AIMS Mathematics, 11(3): 7235–7263.
DOI: 10.3934/math.2026298
Received: 01 January 2026
Revised: 05 March 2026
Accepted: 12 March 2026
Published: 19 March 2026

https://www.aimspress.com/journal/Math

Research article

Numerical and deep learning methods for diffusion model with fractional Laplacian operator and its application to signal denoising

Ishtiaq Ali* **and Muneerah AL Nuwairan***

Department of Mathematics and Statistics, College of Science, King Faisal University, P. O. Box 400, Al-Ahsa, 31982, Saudi Arabia

* **Correspondence:** Email: iamirzada@kfu.edu.sa (I.A); msalnuwairan@kfu.edu.sa (M.N).

Abstract: In modern signal processing applications, denoising signals remain an essential task, as it is difficult to reduce noise without compromising essential structural information. While standard Laplacian operators are limited in their capacity to handle long-range interactions and maintain fine-scale features, classical diffusion models offer a sound mathematical foundation for signal smoothing. To overcome these limitations, in this study, we developed a novel numerical and deep learning approach driven by a nonlocal fractional Laplacian operator of the form $(-\Delta)^s$ $(0 < s < 1)$, which captures long-range interactions within signals. To accurately approximate the fractional Laplacian dynamics, a classical second-order finite difference (FD) discretization of the Laplacian was constructed in space, and the fractional operator was defined through its spectral matrix power. The resulting semi-discrete system was integrated in time using an unconditionally stable Crank-Nicolson (CN) scheme. In addition to this model-based method, a convolutional neural network (CNN) architecture was employed as a refinement step, trained to learn the nonlinear mapping from noisy to clean signals. To evaluate the denoising capability and effectiveness of the proposed methods, different input signals contaminated with additive white Gaussian noise at prescribed signal-to-noise (SNR) ratios were considered. The SNR and root mean square error (RMSE) were used as quantitative performance measures. The results showed that the FD-CN and the CNN methods perform well in conjunction. The numerical scheme smooths the data in a numerically controlled way, and the CNN captures complex local structures and adapts to non-Gaussian noise patterns. In addition, the proposed framework was extended to multichannel real-world ECG signals, demonstrating its robustness in handling correlated noise across multiple leads. The multichannel experiments confirmed that the FD-CN and CNN-based approaches remain effective under realistic, real-data conditions.

Keywords: nonlocal diffusion model; fractional Laplacian; convolutional neural network; pseudospectral analysis; signal denoising
Mathematics Subject Classification: 35R11, 65M06, 68T07

1. Introduction

Fractional partial differential equations (PDEs) have recently attracted significant research interest due to their capability to represent phenomena with memory effects that cannot be captured by normal PDEs. Among other things, fractional calculus has demonstrated itself to be a practical instrument in the fields of material science [1], finance [2], image processing [3], porous medium flow [4], and bioengineering applications, including anomalous diffusion in biological tissues [5], among other applications. The fractional Laplacian plays a central role in fractional calculus. In their seminal work, Caffarelli and Silvestre [6] proved that this operator can be realized as a Dirichlet-to-Neumann map associated with a degenerate elliptic equation posed on the half-space $\mathbb{R}^{n+1}_+ := \mathbb{R}^n \times (0,1)$. The degeneracy arises from a weight function which, depending on the fractional exponent, either vanishes or blows up on the boundary $\partial\mathbb{R}^{n+1}_+ = \mathbb{R}^n$. This extension technique was later adapted to bounded domains $\Omega \subset \mathbb{R}^n$ in [7, 8], leading to a degenerate elliptic boundary-value problem formulated on the infinite cylinder $C := \Omega \times (0,1)$. Since the associated weight belongs to the Muckenhoupt A_2 class, the analysis naturally fits within the framework of weighted Sobolev spaces, and the classical elliptic theory extends with only minor modifications.

Data transformation is a key component of signal processing, as it facilitates the analysis, optimization, and correction of signals such as scientific data, seismic activity, stock market movements, photographs, and video data. In predictive and decision science applications, such as seismic monitoring networks in areas prone to seismic activity, signal processing is a useful tool. Signals are inevitably prone to disruptions from both natural and man-made sources, which can happen both during data collection and transmission. The term "noise" in signal processing describes unneeded changes made to a source signal during the stages of collection, storage, and transmission. Therefore, before further processing the information in the data, it is essential to eliminate the noise in the signals (denoise). This important step makes the signal more reliable as data by getting rid of unnecessary information. It also makes sure that its complicated patterns and inherent qualities are easy to understand. Numerous signal processing techniques have been proposed in the literature, including Fourier analysis [9, 10], wavelet transform [11, 12], variational mode decomposition [13, 14], and empirical mode decomposition [15]. When analyzing transitory and non-stationary signals, like ECGs, the wavelet transform has proven useful. In contrast to Fourier analysis, wavelet analysis breaks a signal into scaled and decaying oscillatory components. Alfaouri and Daqrouq employed the wavelet transform method with thresholding in order to get rid of extra or small wavelet coefficients [16]. The result eliminated a variety of errors in single-dimensional ECG signals.

