Adaptive Denoising of Partial Discharge Using Absolute Difference Optimization Versus Artificial Neural Networks

Kui-Fern Chin[1], Chang-Yii Chai[2], Ismail Saad[3], Yee-Ann Lee[4]
Faculty of Engineering & Technology, *i*-CATS University College, Kuching, Malaysia[1]
Faculty of Engineering, University Malaysia Sabah, Kota Kinabalu, Malaysia[2, 3]
College of Computing-Informatics and Mathematics, University Teknologi MARA Sarawak Branch, Kota Samarahan, Malaysia[4]

***Abstract*—Accurate partial discharge (PD) localization in medium-voltage (MV) power cables is essential for condition-based maintenance, yet it remains unreliable when PD pulses are masked by broadband noise and narrowband interference. The novelty of this work is a controlled denoiser-to-localization benchmarking framework that isolates the denoising front end, while keeping the downstream PD detection and localization backend fixed, allowing localization differences to be attributed solely to denoising decisions. Within this fixed-backend paradigm, an optimization-driven Adaptive Denoising Optimization (ADO) method is introduced as an adaptive discrete wavelet transform (DWT) front end that systematically selects the mother wavelet, decomposition level, and threshold parameters to preserve time-of-arrival (ToA) critical wavefront features rather than only maximizing noise suppression. ADO is evaluated against two learning-based denoisers, a multilayer artificial neural network (ANN) and a lightweight feedforward neural network (FNN), using MATLAB simulations of synthetic PD pulses corrupted by white Gaussian noise (WGN) and discrete spectral interference (DSI) over SNRs from 9.78 dB to -10.34 dB. Performance is quantified using execution time, percentage localization error (PE), median absolute localization error (MedAE), and F1 score. Results show that ADO delivers the most robust localization fidelity, maintaining near-zero PE above -6 dB, keeping PE below 0.3% at -10.34 dB, achieving sub-metre MedAE, and sustaining F1 close to 1.0 across noise levels. In contrast, FNN is the fastest option, reducing runtime by approximately 15% versus ANN and 27% versus ADO, highlighting a practical robustness-efficiency trade-off for real-time MV cable monitoring.**

***Keywords*—Partial discharge localization; adaptive denoising optimization; discrete wavelet transform; artificial neural network**

I. Introduction

Partial discharge (PD) in medium-voltage (MV) power cables is a well-established indicator of insulation deterioration. If PD is not detected and localized reliably, degradation can accelerate, outage risk increases, and maintenance and replacement costs rise. Accurate localization remains challenging because PD pulses are short, broadband, and nonstationary, and are easily distorted or masked by background noise and electromagnetic interference (EMI). As a result, recent studies have combined physics-informed localization with advanced signal analytics, including time-of-arrival (ToA) formulations that account for cable propagation effects such as skin effect and semiconducting layers [1], reviews of ultra-high-frequency (UHF) PD signal processing and machine learning [2], electromagnetic time reversal (EMTR) methods for online MV cable localization with very low error [3], deep models for PD pattern recognition in cable accessories and sensing arrays [4], [5], and unsupervised deep learning for separating multiple PD sources in other high-voltage assets [6]. Collectively, this literature confirms the importance of PD localization, while also showing that realistic interference remains a primary barrier to dependable field deployment.

This barrier is closely tied to a common assumption embedded in many localization backends, including ToA, time-domain reflectometry (TDR), and correlation-based estimators: the first-arriving PD wavefront can be extracted clearly enough to preserve timing cues. In field-like MV cable monitoring, this assumption often fails because measurements are frequently contaminated by white Gaussian noise (WGN) and discrete spectral interference (DSI), which can shift, smooth, or obscure the onset used for ToA estimation. Prior studies and surveys highlight that noise strongly constrains PD feature extraction and localization fidelity [2], [7], including efforts to separate PD from EMI under WGN and DSI conditions [8] and analyses showing that timing-based methods degrade in long cables when propagation effects and noise jointly distort the pulse [9]. These findings motivate denoising as a critical front-end stage, since overall localization error can be dominated by preprocessing rather than by the mathematical form of the backend estimator.

Wavelet-domain denoising, particularly discrete wavelet transforms (DWT) processing, is widely used in PD analysis because it is well matched to transient multiscale signals and provides time-frequency localization of impulsive events [10-14]. However, conventional DWT denoising typically relies on fixed choices of mother wavelet, decomposition level, and thresholding rule that are tuned to a specific dataset or laboratory condition. When the interference spectrum changes or the signal-to-noise ratio (SNR) deteriorates, fixed settings may leave residual noise that masks the first wavefront, or they may distort PD timing through mismatched wavelet bases and over-thresholding [15]. Recent investigations further show that denoising quality is highly sensitive to the wavelet family, decomposition depth, and threshold strategy, and that the best configuration varies with the dominant noise type, implying that no single static setting is consistently optimal across operating conditions [16], [17]. In parallel, learning-based denoisers such as artificial neural networks (ANNs) and related deep

architectures can learn nonlinear noise suppression mappings and have reported strong denoising performance [18-20]. Nevertheless, their effectiveness can degrade when test noise statistics differ from training conditions, and their computational footprint may be undesirable for online or embedded monitoring, especially when robustness is required at very low SNR.

Despite this broad literature, systematic evaluation of denoiser-to-localization coupling under severe WGN and DSI conditions, including very low SNR, remains limited, particularly in controlled studies, where the localization backend is held fixed. Many reported improvements modify both the denoising stage and the localization logic, or change feature extraction and decision thresholds, which makes it difficult to attribute performance gains specifically to the denoiser. This confounding limit practical guidance for selecting a deployable denoising front end, especially when the design goal is not only noise suppression but also preservation of ToA-critical timing features that directly determine localization accuracy.

To address this gap, this study adopts a controlled benchmarking framework that varies only the denoising front end while keeping the downstream localization backend identical across all pipelines. This design removes backend-dependent influences such as different ToA pickers, correlation rules, or post-processing heuristics, so changes in localization accuracy, detection reliability, and runtime can be attributed directly to the denoiser. Within this fixed-backend setting, the proposed adaptive DWT-based Adaptive Denoising Optimization (ADO) [21] is evaluated against two learning-based benchmarks, a multilayer ANN and a lightweight feedforward neural network (FNN), using synthetic PD signals contaminated by WGN and DSI over SNRs from 9.78 dB to -10.34 dB. Performance is quantified using percentage localization error (PE), median absolute localization error (MedAE), F1 score, and execution time, enabling direct assessment of how denoising affects ToA-critical timing preservation and, consequently, localization fidelity under severe interference. The novelty is twofold: 1) establishing an explicit denoiser-to-localization coupling evaluation under a fixed localization backend, and 2) introducing an optimization-driven adaptive wavelet front end that systematically selects denoising configurations (wavelet basis, decomposition depth, and thresholding decisions) to maintain robustness as noise conditions and SNR vary. By reporting both accuracy and computational cost within the same controlled paradigm, the study provides practical guidance on robustness-efficiency trade-offs for real-time and resource-constrained MV cable monitoring.

II. Wavelet Denoising

Wavelet-domain denoising is a common front end for PD processing because the DWT provides a multiresolution representation that can attenuate broadband noise while separating narrowband or tonal interference from impulsive components. In practice, DWT denoising is highly sensitive to three coupled design choices: the mother wavelet (basis), the decomposition depth, and the threshold estimation and shrinkage rule. Parameter settings that work well under one measurement condition can degrade when SNR drops or when the interference spectrum changes, either by leaving residual noise that obscures weak impulses or by over-suppressing high-frequency coefficients and shifting the apparent onset that carries timing information [15]. Recent studies consistently report strong dependence on wavelet family, decomposition level, and threshold strategy, and show that the best configuration depends on the dominant noise type and measurement modality, so a single static setting is rarely robust across scenarios [16], [17].

