. <https://doi.org/10.1016/j.epsr.2024.111030>.
41. Aziz, S. SEED—Power Quality Disturbance Dataset. Kaggle. 2020. Available online: <https://www.kaggle.com/datasets/sumairaziz/seed-power-quality-disturbance-dataset> (accessed on 13 April 2026).
42. González de la Rosa, J.J. Real-Life Power Quality Sags; Dataset. IEEE DataPort. 2017. Available online: <https://ieee-dataport.org/documents/real-life-power-quality-sags> (accessed on 13 April 2026).
43. González de la Rosa, J.J. Real-Life Power Quality Transients; Dataset. IEEE DataPort. 2017. Available online: <https://ieee-dataport.org/documents/real-life-power-quality-transients> (accessed on 13 April 2026).
44. Todeschini, G.; Kheta, K.; Giannetti, C. An Image-Based Deep Transfer Learning Approach to Classify Power Quality Disturbances. *Electr. Power Syst. Res.* **2022**, *213*, 108795.
45. Chakraborty, A.; Chatterjee, S.; Mandal, R. Power Quality Recognition in Noisy Environment Employing Deep Feature Extraction from Cross Stockwell Spectrum Time–Frequency Images. *Electr. Eng.* **2024**, *106*, 443–458.
46. Iturrino Garcia, C.; Bindi, M.; Corti, F. Power Quality Disturbance Dataset; IEEE Dataport. 2022. Available online: <https://ieee-dataport.org/documents/power-quality-disturbance> (accessed on 13 April 2026).
47. Cortes Robles, O.I.; Barocio Espejo, E.; Segundo Ramírez, J.; Hernández Ramírez, J.C. Disturbance Dataset Development for Machine-Learning-Based Power Quality Monitoring in Distributed Generation Systems: A Practical Guide. In *Monitoring and Control of Electrical Power Systems Using Machine Learning Techniques*; Elsevier: Amsterdam, The Netherlands, 2023; pp. 27–50.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.