Dixit and Sharma [17] examined different approaches for implementing wavelet thresholding to remove noise from two-dimensional data. Variational mode decomposition (VMD) is an excellent means to look at signals that are not stationary and have a number of components. This is because it does not depend on fixed frequency bands or pre-defined basis functions, as Fourier and wavelet transformations do. VMD is very flexible so it can record both low- and high-frequency sounds. This can be useful for processing audio, analyzing biomedical signals, and building communication systems. VMD is very useful when other ways of breaking down signals don't work because it can handle signals that change their spectral components over time [18]. The empirical mode decomposition (EMD) method separates a signal into a small number of intrinsic modal functions, but it does this by filtering the signal [19]. The EMD method has some problems, which is why better versions like the

ensemble empirical mode decomposition (EEMD) method have been made [20]. Principal component analysis (PCA) [21] and singular value decomposition (SVD), and its many variations [22, 23] are two other ways to process signals. Intrinsic modes, basis functions, and orthogonal components are the building blocks of these methods. The Fourier and wavelet analysis methods, for instance, show signals as linear combinations of basis functions. However, in wavelets, the frequencies can change. VMD and EMD look at intrinsic modal functions, while SVD and PCA use linear algebra to break signals down into orthogonal parts so that the most important parts of the signals stand out. On the other hand, the diffusion equation shows how signals change over time and gives us a way to process signals that is always changing. Many fields of science and engineering that work with signals use the diffusion equation, which is a parabolic partial differential equation. For example, it can be used to model the behavior of processes driven by signals [24], image processing tasks where images are seen as two-dimensional signals [25, 26], and finance, where the diffusion equation is used to predict how the market will behave [27].

Many important contributions have led to a strong mathematical basis for fractional Laplacian operators. Silvestre [28] was one of the first people to find a regularity result for the fractional obstacle problem. This work set the stage for the current study of nonlocal operators. Thanks to Caffarelli and Silvestre's work, it is now possible to write fractional Laplacians as local degenerate elliptic problems [29]. This made it much easier for both theoretical and numerical work to get done. Kwaśnicki [30] improved the field by giving ten different ways to write the fractional Laplace operator. This made it possible to fully understand its analytical features. Du et al. [31] progressed the study of nonlocal diffusion models by analyzing well-posedness and volume-constrained approximation methods. These are very important for computer simulations of things that happen outside of a certain area. Ros-Oton and Serra [32] made important contributions to the Dirichlet problem about the fractional Laplacian, especially in terms of how it behaves near the edge. This is very important for using nonlocal PDEs in real life. Recent studies have looked into better numerical and fractional-order methods for modeling systems that are driven by diffusion and for processing signals. Owolabi and Pindza used Riesz fractional derivatives to model diffusion processes that do not happen in a normal way. This is a good way to show when patterns show up in unusual ways [33]. In image processing, Jin et al. [34] showed a fast and effective numerical method for segmenting and denoising images that greatly improved the speed and accuracy of the calculations. Chen et al. [35] made a new way to get arterial pulses that can be used to reliably test methods for denoising multichannel signals. This made it possible to assess their work fairly and accurately. Lanza et al. [36] made a model that used a symmetrized fractional variation framework with L_1 precision to get rid of noise in signals. They used the Grünwald–Letnikov method to break it up into smaller pieces. This gets rid of the noise and keeps the important parts of the signal. These works together set up the theoretical framework for modern research on fractional diffusion, nonlocal operators, and how to get close to their numbers.

CNNs are now better than models that use mathematical fractional Laplacians to get rid of noise. They are good at cleaning up signals that are too noisy because they can change shape and identify patterns. Fractional Laplacian operators smooth out and make data that is not local more frequent while keeping the edges sharp. CNNs learn hierarchical representations straight from the training data to avoid these issues. They can use what they learn about various types of signals and levels of noise to help them figure out what is noise and what is important. Recent research has shown that deep convolutional architectures can improve noisy vibration measurements, distributed acoustic sensing

seismic data, and ECG signals. These methods are much better than older filtering or PDE-based methods at getting the right results. By combining CNNs with fractional Laplacian operators, we get a hybrid framework. This framework brings together the stability and interpretability of nonlocal diffusion with the flexibility and ability to learn deep modern networks.

This paper develops a complete discrete numerical model of the one-dimensional fractional diffusion problem. We first break the spatial operator down into smaller parts using a standard finite-difference Laplacian. We then use spectral matrix-power mapping to get the fractional diffusion operator. The CN method is used to move the semi-discrete system forward in time. This sets up a time-stepping method that is second-order and unconditionally stable for second-order dynamics. The proposed framework's efficacy in denoising is assessed using synthetic one-dimensional signals influenced by additive white Gaussian noise. The fully discrete fractional diffusion solution is first applied to these noisy inputs to see how well it works at reducing noise and finding the underlying clean structure. The primary contribution of this study is the integration of the FD–CN discretization of the fractional Laplacian with a supplementary CNN-based residual denoising module, accompanied by a comprehensive stability analysis of the entirely discrete operator. The ε-pseudospectra of the discrete fractional diffusion matrix are utilized to assess stability and resilience. This gives us information about how well the scheme works and how reliable the numbers are, showing that the suggested hybrid numerical-deep learning framework for signal denoising works well.

The rest of the paper is organized as follows. Section 2 describes the mathematical model, the numerical method, and a deep learning technique based on CNNs. Section 3 consists of results and discussion, while Section 4 concludes the summary of the main findings.