To reduce this sensitivity, several adaptive or optimization-driven wavelet schemes have been explored in PD-related applications. Representative directions include sub-band identification or segmentation guided by impulsiveness indicators, such as spectral-kurtogram-guided approaches that emphasize informative bands and suppress noise-dominated coefficients [22], optimized wavelet denoising for recovering weak events in low-amplitude measurements [23], and adaptive parameter optimization in wavelet-threshold denoising, where optimization algorithms select threshold-related parameters based on the field noise level and a denoising quality objective [24]. Other work has emphasized robustness to translation effects and threshold estimation, such as shift-invariant wavelet denoising with empirical Bayes thresholding for low-SNR UHF PD wavefront detection and event segmentation [25]. While these approaches demonstrate the value of adaptivity, they often optimize denoising quality in isolation, or adapt only a subset of the wavelet design space, without explicitly constraining the solution to preserve ToA-critical timing cues that govern downstream localization fidelity [25].

Against this background, the novelty of ADO is an optimization-driven wavelet selection strategy that is explicitly designed for localization readiness. ADO performs optimization per signal realization, meaning each noisy PD waveform is processed independently, and it searches a predefined candidate set of mother wavelets, decomposition levels, and threshold configurations (threshold estimator and shrinkage rule) [21]. For each candidate configuration, the signal is denoised and evaluated using an objective function that balances suppression of WGN and DSI residuals with preservation of ToA-relevant structure by penalizing timing distortion of the earliest wavefront, for example, onset shift or correlation-lag deviation relative to the pre-denoising waveform. The final configuration is chosen using a minimum-cost rule, selecting the option that best satisfies this denoising-timing trade-off for the current noise realization. This bounded, discrete search avoids offline training, adapts directly to changing interference statistics, and improves robustness by discouraging over-denoising solutions that appear clean in energy terms, but distort the first-arrival features that localization depends on [15-17], [21].

III. PD Signals and Noise Modeling

Fig. 1 illustrates the online PD location system for power cables, which employs a multi-end measurement scheme to estimate the position of PD events. Three PD sensors, denoted A, B, and C, are installed along an underground cable at equal spacing of 2.5 km to capture the propagating PD pulse at each location. A substation receiver running the PD localization algorithm is connected to these sensors. When a PD event occurs, the pulse travels along the cable and is recorded as sigA,

sigB, and sigC, and the algorithm estimates the PD source location by comparing the signals' arrival times and waveform characteristics.

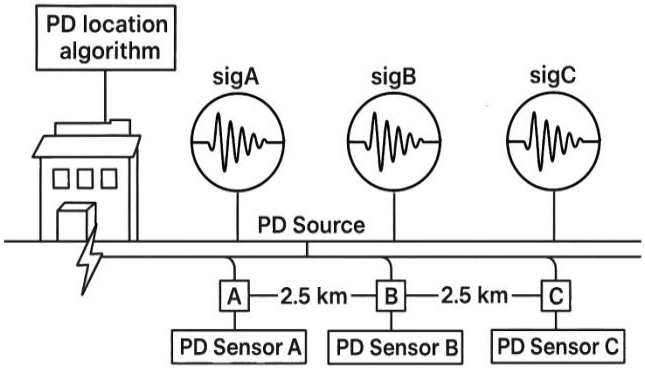

Fig. 1. Schematic representation of an online PD location estimation system for a power cable.

Fig. 2 presents the overall study workflow designed to isolate the effect of the denoising front end on PD localization. Noisy PD signals corrupted by WGN and DSI are first processed by one of the three denoisers (ADO, ANN, or FNN). The denoised outputs are subsequently fed into the same fixed PD localization backend to ensure a fair comparison, and performance is evaluated using localization accuracy, F1 score, and execution time to assess robustness and computational efficiency.

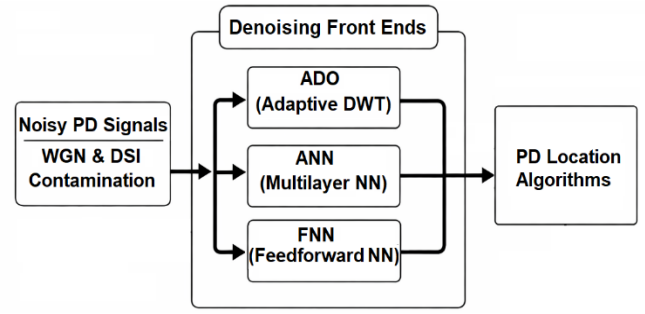

Fig. 2. Study workflow for benchmarking ADO against ANN and FNN under a fixed PD localization backend.

For simulation and localization analysis, the high-frequency PD pulse is modeled using Eq. (1) [8], which serves as the reference waveform:

$$s(t) = A[e^{-a_1 t}\cos(w_d t - \varphi) - e^{-a_2 t}\cos(\varphi)] \quad (1)$$

where, A is the amplitude coefficient, fixed at 0.01; a_1 and a_2 are damping factors set to 1 Ms^{-1} and 10 Ms^{-1}, respectively; $w_d = 2\pi f_d$ is the angular frequency of the damped oscillation; f_d is chosen as 1 MHz; and the phase term φ is given by $\tan^{-1}(w_d/a_2)$.

PD signal propagation along the cable is simulated with a sampling frequency f_s of 100 MHz. The propagation velocity, v_f, is obtained from Eq. (2) and then adjusted using laboratory experiments on a MV three-core cable (50 mm^2 Cu/XLPE/PVC, 8.7/15 kV), which yields an experimental value of 156 m/μs [8].

$$v_f = v_s/\sqrt{\varepsilon} \quad (2)$$

Here, v_s is the speed of electromagnetic waves in free space (300 m/μs); ε is the effective permittivity of the insulation and semiconductor layers of the cable.

To approximate realistic operating conditions, the simulated PD signals are corrupted by two typical interference sources: WGN and DSI. WGN is a broadband random noise with a Gaussian amplitude distribution and a flat power spectral density, representing thermal and electronic noise commonly present in measurement equipment. DSI is a narrowband periodic disturbance modeled as one or more sinusoidal components, capturing interference from power electronic devices, communication carriers, and other periodic emissions. WGN is generated in MATLAB, and its level is varied from 0 dB to -20 dB in -2 dB steps. DSI is applied at four discrete frequencies (600 kHz, 800 kHz, 1.5 MHz, and 5 MHz) with amplitudes ranging from 0.05 mV to 0.55 mV in 0.05 mV increments [8]. The DSI waveform is given by Eq. (3):

$$DSI(t) = A_{max}\sum_{t=1}^{N} 2\pi f_i t \quad (3)$$

The SNR of the noise-contaminated PD signals is then computed using Eq. (4) and Eq. (5), as in [8]:

$$SNR = \left(\frac{A_{signal}}{A_{noise}}\right)^2 \quad (4)$$

where, A_{signal} is the amplitude of the PD signal, and A_{noise} is the amplitude of the added noise. The corresponding SNR in decibels is:

$$SNR_{dB} = 10log_{10}(SNR) \quad (5)$$

IV. Denoising Methods

This study evaluates the proposed ADO denoising front end against two learning-based baselines, ANN and FNN, chosen to serve a clear benchmarking purpose rather than only practicality. ANN represents a higher-capacity neural denoiser that can model more complex nonlinear mappings, while FNN provides a lightweight alternative with reduced model complexity and different generalization behavior; together, they bracket a realistic range of neural denoising designs and allow us to test whether the localization performance limits arise from the learning paradigm itself or from model capacity and complexity. ADO is coupled to localization through its explicit goal of preserving ToA information: its adaptive DWT decomposition, wavelet selection, and thresholding are optimized to retain the earliest PD wavefront and sharp leading-edge transients while suppressing WGN and DSI. The denoised outputs are then passed unchanged into a fixed, identical ToA-based localization backend for all methods, so any change in estimated PD location is attributable to how well the denoiser preserves ToA-critical features rather than to differences in the localization algorithm.