2. Description of the mathematical model and numerical scheme

In this study, we consider the following diffusion equation with a noise-removing property to numerically approximate the evolution of a one-dimensional signal governed by a fractional diffusion process. We consider the initial-boundary value problem

$$\frac{\partial \mathcal{S}}{\partial \mathrm{t}}(\mathrm{x},\mathrm{t}) = \kappa(-\Delta)^{s}\mathcal{S}(\mathrm{x},\mathrm{t}), \qquad \mathrm{x} \in (-\mathcal{L},\mathcal{L}),\ \mathrm{t} > 0, \tag{2.1}$$

with homogeneous Dirichlet boundary conditions

$$\mathcal{S}(-\mathcal{L},\mathrm{t}) = \mathcal{S}(\mathcal{L},\mathrm{t}) = 0, \tag{2.2}$$

and initial data,

$$\mathcal{S}(\mathrm{x},0) = \mathcal{S}_0(\mathrm{x}), \tag{2.3}$$

where $\kappa > 0$ is the diffusion coefficient and $0 < s \leq 1$ denotes the fractional order of the operator. When $s = 1$, the model reduces to the classical heat equation, whereas $0 < s < 1$ produces a nonlocal fractional smoothing effect. In the conventional case $s = 1$, the function $\mathcal{S}(\mathrm{x},\mathrm{t})$ represents the evolving signal, $\mathcal{S}_0(\mathrm{x})$ is the given noisy input, and the parameter κ controls the rate of diffusion. The spatial operator $\Delta = \frac{\partial^2}{\partial \mathrm{x}^2}$ is a second-order derivative that measures how quickly the signal changes with respect to space and how curved it is. This curvature-driven mechanism controls how information moves through signal regions with steep gradients or high-frequency oscillations. These regions diffuse more quickly, which smooths them out, while larger structural features stay mostly the same. The first

noisy signal $\mathcal{S}_0(\mathrm{x})$ is spread out over time ($\mathrm{t} > 0$) in a way that lets noise move from areas where it is very concentrated to areas nearby. The magnitude of the diffusion coefficient κ determines how quickly this smoothing takes place. Solving the diffusion equation with the given initial and boundary conditions thus provides a natural means of attenuating noise: as t increases, the solution $\mathcal{S}(\mathrm{x}, \mathrm{t})$ becomes progressively smoother, yielding a cleaner representation of the underlying signal. To solve the model Eq (2.1), subject to given initial and boundary conditions, the spatial discretization is first constructed using a classical second-order FD approximation of the Laplacian. The fractional diffusion operator is then defined through a spectral fractional power of the resulting discrete Laplacian matrix.

Consider a uniform grid with $\mathcal{N}$ interior nodes on $(-\mathcal{L}, \mathcal{L})$,

$$\mathrm{x}_i = -\mathcal{L} + i\,\Delta\mathrm{x}, \qquad i = 1, 2, \ldots, \mathcal{N}, \qquad \Delta\mathrm{x} = \frac{2\mathcal{L}}{\mathcal{N}+1},$$

and enforce the boundary conditions by setting $\mathcal{S}(\mathrm{x}_0, \sqcup) = \mathcal{S}(\mathrm{x}_{\mathcal{N}+1}, \mathrm{T}) = 0$.

Let the vector of nodal values be

$$\mathbb{S}(\mathrm{t}) = \begin{bmatrix} \mathcal{S}(\mathrm{x}_1, \mathrm{t}) \\ \mathcal{S}(\mathrm{x}_2, \mathrm{t}) \\ \vdots \\ \mathcal{S}(\mathrm{x}_{\mathcal{N}}, \mathrm{t}) \end{bmatrix} \in \mathcal{R}^{\mathcal{N}}.$$

The second derivative at interior points is approximated by the centered FD formula,

$$\frac{\partial^2 \mathcal{S}}{\partial \mathrm{x}^2}(\mathrm{x}_i) \approx \frac{\mathcal{S}(\mathrm{x}_{i-1}) - 2\mathcal{S}(\mathrm{x}_i) + \mathcal{S}(\mathrm{x}_{i+1})}{(\Delta\mathrm{x})^2}, \qquad i = 1, \ldots, \mathcal{N},$$

leading to the standard discrete Laplacian of the form:

$$\mathcal{A} = \frac{\kappa}{(\Delta x)^2} \begin{bmatrix} -2 & 1 & & & 0 \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ 0 & & & 1 & -2 \end{bmatrix} \in \mathcal{R}^{\mathcal{N}\times\mathcal{N}}, \tag{2.4}$$

where the matrix $\mathcal{A}$ is real, symmetric, and strictly negative definite.