A. Absolute Difference Optimization (ADO) Technique

Algorithm 1 presents the proposed ADO-based adaptive DWT denoising [21] procedure for PD signals, where the decomposition level is selected automatically to support

subsequent peak detection and ToA-based localization. A noisy observation is first formed by superimposing a synthetic clean PD waveform with mixed interference so that a clean reference remains available for objective evaluation. The noisy signal is constructed as Eq. (6):

$$x[n] = s_{PD}[n] + w_{WGN}[n] + d_{DSI}[n], \quad n = 1,\dots,N, \tag{6}$$

where, $s_{PD}[n]$ is the clean PD waveform, $w_{WGN}[n]$ denotes broadband WGN, and $d_{DSI}[n]$ represents DSI. This mixed noise model reflects practical environments, where both wideband background noise and narrowband harmonic or switching components coexist.

Algorithm 1: ADO-based adaptive DWT denoising for PD signals

```
Initialize
   Define wavelet type:
      wav ← 'db3'
   Define decomposition-level search range:
      Lmin ← 1
      Lmax ← 10
   Set tracking variables for optimal level:
      bestADO ← +∞
      bestLevel ← Lmin
      bestSignal ← empty

Compute
   Signal modelling (synthetic PD + noise):
      Generate clean PD waveform s_PD
      Generate white Gaussian noise WGN
      Generate discrete spectral interference DSI
      Form noisy signal:
         x ← s_PD + WGN + DSI
   Set initial decomposition level:
      L ← Lmin

While (L ≤ Lmax) do
   Apply DWT at level L:
      [cA_L, cD_L] ← DWT(x, wav, level = L)

   Denoising stage (reconstruct signal):
      y_L ← ReconstructDenoisedSignal(cA_L, cD_L)

   Peak detection (MATLAB findpeaks):
      peaks_L ← findpeaks(y_L)

   Compute absolute difference (AD):
      AD_L ← ComputeAbsoluteDifference(peaks_L,
referencePeaks)

   Calculate ADO factor:
      ADO_L ← ComputeADOFactor(AD_L)

   Check if current ADO is the minimum so far:
      if ADO_L < bestADO then
         bestADO ← ADO_L
         bestLevel ← L
         bestSignal ← y_L
      end

   Increment decomposition level:
      L ← L + 1
End

End

Output
   Denoised PD signal ŷ ← bestSignal at optimal decomposition
level bestLevel,
   which is subsequently used for PD peak detection and
localization.
```

For each candidate DWT decomposition level $L \in [1, 10]$, the noisy signal $x[n]$ is transformed using a fixed mother wavelet (Daubechies 3, db3), and the wavelet coefficients. The level-L DWT can be written as Eq. (7):

$$x[n] \xrightarrow{DWT} \{a_L|k|, d_j|k| \;\; j = 1,\dots,L\} \tag{7}$$

where, $a_L|k|$ are the approximation coefficients at level L and $d_j|k|$ are the detail coefficients at intermediate levels decomposition level j. After applying the chosen coefficient modification or thresholding strategy, the denoised signal $y_L[n]$ is reconstructed through the inverse DWT as Eq. (8):

$$y_L[n] = IDWT(a_L|k|, d_1|k|, \dots, d_L|k|) \tag{8}$$

The role of decomposition level L is to balance suppression and preservation: shallow decompositions may leave residual high-frequency noise, whereas excessively deep decompositions can over-smooth narrow PD impulses and distort the first arrival.

Each reconstructed signal $y_L[n]$, is then evaluated using peak features that are directly relevant to localization. PD peaks are extracted using MATLAB findpeaks, producing $\{p_{L,i}\}_{i=1}^{M}$., and compared against reference peaks $\{p_{ref,i}\}_{i=1}^{M}$ derived from the clean signal. The discrepancy is quantified by an absolute difference (AD) measure in Eq. (9).

$$AD_L = \sum_{i=1}^{M} |p_{L,i} - p_{ref,i}| \tag{9}$$

Absolute differences are preferred over squared errors because they remain in the original units (samples or amplitude), making them directly related to practical timing and localization deviations.

The AD measure is then condensed into an ADO factor, which serves as the scalar objective for decomposition level selection, as in Eq. (10):

$$ADO_L = |AD_L - K| \tag{10}$$

where, K = 0.505 is an empirically chosen reference that represents a balanced operating point between noise suppression and retention of PD peak structure. The optimal decomposition level is the one that minimizes ADO_L; lower values correspond to more faithful preservation of PD characteristics.

The algorithm iterates over all candidate levels $L = 1,\dots,L_{max}$, computing ADO_L for each. A running minimum is maintained as Eq. (11):

$$ADO_L = \underbrace{\min}_{1\le L\le L_{max}} ADO_L,$$
$$L^* = arg \underbrace{\min}_{L} ADO_L, \quad (11)$$

where, L^* denotes the optimal decomposition level. Whenever a new level yields $ADO_L < ADO_{min}$, the corresponding denoised signal $y_L[n]$ is stored as the current best candidate. Once all levels have been tested, the algorithm outputs Eq. (12):

$$\hat{y}[n] = y_{L^*}[n] \quad (12)$$

as the final denoised signal for subsequent PD peak detection and localization.

The ADO configuration is deliberately constrained to keep the method interpretable and computationally efficient, while still preserving ToA-critical content. The db3 wavelet is fixed because its compact support and pulse-like morphology match impulsive PD transients and help protect the sharp leading edge required by the fixed localization backend. The bounded search L = 1 to 10 provides sufficient multi-resolution flexibility to isolate PD-dominated sub-bands under mixed WGN and DSI without an excessive runtime increase, since cost scales mainly with record length and the number of tested levels. Finally, selecting $L*$ via the peak-based ADO objective links denoising to the fidelity of PD peak reconstruction, reducing over-smoothing and time shifting that would otherwise propagate into ToA error and degraded localization.

B. Artificial Neural Network (ANN) Technique

Algorithm 2 implements an ANN-based window-to-window denoising scheme for PD signals, where a fully connected multilayer perceptron is used as a data-driven front end to map noisy PD waveform segments to their clean counterparts on a frame-by-frame basis. The ANN is included as a representative learning based denoiser that can learn nonlinear signal noise relationships while remaining lighter than convolutional or recurrent models, which makes it suitable for long records and potential embedded deployment. The denoised output $\hat{y}[n]$is subsequently passed to the same fixed peak detection and ToA-based localization backend used throughout this study, so the impact of ANN denoising is assessed directly through its effect on timing and localization outcomes.

Algorithm 2: ANN-based window-to-window denoising for PD signals

```
Initialize
  Input noisy PD signal x, trained ANN denoiser net,
      window length win and hop size hop
  Set N ← length(x)
  Allocate reconstruction buffer yrec[1..N] ← 0
  Allocate overlap counter  wsum[1..N] ← 0
  Choose small constant ε to avoid division by zero

Compute
  Reshape x to a column vector
  Compute mean μx ← mean(x) and standard deviation σx ←
std(x) + ε
  Normalize input signal:
    for n = 1 to N do
      z[n] ← (x[n] − μx) / σx
    end
  Compute number of frames:
    nFrames ← 1 + floor((N − win) / hop)
  If nFrames < 1 then
    Set ŷ ← x and terminate the algorithm

While (frames remain to be processed) do
  For each frame index k = 1 to nFrames do
    Determine frame boundaries:
      a ← 1 + (k − 1) * hop
      b ← a + win − 1
      if b > N then b ← N
    Extract normalized frame:
      zk ← z[a : b]
    Apply ANN denoiser:
      ŷk ← net(zk)
    Accumulate output via overlap–add:
      yrec[a : b] ← yrec[a : b] + ŷk
      wsum[a : b] ← wsum[a : b] + 1
  end

  For n = 1 to N do
    if wsum[n] = 0 then wsum[n] ← 1
    Compute averaged normalized estimate:
      ẑ[n] ← yrec[n] / wsum[n]
    De-normalize to original scale:
      ŷ[n] ← ẑ[n] * σx + μx
  end

End

Output
  Denoised PD signal ŷ, which is subsequently used for peak
detection and PD localization. For multiple channels (A, B, C),
the same procedure is applied independently to each signal.
```

Let $[n]$, n=1, …, N, denote the noisy PD signal. As summarized in Algorithm 2, the method begins with global normalization to stabilize the input distribution. The sample mean and standard deviation are computed as Eq. (13) and Eq. (14):

$$\mu_x = \frac{1}{N}\sum_{n=1}^{N} x[n] \quad (13)$$

$$\sigma_x = \sqrt{\frac{1}{N-1}\sum_{n=1}^{N}(x[n]-\mu_x)^2}+\varepsilon \quad (14)$$

where, ε is a small constant to avoid division by zero. The normalized sequence is shown in Eq. (15):

$$z[n] = \frac{x[n]-\mu_x}{\sigma_x}+\varepsilon \quad (15)$$

This step reduces sensitivity to absolute amplitude variation between realizations and channels, which helps ANN training converge more reliably and improves generalization when PD magnitude changes across operating conditions.

The normalized signal $[n]$ is segmented into overlapping frames of length win with hop size hop. In this implementation, win = 512 samples and hop = 256 samples (50% overlap) were selected to capture the PD pulse and its local noise context while keeping the ANN input dimension and parameter count manageable. The overlap condition $hop < win$ reduces blocking artifacts and ensures that pulses near frame boundaries are fully represented in at least one frame, which improves reconstruction continuity and ToA stability. For each frame index k, the frame boundaries are defined by Eq. (16):

$$a_k = 1 + (k-1)hop, \quad b_k = min(a_k + win - 1, N), \tag{16}$$

and the corresponding frame is shown in Eq. (17).

$$z_k = [z[a_k], \ldots, z[a_k]]^T \tag{17}$$

Each frame z_k is processed by a trained multi-layer ANN implementing the mapping as Eq. (18):

$$\hat{y}_k = f_\theta(z_k) \tag{18}$$

where, $f_\theta(.)$ is a sequence of affine transformations and nonlinear activations parameterized by weights and biases θ. The network is trained offline using noisy–clean frame pairs by minimizing a mean-squared error (MSE) loss, as shown in Eq. (19):

$$\hat{y}_k = \frac{1}{K}\sum_{k=1}^{K} \left\| f_\theta(z_k^{noisy}) - y_k^{clean} \right\|_2^2, \tag{19}$$

which encourages faithful reconstruction of PD pulse shape and amplitude that drive reliable peak picking and ToA estimation in the subsequent localization stage.

The frame-wise outputs are finally recombined using an overlap-add to obtain a continuous denoised waveform. A reconstruction buffer $y_{rec}[n]$ and overlap counter $w_{sum}[n]$ are accumulated over all frames. The final normalized estimate is then computed as Eq. (20):

$$\hat{z}[n] = \frac{y_{rec}[n]}{max(w_{sum}[n], 1)} \tag{20}$$

and the denoised signal in the original scale is recovered via de-normalization in Eq. (21):

$$\hat{y}[n] = \hat{z}[n]\sigma_x + \mu_x \tag{21}$$

The same procedure is applied independently to each measurement channel, which avoids mixing channel-specific responses and helps maintain consistent inter-sensor timing behavior for localization.

For ANN training, the parameter settings were selected to balance model capacity, training stability, and coverage of realistic noise conditions. The ANN denoiser is a small fully connected network with two hidden layers (128 and 64 neurons), which is large enough to learn the nonlinear mapping from noisy frames z_k to denoised frames $\hat{y}_k$ in Algorithm 2, while keeping the model lightweight. Training uses the scaled conjugate gradient solver (trainscg) to obtain stable convergence for this regression task. To control overfitting and keep training time predictable, training is capped at 150 epochs with validation-based early stopping (max_fail = 6) using an 80/10/10 split for training, validation, and testing. Generalization is encouraged by generating 600 noisy–clean frame pairs across multiple SNR conditions (20, 10, 5, 0, and −5 dB) using AWGN with measured signal power, and by adding multi tone discrete spectral interference at 0.6, 0.8, 1.5, and 5 MHz with an amplitude of 0.20 mV so the network learns to handle both broadband and narrowband disturbances. Finally, per-shot z-score normalization during dataset generation reduces sensitivity to amplitude scaling, and a fixed random seed (rng(42,'twister')) ensures the training results are reproducible.

C. Feedforward Neural Network (FNN) Technique

Algorithm 3 summarizes the FNN-based window-to-window denoising procedure for PD signals. The FNN is included as a lightweight counterpart to the multi-layer ANN denoiser: it follows the same framing, normalization, and overlap-add reconstruction steps, and it uses the same windowing settings and noise-condition ranges as the ANN for a fair comparison. The key difference is that the FNN adopts a shallower fully connected architecture with fewer hidden units and parameters, which reduces execution time and memory demand while still learning the core nonlinear mapping from noisy frames to denoised PD frames. This design allows the study to isolate the effect of model capacity on denoising quality and, ultimately, on peak preservation and localization performance.

Algorithm 3: FNN-based window-to-window denoising for PD signals

```
Initialize
   Input noisy PD signal x, trained FNN denoiser net,
        window length win and hop size hop
   Set N ← length(x)
   Allocate reconstruction buffer yrec[1..N] ← 0
   Allocate overlap counter  wsum[1..N] ← 0
   Choose small constant ε to avoid division by zero

Compute
   Reshape x to a column vector
   Compute mean μx ← mean(x) and standard deviation σx ←
std(x) + ε
   Normalize input signal:
      for n = 1 to N do
         z[n] ← (x[n] − μx) / σx
      end
   Compute number of frames:
      nFrames ← 1 + floor((N − win) / hop)
   If nFrames < 1 then
      Set ŷ ← x and terminate the algorithm

While (frames remain to be processed) do
   For each frame index k = 1 to nFrames do
      Update
         Determine frame boundaries:
            a ← 1 + (k − 1) * hop
            b ← a + win − 1
```

```
        if b > N then b ← N
      Extract normalized frame:
        zk ← z[a : b]
      Apply FNN denoiser:
        ŷk ← net(zk)
      Accumulate output via overlap–add:
        yrec[a : b] ← yrec[a : b] + ŷk
        wsum[a : b] ← wsum[a : b] + 1

  Update and analyze
    For n = 1 to N do
      If wsum[n] = 0 then wsum[n] ← 1
      Compute averaged normalized estimate:
        ẑ[n] ← yrec[n] / wsum[n]
      De-normalize to original scale:
        ŷ[n] ← ẑ[n] * σx + μx
    end

    If (denoising is required for multiple PD channels) then
      Repeat the above procedure for each channel (A, B, C)
with its
      corresponding noisy input and trained FNN
    End if

End

Output
  Denoised PD signal ŷ for subsequent peak detection and
localization
```

Let $[n]$, n=1, …, N, denote the noisy PD signal. The FNN pipeline adopts exactly the same global normalization, framing, and overlap-add reconstruction strategy as the ANN method, using Eq. (13) to Eq. (17) for preprocessing and Eq. (20) to Eq. (21) for recombination and de-normalization. This ensures that both denoisers operate on identically scaled inputs and produce outputs in the same physical units, enabling a fair comparison.

For each frame z_k, the trained FNN denoiser that applies a relatively shallow nonlinear mapping as Eq. (22):

$$\hat{y}_k = g_{\emptyset}(z_k) \tag{22}$$

where, $g_{\emptyset}(.)$ denotes a fully connected network with a reduced number of layers and neurons compared with the ANN. A typical realization uses a single hidden layer, as shown in Eq. (23):

$$\begin{aligned} h &= \varphi(W_1 z_k + b_1), \\ \hat{y}_k &= W_2 h + b_2 \end{aligned} \tag{23}$$

where, $\varphi(\cdot)$ is a nonlinear activation function (e.g., ReLU), and $\phi = \{W_1, b_1, W_2, b_2\}$ is the parameter set. As with the ANN, the FNN is trained offline on noisy-clean frame pairs by minimizing the MSE loss in Eq. (19). However, because ϕ contains substantially fewer parameters than the ANN's θ, the FNN exhibits lower computational cost, reduced memory footprint, and a lower risk of overfitting. In the PD denoising context, this can be beneficial when the training dataset is limited or when the deployment environment requires fast inference and deterministic latency.