2.1. Spectral construction of the discrete fractional Laplacian

Since $\mathcal{A}$ is symmetric, it admits the eigen decomposition of the form:

$$\mathcal{A} = V\Lambda V^{\top}, \tag{2.5}$$

with $V \in \mathcal{R}^{\mathcal{N}\times\mathcal{N}}$ orthogonal, and

$$\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_{\mathcal{N}}), \qquad \lambda_j < 0.$$

The eigenpairs satisfy

$$\mathcal{A}\,v_j = \lambda_j v_j.$$

To discretise the fractional Laplacian, we apply the fractional spectral mapping $\lambda_j \mapsto -(-\lambda_j)^s$, defining

$$\Lambda_s = \mathrm{diag}(\mu_1, \ldots, \mu_N), \qquad \mu_j := -(-\lambda_j)^s,$$

and the discrete fractional diffusion operator

$$L_s := V \Lambda_s V^\top. \tag{2.6}$$

The matrix L_s is symmetric negative definite and provides a consistent finite-dimensional approximation of $\kappa\, \Delta^s$. Using this operator, the spatially semi-discrete form of (2.1) becomes

$$\frac{d\mathbf{S}}{d\mathrm{t}}(\mathrm{t}) = L_s\, \mathbf{S}(\mathrm{t}), \qquad \mathrm{t} > 0. \tag{2.7}$$

2.1.1. Crank-Nicolson time discretization scheme

Let $t_k = k\Delta t,\ k = 0, \ldots, K$, with uniform time step $\Delta \mathrm{t} = T/K$, and denote

$$\mathbf{S}^k \approx \mathbf{S}(t_k).$$

Applying the CN method to (2.7), we approximate:

$$\frac{d\mathbf{S}}{dt} \approx \frac{\mathbf{S}^{k+1} - \mathbf{S}^k}{\Delta t} = L_s \left(\frac{\mathbf{S}^{k+1} + \mathbf{S}^k}{2} \right),$$

leading to the fully discrete system

$$\left(I - \frac{\Delta t}{2} L_s \right) \mathbf{S}^{k+1} = \left(I + \frac{\Delta t}{2} L_s \right) \mathbf{S}^k, \qquad k = 0, \ldots, K-1. \tag{2.8}$$

Since L_s is symmetric and negative definite, the CN scheme is unconditionally stable and preserves the dissipative character of the fractional diffusion operator. Given the initial vector,

$$\mathbf{S}^0 = \begin{bmatrix} \mathcal{S}_0(\mathrm{x}_1) \\ \mathcal{S}_0(\mathrm{x}_2) \\ \vdots \\ \mathcal{S}_0(\mathrm{x}_N) \end{bmatrix},$$

the recurrence (2.8) provides the complete numerical solution. In practice, the matrix

$$M := I - \frac{\Delta t}{2} L_s$$

is factorized once, and each time step is computed via

$$\mathbf{S}^{k+1} = M^{-1} \left(I + \frac{\Delta t}{2} L_s \right) \mathbf{S}^k.$$

The FD-CN scheme (2.8) thus yields a robust and accurate fully discrete approximation of the one-dimensional fractional diffusion equation (2.1), and forms the basis of the numerical computations presented in this work.

2.2. Time-fractional diffusion model with memory effects

To incorporate temporal memory effects into the diffusion process, we extend model (2.1) by replacing the classical first-order time derivative with a Caputo fractional derivative of order $0 < \alpha < 1$. The resulting time-fractional diffusion model is given by

$$ {}^{C}D_t^{\alpha}\mathbf{S}(t) = L_s\,\mathbf{S}(t), \qquad t > 0, \tag{2.9}$$

subject to the same initial condition

$$\mathbf{S}(0) = \mathbf{S}^0.$$

Here ${}^{C}D_t^{\alpha}$ denotes the Caputo fractional derivative defined by

$$ {}^{C}D_t^{\alpha}\mathbf{S}(t) = \frac{1}{\Gamma(1-\alpha)} \int_0^t (t-\tau)^{-\alpha} \frac{d\mathbf{S}(\tau)}{d\tau}\, d\tau,$$

where $\Gamma(\cdot)$ is the gamma function. When $\alpha = 1$, the model reduces to the classical semi-discrete system (2.7).

2.2.1. L1 time discretization scheme

Let $t_n = n\Delta t$, $n = 0, 1, \ldots, K$, and denote

$$\mathbf{S}^n \approx \mathbf{S}(t_n).$$

The Caputo derivative is approximated using the classical L1 scheme,

$$ {}^{C}D_t^{\alpha}\mathbf{S}(t_n) \approx \frac{1}{\Delta t^{\alpha}\Gamma(2-\alpha)} \sum_{k=0}^{n-1} a_k \left(\mathbf{S}^{n-k} - \mathbf{S}^{n-k-1}\right),$$

where the weights are defined by

$$a_k = (k+1)^{1-\alpha} - k^{1-\alpha}.$$

Substituting this approximation into (2.9) yields the fully discrete system

$$\left(\frac{1}{\Delta t^{\alpha}\Gamma(2-\alpha)} I - L_s\right)\mathbf{S}^n = \frac{1}{\Delta t^{\alpha}\Gamma(2-\alpha)} \sum_{k=0}^{n-1} a_k \mathbf{S}^{n-1-k}, \qquad n \geq 1. \tag{2.10}$$

The matrix

$$M_{\alpha} := \frac{1}{\Delta t^{\alpha}\Gamma(2-\alpha)} I - L_s$$

is symmetric positive definite since L_s is symmetric negative definite. Therefore, the linear system (2.10) is uniquely solvable at each time step. The fractional order α controls the strength of temporal memory. The values of $\alpha < 1$ introduce long-range memory effects, resulting in slower diffusion dynamics compared to the classical case. In the context of signal denoising, this provides additional flexibility in balancing smoothing and feature preservation.