After frame-wise processing, the FNN outputs $\hat{y}_k$ are merged using the same overlap-add and de-normalization steps as the ANN formulation, yielding the final denoised signal $\hat{y}[n]$. For multi-channel PD measurements, the procedure is applied independently to each channel. Overall, the FNN-based scheme preserves the ANN's signal processing structure, while offering a leaner network architecture tailored to scenarios where reduced latency, energy consumption, and implementation complexity are prioritized.

V. Results and Discussion

The ADO, ANN, and FNN pipelines were evaluated in MATLAB on an Intel® Core™ i5-5200U CPU @ 2.2 GHz with 16 GB RAM using four complementary metrics: execution time, PE, MedAE, and F1-score. The simulations employed a multi-end measurement setup along a 2.5 km cable with a fixed true PD location, and the signals were corrupted by composite noise comprising WGN and DSI. This multi-end configuration reflects practical online monitoring architectures that estimate PD location from ToA differences between sensors, while WGN and DSI represent broadband stochastic noise and narrowband interference, respectively. For each noisy realization, the denoised outputs from each pipeline were passed to the same localization method, ensuring that performance differences could be attributed solely to the denoising front end.

Execution time was measured using MATLAB's tic/toc around the full denoising-localization chain, providing a direct estimate of computational cost under identical hardware and software conditions.

The PE quantifies the relative deviation of the estimated PD location from the true location and is defined as Eq. (24):

$$\begin{aligned} &\%\ Error \\ &= \frac{|\ PDloc_{true} - PDloc_{est}|}{PDloc_{true}} \times 100\% \end{aligned} \tag{24}$$

where, $PDloc_{true}$ is the actual PD position, and $PDloc_{est}$ is the estimated PD location. PE is used because it normalizes the error to the cable length, allowing fair comparison across different cable spans and giving utilities an intuitive indication of how far, in percentage terms, the estimate deviates from the true fault location when planning sectionalized repairs.

MedAE measures the typical magnitude of localization errors across Monte Carlo trials and is given by Eq. (25):

$$\begin{aligned} MedAE\ &= median\ (|P_i\ - A_i\ |), \\ i &=\ 1, 2, \ldots, N \end{aligned} \tag{25}$$

where, P_i and A_i are the estimated and true PD locations for the i-th trial, and N is the total number of realizations. MedAE is chosen instead of mean absolute error (MAE) because the median is far less sensitive to occasional extreme mis-localizations that may occur under very severe noise or rare algorithm failures. It therefore provides a robust indicator of the typical performance.

Finally, the F1-Score recasts localization into a tolerance-based detection problem. A trial is counted as a correct localization (positive) if the absolute error lies within a tolerance window of ±100 m. Binary labels are formed accordingly, and

the F1-Score is computed from precision (P) and recall (R) as Eq. (26) to Eq. (28):

$$F1\ Score = \frac{2 \times (Precision\ x\ Recall)}{Precision\ +\ Recall} \quad (26)$$

where,

$$= \frac{Precision}{True\ Positives} \quad (27)$$
$$True\ Positives\ +\ False\ Positives$$

$$Recall = \frac{True\ ositives}{True\ Positives\ +\ False\ Negatives} \quad (28)$$

Here, true positives correspond to trials where the estimated location falls within ±100 m of the true PD position, while false negatives are trials outside this tolerance. This tolerance-based F1-Score is used because, in field practice, operators are less concerned with centimeter-level accuracy than with confining the fault to a sufficiently short cable section that can be isolated and inspected. By combining precision and recall under an engineering-relevant tolerance, the F1-Score provides a practical, balanced measure of good localization, penalizing both missed acceptable localizations and large mis-localizations.

A. Execution Time

Fig. 3 compares the execution time of the ADO, ANN, and FNN denoising pipelines over SNR levels from 9.78 dB to -10.34 dB using the same dataset and computing setup. FNN is consistently the fastest method, with a mean runtime of about 619 s versus 729 s for ANN and 853 s for ADO, giving average savings of roughly 15% and 27%, respectively. FNN runtimes drop from around 680-720 s at high SNR to about 475-500 s at the lowest SNR, while ANN shows the widest variability (about 508-837 s for most SNRs) and a clear spike to roughly 1113 s at -2.38 dB. ADO remains the slowest, but is comparatively stable across SNR, ranging from about 751 s to 989 s. Overall, all pipelines are complete within approximately 475-1115 s, indicating suitability for offline or semi-real-time PD analysis.

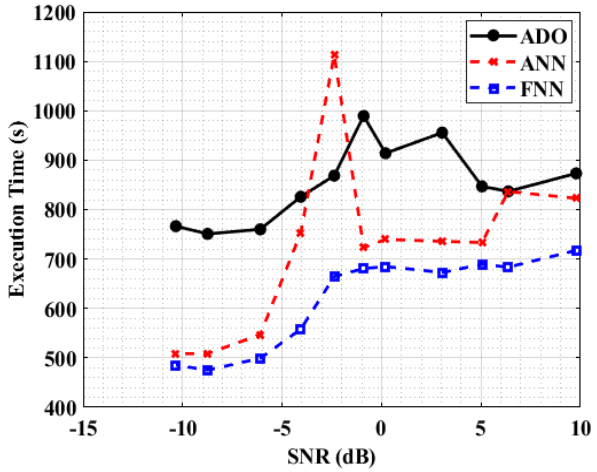

Fig. 3. Comparison of execution time for ADO-, ANN-, and FNN-based denoising pipelines over SNR levels ranging from 9.78 dB to −10.34 dB.

The runtime behavior reflects different computational cost drivers that matter for deployment. FNN's shallow architecture and small parameter count reduce per-epoch cost and typically converge quickly, which suits edge monitoring where latency, power, and throughput are dominant constraints, provided ToA fidelity is preserved. ANN offers intermediate speed but can be sensitive to the optimization landscape, leading to unpredictable delays that complicate buffering and scheduling. ADO is slower because it repeatedly performs multi-level DWT decompositions and reconstructions while searching across candidate configurations, but its latency is more predictable because it is mainly governed by signal length and the size of the search.

In practical PD localization systems, denoiser selection should therefore balance robustness, ToA preservation, computational budget, and operating constraints. FNN is appropriate for resource-limited, near-sensor screening when low latency is essential and uncertain windows can be flagged for later verification. ADO is better suited to offline diagnostics, periodic batch processing, or harsh EMI and low-SNR environments where timing fidelity and robustness outweigh runtime. A hybrid strategy is often most effective in utility monitoring: run FNN or ANN for continuous screening and invoke ADO selectively for low-confidence cases, low-SNR intervals, or high-value assets, improving reliability without paying ADO's full computational cost on every record.

For ADO, the dominant cost driver is search breadth (the number of tested decomposition levels and wavelet options), followed by signal length and windowing; restricting the search or processing a shorter pulse-centered window reduces runtime but can reduce robustness if the earliest wavefront is not retained. Threshold strength and selection-criterion tightness also trade off against timing fidelity, since overly aggressive settings risk smoothing or shifting first arrivals. For ANN and FNN, model size governs per-epoch cost, while training controls such as maximum epochs and early-stopping patience drive runtime variability; tightening these improves predictability but may degrade denoising if overly restrictive. Across all pipelines, the key design principle is to tune the front end to preserve the earliest PD wavefront and minimize ToA shift, since this ultimately governs end-to-end localization accuracy.