2.3. Deep learning method for the model equation (2.1)

Along with the FD method and the CN scheme for solving the fractional diffusion model (2.1), we employ a data-driven convolutional neural network (CNN) for signal denoising. We also use a data-driven CNN to clean up signals. The CNN acts as a nonlinear filter that learns how to change a noisy input signal into a clean or regularized output. It works like the fractional diffusion operator, which smooths out noise and sometimes improves it. The formulation is based on recent progress in using fully convolutional architectures to clean up electrocardiogram (ECG) data [37]. CNNs are well known for how well they can fix images, but their basic structure, which combines hierarchical feature extraction with local convolutional filtering, also works well for getting rid of noise in signals. Because of this, more and more research is being done on how CNNs can be used to improve time-series data in areas like biomedical engineering, geophysics, and audio processing.

Deep learning has recently become a useful way to remove noise from signals. CNNs are great in figuring out the difference between noise and important structure because they can quickly learn spatial and temporal patterns from data. One big benefit of deep learning is that it does not depend on any ideas you already have about the noise model or the way you get the data. This flexibility makes it good for datasets that are not all the same, where the noise statistics change depending on the measurement, device, or conditions used to collect the data. Biological signal processing made a lot of progress in the early stages, but significant early advancements occurred in biological signal processing, particularly in the enhancement of ECG signal clarity. Recent advancements in deep learning have significantly enhanced the capacity to eliminate noise from biological and geophysical signals. Qiu et al. [38] demonstrated a two-stage deep convolutional architecture for ECG applications that effectively eliminates noise while preserving the integrity of the features' shape. Chiang et al. [39] employed fully convolutional denoising autoencoders to eliminate noise in high-quality ECG readings. Dong et al. [40] introduced a deep-learning approach specifically designed for multiarea surface seismic data, while Zhu et al. [41] created a deep neural network framework for the denoising and decomposition of seismic signals. Pan et al. [42] introduced a bidirectional denoising convolutional autoencoder model for the purification of noisy vibration signals acquired by wavefront sensors, while Li et al. [43] employed self-supervised and transfer learning techniques to refine distributed acoustic sensing data without prior knowledge of its content. When taken as a whole, these studies demonstrate the adaptability and potent denoising capabilities of contemporary deep learning architectures across a variety of signal types.

Numerous studies demonstrate that CNNs are not only useful for image restoration but also offer a strong and efficient framework for recovering clean signals in a range of scientific and engineering applications. Their capacity to learn nonlinear, context-dependent denoising operators makes them particularly attractive in situations where conventional PDE-based or transform-based approaches tend to smooth significant structures. This supports the current work's use of a residual 1D CNN, whose efficacy can be directly compared to that of diffusion and fractional Laplacian filter techniques.

We design a special one-dimensional CNN denoiser that learns the nonlinear mapping between noisy observations and their clean counterparts. This architecture integrates effectively with the FD solver for the fractional diffusion model. The CNN takes a noisy signal segment of length $N = 512$ sample bits long and processes it through a symmetric encoder-decoder structure with skip connections that are made by upsampling layers. A one-dimensional convolution (Conv1D) comes first in each convolutional block, and then a nonlinear activation comes next. The encoder uses max-

pooling layers to gradually compress the representation, while the decoder uses UpSampling1D and Conv1DTranspose layers to bring the resolution back to what it was before. This hierarchical feature extraction enables the model to perceive both the local oscillations and the global smoothness patterns generated by fractional diffusion. Figure 1 shows how the CNN is put together in detail. The CNN takes a noisy signal segment $\mathbf{S}_{\text{noisy}} \in \mathbb{R}^{512}$ (512 samples, one channel) and runs it through a symmetric encoder-decoder structure with upsampling layers that let you skip connections. A ReLU activation comes after a 1D convolution (Conv1D) in each convolutional block. The encoder uses max-pooling to slowly make the representation smaller, and the decoder uses UpSampling1D and Conv1DTranspose layers to bring the original resolution back. Let $\mathcal{S}_{\text{clean}}(x)$ be a reference clean signal defined on $(-\mathcal{L}, \mathcal{L})$ and sampled on the uniform grid $\{x_i\}_{i=1}^{512}$ that the numerical solver uses. We create a noisy realization by

$$\mathcal{S}_{\text{noisy}}(x_i) = \mathcal{S}_{\text{clean}}(x_i) + \eta_i, \qquad i = 1, \ldots, 512,$$

where η_i is additive white Gaussian noise at a prescribed SNR (between 3 and 7 dB during training). Each training sample is a pair

$$(\mathbf{S}_{\text{noisy}}, \mathbf{S}_{\text{target}}) \in \mathbb{R}^{512},$$

where $\mathbf{S}_{\text{target}}$ corresponds either to the clean signal or to the fractional diffusion solution at the final time. As opposed to using outputs from the fractional diffusion solver as targets, the CNN is trained on clean reference signals. This lets the network learn the true structure of the signal without getting the smoothing bias that comes from diffusion. The network is a fully convolutional autoencoder that preserves the grid structure and avoids dense layers. All convolutions use kernel size 7, the same padding, and increasing filter counts. Table 1 lists every layer with its output shape.

Table 1. Layer-by-layer specification of the CNN architecture.

Layer	Type	Kernel/Stride	Filters	Output shape
Encoder				
Conv1	Conv1D + ReLU	7×1, stride 1	32	$(512, 32)$
Pool3	MaxPool1D	size 2, stride 2	–	$(256, 32)$
Conv2	Conv1D + ReLU	7×1, stride 1	64	$(256, 64)$
Pool2	MaxPool1D	size 2, stride 2	–	$(128, 64)$
Conv3	Conv1D + ReLU	7×1, stride 1	128	$(128, 128)$
Pool1	MaxPool1D	size 2, stride 2	–	$(64, 128)$
Bottleneck				
Bottleneck	Conv1D + ReLU	7×1, stride 1	256	$(64, 256)$
Decoder				
TConv1	Conv1DTranspose + ReLU	7×1, stride 2	128	$(128, 128)$
Up1	UpSampling1D	factor 2	–	$(256, 128)$
TConv2	Conv1DTranspose + ReLU	7×1, stride 1	64	$(256, 64)$
Up3	UpSampling1D	factor 2	–	$(512, 64)$
TConv3	Conv1DTranspose + ReLU	7×1, stride 1	32	$(512, 32)$
Output				
Output	Conv1D (linear)	7×1, stride 1	1	$(512, 1)$

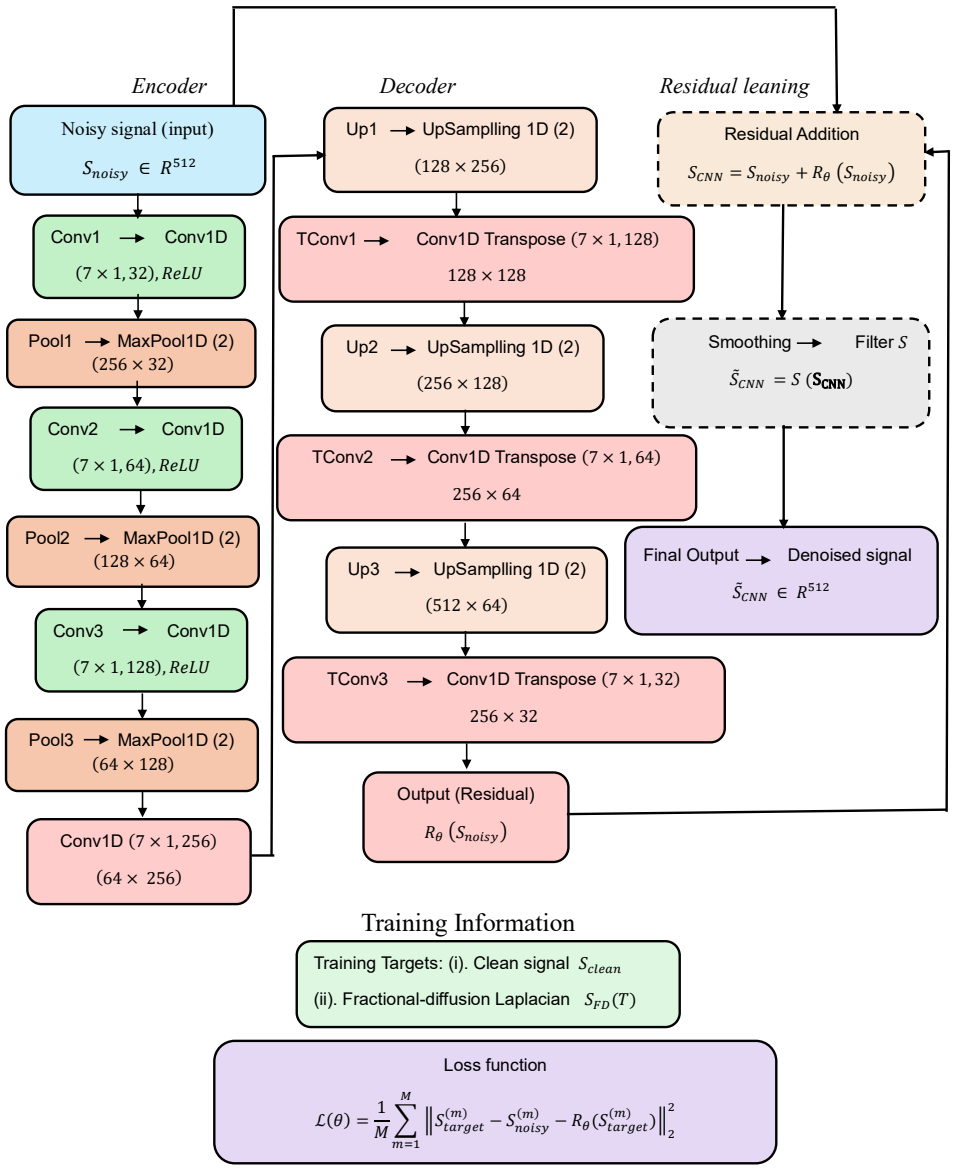

Figure 1. Convolutional neural network architecture.

The encoder compresses the spatial dimension from 512 to 64 while increasing the channel count, and the bottleneck expands the channel dimension to 256 to capture nonlocal interactions. The decoder symmetrically restores the length to 512 while reducing the number of channels. The final linear Conv1D produces the predicted residual

$$\mathbf{R}_\theta(\mathbf{S}_{\text{noisy}}).$$

The denoised signal is obtained through residual addition.

$$\mathbf{S}_{\text{CNN}} = \mathbf{S}_{\text{noisy}} + \mathbf{R}_\theta(\mathbf{S}_{\text{noisy}}),$$

which accelerates convergence and improves accuracy by allowing the network to focus on learning the correction rather than the entire signal mapping. This formulation closely mirrors the incremental smoothing effect of the fractional diffusion equation.