B. Percentage Error (PE)

Fig. 4 compares the PE of ADO, ANN, and FNN from 9.78 dB to -10.34 dB. ADO is consistently the most accurate, keeping PE between 0.0078% and 0.278% and staying below 0.03% for all $SNR \geq -4.06\ dB$. In contrast, ANN and FNN start at about 9.2% and 9.0% at 9.78 dB, remain above 4.7% under moderate noise, and rise sharply below -4 dB to roughly 21.4% and 22.5% at -10.34 dB. Consistent with this robustness, Table I shows that ADO selects lower decomposition levels (1–3) at higher SNR and shifts to deeper levels (4–5) as noise increases, keeping the estimated PD location close to 2000 m even in the most challenging cases.

These PE trends indicate that localization accuracy is dominated by timing fidelity rather than denoising strength alone. The large errors of ANN and FNN, even when SNR is high, imply that small reconstruction distortions can shift the effective ToA. In ToA-based MV cable localization, a one-to-two sample shift can translate into metres of location error, so smoothing or altering the first-arrival structure can degrade

localization despite producing visually clean signals. ADO's near-zero PE over moderate and high SNR, and only modest degradation at very low SNR, suggests that its wavelet-domain processing preserves ToA-critical features more reliably as interference changes.

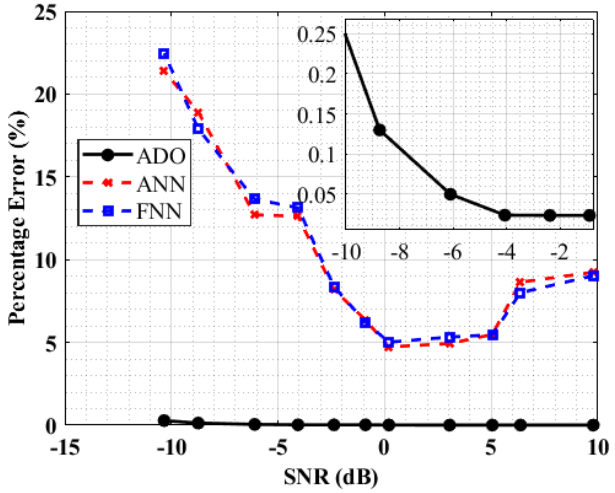

Fig. 4. Comparison of PE for ADO-, ANN-, and FNN-based denoising pipelines over SNR levels ranging from 9.78 dB to −10.34 dB.

TABLE I. ANALYSIS OF AD, ADO, AND PD LOCALIZATION ERROR ACROSS VARYING DWT DECOMPOSITION LEVELS AT A FIXED SNR OF -10.34 DB (WGN = -20 DB AND DSI = 0.55 MV)

SNR (dB)	AD	ADO	Estimated PD Location (m)	PD Location Error (%)	Decomp Level (DWT)
9.7812	0.5590	0.0540	1999.8439	0.0078	1
6.3698	0.4056	0.0056	1999.8439	0.0078	2
5.0484	0.3108	0.0892	1999.8439	0.0078	3
3.0471	0.3833	0.0167	1999.8439	0.0078	3
0.2003	0.4892	0.0892	1999.7401	0.0130	3
-0.9005	0.2566	0.1434	1999.5320	0.0234	4
-2.3772	0.3090	0.0910	1999.5320	0.0234	4
-4.0638	0.3597	0.0403	1999.5260	0.0237	4
-6.0967	0.4430	0.0430	2000.9960	0.0498	4
-8.7425	0.2855	0.1145	2002.6000	0.1300	5
-10.34	0.2900	0.1322	2005.5600	0.278	5

For system design, ADO is the safest default for field deployment when noise conditions are uncertain and accurate localization is required, since it maintains near-perfect accuracy down to -4.06 dB and remains below 0.3% PE at -10.34 dB. When computing is severely limited and approximate localization is acceptable, a lightweight neural front end such as FNN can be used for screening, with low-confidence or low-SNR windows escalated to ADO before producing the final location estimate. If ANN or FNN must serve as the primary front end, their evaluation and training should explicitly include timing-based criteria, because waveform similarity alone may not reflect ToA shifts that drive localization error. At low SNR, the deeper decomposition selected by ADO has a clear physical interpretation because finer subband separation helps preserve PD-dominated components, while suppressing broadband noise and interference.

For ADO, decomposition level is the dominant factor because it determines subband separation and ToA preservation; fixing the level (for example, always level 3) would likely increase PE at low SNR by reducing adaptability. Threshold strength and the AD/ADO selection criterion form a second lever, balancing residual interference suppression against the risk of attenuating the earliest wavefront, while the search breadth across wavelets and levels trades robustness against computation. For ANN/FNN, the loss function and training target are most influential, as smoothness-driven objectives such as MSE can blur transients and shift arrival time, whereas timing-aware losses or first-arrival weighting better protect localization-critical structure. Model capacity and training-test mismatch also affect performance, but the observed PE gap suggests that improving neural localization requires explicit timing-preserving design and training data that match expected EMI and SNR conditions.

C. Median Absolute Error (MedAE)

Fig. 5 compares the MedAE of ADO, ANN, and FNN across SNR levels from 9.78 dB to -10.34 dB. ADO performs best throughout, maintaining MedAE = 0.000 m for all $SNR \geq 0.2\ dB$ and increasing only to about 0.3-0.8 m when SNR becomes negative. In contrast, ANN and FNN produce much larger MedAE at every SNR point, starting around 65-117 m at high SNR and rising to roughly 280-800 m as noise increases. Their curves are nearly identical, indicating a shared limitation in maintaining precise timing under noisy conditions.

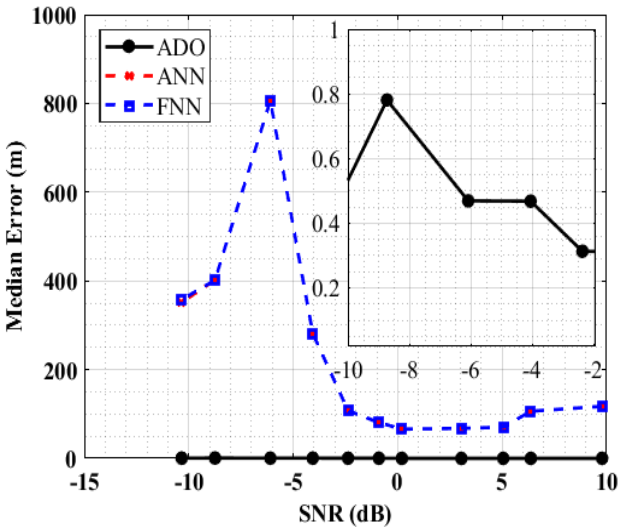

Fig. 5. Comparison of MedAE for ADO-, ANN-, and FNN-based denoising pipelines over SNR levels ranging from 9.78 dB to −10.34 dB.

MedAE is especially informative for PD localization because it represents typical field performance rather than being driven by occasional outliers. ADO's near-zero MedAE at moderate and high SNR implies that its denoising preserves the leading-edge timing needed for accurate time-difference-of-arrival estimation, so at least half of the events are localized essentially at the true position within the model's resolution.

Even at negative SNR, ADO remaining below 1 m indicates strong protection of ToA-critical features against time shifting as interference increases. By contrast, the large MedAE of ANN and FNN suggests a systematic failure mode, where smoothing and broadening of impulsive PD transients introduce timing bias and jitter that are amplified into large location errors by the ToA backend.

These findings translate into clear deployment guidance. For high-confidence localization that supports maintenance actions such as joint targeting and dig planning, ADO should be the primary denoising front end because it keeps typical errors near zero at non-negative SNR and within sub-meter levels even in noise-dominated regimes. ANN or FNN are only suitable when the goal is rapid event screening rather than precise localization, and should be paired with safeguards such as confidence scoring and escalation of difficult or low-SNR windows to an ADO pass before finalizing a location estimate. More broadly, ADO's stability across SNR makes it the safer default in MV underground networks, where noise conditions can vary significantly across assets and operating states.

For ADO, the decomposition level and its selection rule are most critical, since deeper decomposition at low SNR improves subband separation and helps preserve the first-arrival structure; fixing the level (e.g., always level 2 or 3) would likely increase MedAE under negative SNR. Threshold magnitude is the next key lever because over-thresholding can attenuate the earliest wavefront, while under-thresholding leaves timing-corrupting interference, and mother wavelet choice affects how well sharp edges are preserved. For ANN/FNN, the dominant factor is the training objective and loss weighting, as standard reconstruction losses promote smoothing and can shift arrival time; timing-aware losses that emphasize the first-arrival region are therefore the most direct route to reducing MedAE. Training-test mismatch, noise diversity, and model capacity also matter, but improving localization requires explicitly timing-preserving design rather than network size reduction alone.

D. F1-Score

Table II summarizes precision, recall, and F1-score for the ADO, ANN, and FNN denoising-detection pipelines from 9.78 dB to -10.34 dB. ADO achieves an F1-score of 1.0000 from 9.78 dB down to -4.06 dB with precision and recall both equal to 1.0000. Even under severe noise (-6.10 dB to -10.34 dB), ADO remains strong with F1-scores between 0.9787 and 0.9950, driven by precision = 1.0000 and only a modest recall drops to 0.9583-0.9900. In contrast, ANN and FNN show much lower F1-scores across all SNRs; although their precision stays at 1.0000, recall is limited (0.45-0.56 at higher SNR) and collapses at low SNR (0.03-0.08), reducing F1 to below 0.17 and as low as 0.0667 at -10.34 dB.

These detection trends directly affect PD localization because reliable localization requires consistent event capture, and missed detections reduce event coverage, create gaps in activity timelines, and decrease the likelihood of obtaining ToA-informative wavefront segments needed by the localization backend. In this sense, the F1-score is a gating metric for localization readiness. ADO's near-unity F1 across the full SNR range indicates that it preserves PD-noise separability under both WGN and DSI after denoising, while the small recall reduction at the lowest SNR reflects a controlled trade-off, where stronger suppression occasionally removes marginal PD components, but does not materially reduce event capture.

TABLE II. COMPARISON OF PRECISION, RECALL, AND F1-SCORE FOR ADO, ANN, AND FNN DENOISING–DETECTION PIPELINES OVER VARYING SNR CONDITIONS IN PD EVENT DETECTION

SNR (dB)	F1 score of ADO	F1 score of ANN	F1 score of FNN
9.7812	1.0000 (P = 1.0000, R = 1.0000)	0.6207 (P = 1.0000, R = 0.4500)	0.6207 (P = 1.0000, R = 0.4500)
6.3698	1.0000 (P = 1.0000, R = 1.0000)	0.6301 (P = 1.0000, R = 0.4600)	0.6301 (P = 1.0000, R = 0.4600)
5.0484	1.0000 (P = 1.0000, R = 1.0000)	0.7179 (P = 1.0000, R = 0.5600)	0.7179 (P = 1.0000, R = 0.5600)
3.0471	1.0000 (P = 1.0000, R = 1.0000)	0.7500 (P = 1.0000, R = 0.6000)	0.7500 (P = 1.0000, R = 0.6000)
0.2003	1.0000 (P = 1.0000, R = 1.0000)	0.7578 (P = 1.0000, R = 0.6100)	0.7578 (P = 1.0000, R = 0.6100)
-0.9005	1.0000 (P = 1.0000, R = 1.0000)	0.7226 (P = 1.0000, R = 0.5657)	0.7226 (P = 1.0000, R = 0.5657)
-2.3772	1.0000 (P = 1.0000, R = 1.0000)	0.6222 (P = 1.0000, R = 0.4516)	0.6222 (P = 1.0000, R = 0.4516)
-4.0638	1.0000 (P = 1.0000, R = 1.0000)	0.3960 (P = 1.0000, R = 0.2469)	0.3960 (P = 1.0000, R = 0.2469)
-6.0967	0.9950 (P = 1.0000, R = 0.9900)	0.1538 (P = 1.0000, R = 0.0833)	0.1538 (P = 1.0000, R = 0.0833)
-8.7425	0.9948 (P = 1.0000, R = 0.9896)	0.1639 (P = 1.0000, R = 0.0893)	0.1639 (P = 1.0000, R = 0.0893)
-10.3400	0.9787 (P = 1.0000, R = 0.9583)	0.0667 (P = 1.0000, R = 0.0345)	0.0667 (P = 1.0000, R = 0.0345)

From a deployment perspective, the perfect-precision but low-recall behavior of ANN and FNN is problematic as a primary trigger in many field settings. While detected events are almost always correct, many true PD events are missed even at high SNR, and recall collapses under noise, which can falsely suggest low PD activity and bias downstream localization toward only the strongest pulses. For condition monitoring, where missed PD events are unacceptable, ADO is therefore the preferred front-end because it maintains a near-balanced precision-recall profile across wide SNR variation. Where compute and power constraints motivate neural methods, ANN/FNN are better used as low-latency screeners in a two-stage workflow, with ADO applied selectively to buffered windows during suspected activity, low-SNR periods, or low-confidence cases to recover missed events before localization.

For ADO, threshold aggressiveness is the key lever at low SNR: tightening thresholds preserves precision but can reduce recall by suppressing weak PD components, so recall is typically the first metric to move under threshold scaling. Decomposition level selection and mother wavelet choice also

affect separability, because subband isolation and morphology matching determine whether PD transients remain distinct enough to cross the decision threshold under interference; fixing the level rather than adapting it would likely reduce recall at low SNR. For ANN/FNN, the dominant driver is the decision threshold and calibration, since precision = 1.0000 implies overly conservative triggering; lowering the threshold can raise recall, but must be paired with false-alarm control. Training objectives that penalize missed detections and broader training noise diversity across realistic WGN and DSI conditions are also critical, because the recall collapses at very low SNR indicates sensitivity to training-test mismatch.

VI. Conclusion and Recommendations

This study shows that denoising is a core element of MV cable PD localization because it directly governs timing integrity and, therefore, the reliability of ToA-based estimates. Using a controlled benchmark in which only the denoising front end is varied while the localization backend is kept fixed, the key message is that localization improves when the denoiser preserves the earliest PD wavefront features under interference, rather than simply reducing overall noise energy. From a design standpoint, a deterministic and training-free adaptive wavelet front end such as ADO is recommended when robust, high-confidence localization is required across changing noise conditions, because it adapts to the observed interference while discouraging timing distortion that would otherwise propagate into localization error. When computational speed and simplicity are the dominant constraints, a lightweight neural option such as the FNN can be considered as an efficiency-oriented alternative, but its robustness should be verified under the expected field noise variability instead of being assumed from laboratory-like conditions. These implications are especially relevant for low-cost monitoring hardware, since the evaluation was performed on a modest CPU platform representative of practical embedded units rather than GPU-accelerated workstations, where model handling and inference overheads can limit the deployability of learning-based pipelines. The main limitation is that the study relied on MATLAB simulations with synthetic PD waveforms in a controlled scenario and a noise model limited to WGN and DSI, so it cannot fully capture field variability such as joints, reflections, multiple concurrent sources, sensor coupling differences, and nonstationary interference. Future work should therefore validate the conclusions using measured PD data from real cable assets and defect types, and further optimize ADO for embedded implementation through streamlined search strategies and efficient realizations, with targeted acceleration where available.

Acknowledgment

This research was supported by the *i*-CATS Postgraduate Scholarship. The authors sincerely thank the High Voltage Laboratory and HiPER UMS for their invaluable technical assistance throughout this study. Generative AI was used exclusively for citation management and language refinement, with no intellectual contribution to this work's content, findings, or interpretations.

References

[1] B. Dhamala and M. Ghassemi, "Partial Discharge Localization along Medium Voltage Cables," in *Proc. 2025 IEEE International Electric Machines and Drives Conference (IEMDC)*, Houston, TX, USA, May 2025, pp. 834–839, doi: https://doi.org/10.1109/IEMDC60492.2025.11061155.