The network parameters θ are optimized by minimizing the mean-squared error loss

$$\mathcal{L}(\theta) = \frac{1}{M}\sum_{m=1}^{M}\left\|\mathbf{S}_{\text{target}}^{(m)} - \mathbf{S}_{\text{noisy}}^{(m)} - \mathbf{R}_\theta(\mathbf{S}_{\text{noisy}}^{(m)})\right\|_2^2, \tag{2.11}$$

which explicitly enforces the residual formulation. The Adam optimizer ensures fast and stable convergence without overfitting. Training data are generated by contaminating a parametric family of Gaussian peaks with additive white Gaussian noise at SNR levels between 3 and 7 dB. Once trained, the CNN reproduces the qualitative features of nonlocal diffusion smoothing at negligible computational cost. It thus provides an effective surrogate for the fractional diffusion solver, reducing the computational burden by several orders of magnitude while maintaining high denoising quality.

2.4. Pseudospectral behavior of the discrete fractional diffusion Laplacian operator

In this section, we investigate the pseudospectral properties of the discrete fractional diffusion operator obtained after semi-discretizing the one-dimensional model equation of the form:

$$\partial_t \mathcal{S} = -(-\Delta)^s, \qquad x \in (-L, L), \quad 0 < s \le 1,$$

using a second-order centered finite difference scheme with homogeneous Dirichlet boundary conditions. Let $(-\mathcal{L}, \mathcal{L})$ be partitioned into $\mathcal{N}_{\text{int}}$ interior nodes with uniform spacing:

$$h = \frac{2\mathcal{N}}{\mathcal{N}_{\text{int}} + 1}.$$

The resulting discrete Laplacian $\mathcal{D}_{xx} \in \mathbb{R}^{\mathcal{N}_{\text{int}} \times \mathcal{N}_{\text{int}}}$ is given by

$$\mathcal{D}_{xx} = \frac{1}{h^2}\begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix},$$

which is real symmetric and strictly negative definite. Let $\{(\lambda_j, v_j)\}_{j=1}^{N_{\text{int}}}$ denote its eigenpairs, where

$$\mathcal{D}_{xx} v_j = \lambda_j v_j, \qquad \lambda_j < 0.$$

Collecting the eigenvectors in the orthogonal matrix $V = [v_1, \ldots, v_{N_{\text{int}}}]$, the discrete fractional diffusion generator is defined spectrally by

$$B := -(-\mathcal{D}_{xx})^s = V\,\mathrm{diag}(\mu_1, \ldots, \mu_{N_{\text{int}}})\,V^\top, \qquad \mu_j = -(-\lambda_j)^s < 0.$$

Although $\mathcal{D}_{xx}$ is normal, the nonlinear spectral transformation $\lambda \mapsto -(-\lambda)^s$ generally destroys normality under perturbations. Consequently, the spectral behavior of B cannot be deduced solely from its eigenvalues, and one must instead study its ε–pseudospectra. For a matrix B, the ε–pseudospectrum is defined by

$$\Lambda_\varepsilon(B) = \left\{z \in \mathbb{C} : \|(zI - B)^{-1}\| \ge \varepsilon^{-1}\right\} = \bigcup_{\|\chi\| \le \varepsilon} \sigma(B + \chi),$$

and is approximated numerically via Trefethen's poor–man's method [44]. In particular, for each prescribed perturbation level $\varepsilon \in \{10^{-2}, 10^{-4}\}$, as shown in Figures 2 and 3, we generate random Hermitian perturbations χ_p satisfying $\|\chi_p\| \approx \varepsilon$ and compute the eigenvalues of the perturbed generators

$$B_p = -\big(-\mathcal{D}_{\mathrm{xx}} + \chi_p)\big)^s, \qquad p = 1, \ldots, P.$$

The sets $\sigma(B_p)$ form a cloud of sample points approximating $\Lambda_\varepsilon(B)$. To explore the effect of spatial resolution, we parametrize the number of interior grid points as

$$\mathcal{N}_{\mathrm{int}} = Q\,\mathrm{r}, \qquad Q, \mathrm{r} \in \{4, 8, 16\},$$

mirroring the hierarchical structure used in overlapping spectral-element constructions. To interpret the pseudospectra in the context of time discretization, we overlay the stability boundary of the Crank-Nicolson method. The absolute stability region of CN is the left half-plane,

$$\Re(\omega) \le 0, \qquad \omega = \Delta t\,\lambda,$$

so unconditional stability is ensured provided that the spectrum (and pseudospectrum) lies entirely in $\Re(z) < 0$. The qualitative outcome of the numerically calculated ε-pseudospectra is as follows. First, according to the dissipative nature of the continuous fractional Laplacian, the unperturbed eigenvalues strictly lie on the negative real axis and drift further to the left as the spatial resolution increases. Second, the pseudospectral cloud expands but stays completely contained in the left half-plane for the larger perturbation $\varepsilon = 10^{-2}$, suggesting moderate non-normal sensitivity brought on by the fractional spectral mapping. Third, all pseudospectral approximations satisfy $\Re(z) < 0$ for both $\varepsilon \in \{10^{-2}, 10^{-4}\}$, confirming the CN method's unconditional stability for the fully discrete fractional diffusion operator. Increasing Q or r results in tighter spectral localization and reduced pseudospectral spreading, indicating enhanced accuracy of the finite difference approximation.