[2] L. Jiachuan, X. Wang, W. Zhou, J. Zhang, D. Dai, and G. Zhu, "A Comprehensive Review of Signal Processing and Machine Learning Technologies for UHF PD Detection and Diagnosis (I): Preprocessing and Localization Approaches," *IEEE Access*, vol. 9, pp. 69876–69904, 2021, doi: https://doi.org/10.1109/access.2021.3077483.

[3] A. Ragusa, H. G. Sasse, and A. Duffy, "On-Line Partial Discharge Localization in Power Cables Based on Electromagnetic Time Reversal Theory - Numerical Validation," *IEEE Transactions on Power Delivery*, vol. 37, no. 4, pp. 2911–2920, Aug. 2022, doi: https://doi.org/10.1109/TPWRD.2021.3119236.

[4] C.-K. Chang, H.-H. Chang, and B. K. Boyanapalli, "Application of Pulse Sequence Partial Discharge Based Convolutional Neural Network in Pattern Recognition for Underground Cable Joints," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 29, no. 3, pp. 1070–1078, Jun. 2022, doi: https://doi.org/10.1109/tdei.2022.3168328.

[5] Z. Zhang et al., "Partial Discharge Pattern Recognition Based on a Multifrequency F–P Sensing Array, AOK Time–Frequency Representation, and Deep Learning," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 29, no. 5, pp. 1701–1710, Aug. 2022, doi: https://doi.org/10.1109/tdei.2022.3199189.

[6] S. Mantach, M. Partyka, V. Pevtsov, A. Ashraf, and B. Kordi, "Unsupervised Deep Learning for Detecting Number of Partial Discharge Sources in Stator Bars," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 30, no. 6, pp. 2887–2895, Dec. 2023, doi: https://doi.org/10.1109/tdei.2023.3306324.

[7] M. R. Hussain, S. S. Refaat, and H. Abu-Rub, "Overview and Partial Discharge Analysis of Power Transformers: A Literature Review," *IEEE Access*, vol. 9, pp. 64587–64605, 2021, doi: https://doi.org/10.1109/access.2021.3075288.

[8] C. K. Fern et al., "Fault localization on power cables using time delay estimation of partial discharge signals," *International Journal of Electrical and Computer Engineering*, vol. 13, no. 6, p. 6000, Sep. 2023, doi: https://doi.org/10.11591/ijece.v13i6.pp6000-6015.

[9] M. Alqtish et al., "A review of partial discharge electrical localization techniques in power cables: Practical approaches and circuit models," *Energies*, vol. 18, no. 10, Art. no. 2583, 2025, doi: https://doi.org/10.3390/en18102583.

[10] G. V. R. Xavier, R. De a Coelho, H. S. Silva, A. J. R. Serres, E. G. Da Costa, and A. S. R. Oliveira, "Partial discharge location through application of stationary discrete Wavelet transform on UHF signals," *IEEE Sensors Journal*, vol. 21, no. 21, pp. 24644–24652, Sep. 2021, doi: https://doi.org/10.1109/jsen.2021.3116491.

[11] S. Govindarajan, M. Natarajan, J. A. Ardila-Rey, and S. Venkatraman, "Partial discharge location identification using permutation entropy based instantaneous energy features," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, Jan. 2021, doi: https://doi.org/10.1109/tim.2021.3121477.

[12] Y. Yan, R. Trinchero, I. S. Stievano, H. Li, and Y.-Z. Xie, "An Automatic Tool for Partial Discharge De-Noising via Short-Time Fourier Transform and Matrix Factorization," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–12, Jan. 2022, doi: https://doi.org/10.1109/tim.2022.3216583.

[13] L. Lu, K. Zhou, G. Zhu, X. Yang, and B. Chen, "Partial Discharge Location Algorithm Based on Total Least-Squares With Matérn Kernel in Cable Systems," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 3, pp. 2421–2431, Mar. 2023, doi: https://doi.org/10.1109/tii.2022.3153835.

[14] J. Wang, L. Lu, G. Zhu, K.-L. Yin, K. Zhou, and B. Chen, "Unbiased Euclidean Direction Search Algorithm for Partial Discharge Location in Cable Systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 74, pp. 1–13, Jan. 2025, doi: https://doi.org/10.1109/tim.2025.3538076.

[15] M. A. Shams, H. I. Anis, and M. El-Shahat, "Denoising of Heavily Contaminated Partial Discharge Signals in High-Voltage Cables Using Maximal Overlap Discrete Wavelet Transform," *Energies*, vol. 14, no. 20, p. 6540, Oct. 2021, doi: https://doi.org/10.3390/en14206540.

[16] K. Friebe and F. Jenau, "Evaluation Study on Wavelet Denoising of Antenna-Based PD Measurements in Strong Interference Environments Considering a New Reliability Score of Pulse Detection," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 31, no. 6, pp. 2887–2896, Dec. 2024, doi: https://doi.org/10.1109/tdei.2024.3374243.

[17] M. A. Sahnoune et al., "Noise Source Effect on the Quality of Mother Wavelet Selection for Partial Discharge Denoising," *IEEE Access*, vol. 12, pp. 132729–132743, Jan. 2024, doi: https://doi.org/10.1109/access.2024.3459474.

[18] J. Yeo et al., "Localisation of Partial Discharge in Power Cables Through Multi-Output Convolutional Recurrent Neural Network and Feature Extraction," *IEEE Transactions on Power Delivery*, vol. 38, no. 1, pp. 177–188, Jun. 2022, doi: https://doi.org/10.1109/tpwrd.2022.3183588.

[19] C. Kumar, B. Ganguly, D. Dey, and S. Chatterjee, "Wavelet-Based Convolutional Neural Network for Denoising Partial Discharge Signals Extracted via Acoustic Emission Sensors," *IEEE Sensors Letters*, vol. 8, no. 7, pp. 1–4, Jul. 2024, doi: https://doi.org/10.1109/lsens.2024.3414954.

[20] A. H. Alshalawi and F. S. Al-Ismail, "Partial Discharge Detection Based on Ultrasound Using Optimized Deep Learning Approach," *IEEE Access*, vol. 12, pp. 5151–5162, Jan. 2024, doi: https://doi.org/10.1109/access.2024.3350555.

[21] C. K. Fern et al., "Adaptive Wavelet De-noising Algorithm using Absolute Difference Optimization Technique for Partial Discharge Signal," *Journal of Engineering and Science Research*, vol. 7, no. 3, pp. 26–31, Jun. 2023, doi: https://doi.org/10.26666/rmp.jesr.2023.3.4.

[22] J. Zhong, X. Bi, Q. Shu, D. Zhang, and X. Li, "An Improved Wavelet Spectrum Segmentation Algorithm Based on Spectral Kurtogram for Denoising Partial Discharge Signals," IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1–8, Jan. 2021, doi: https://doi.org/10.1109/tim.2021.3071224.

[23] Q. Lin, F. Lyu, S. Yu, H. Xiao, and X. Li, "Optimized denoising method for weak acoustic emission signal in partial discharge detection," IEEE Transactions on Dielectrics and Electrical Insulation, vol. 29, no. 4, pp. 1409–1416, Jun. 2022, doi: https://doi.org/10.1109/tdei.2022.3183662.

[24] X.-C. Hua et al., "A Novel Adaptive Parameter Optimization Method for Denoising Partial Discharge Ultrasonic Signals," IEEE Transactions on Dielectrics and Electrical Insulation, vol. 30, no. 6, pp. 2734–2743, Nov. 2023, doi: https://doi.org/10.1109/tdei.2023.3331663.

[25] Francisco, Rodrigo, George, P. D. Alvim, S. Júnior, and H. S. Silva, "Wavefront Detection and Event Segmentation Method for Partial Discharge Signal Analysis," IEEE Access, vol. 13, pp. 111602–111613, Jan. 2025, doi: https://doi.org/10.1109/access.2025.3582635.