The results indicate that the discrete operator $-(-\mathcal{D}_{\mathrm{xx}})^s$ preserves the fundamental dissipative characteristics of the continuous fractional diffusion generator, providing a solid basis for the CN time-stepping scheme used in the numerical experiments.

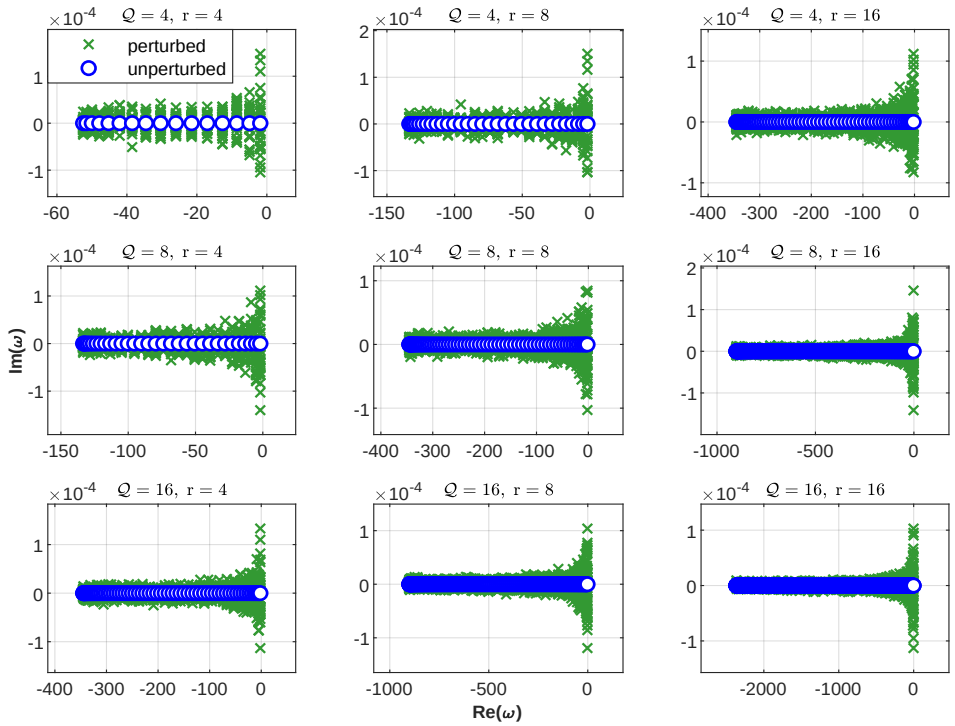

Figure 2. Pseudospectra of the discretization of the diffusion fractional Laplacian operator with perturbation $\epsilon = 10^{-2}$, $s = 0.7$.

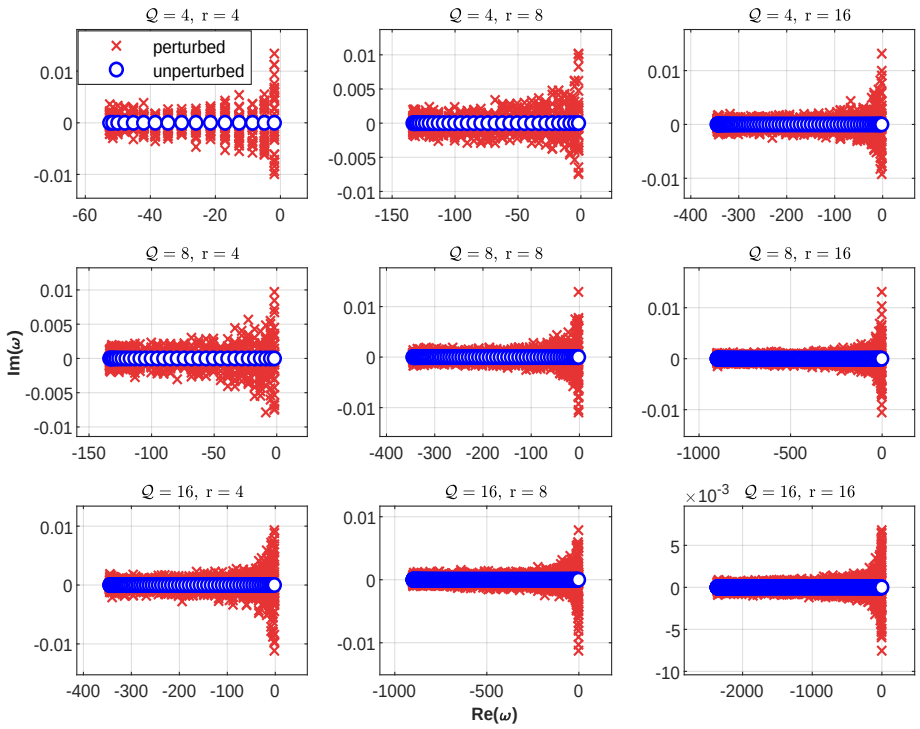

Figure 3. Pseudospectra of the discretization of the diffusion fractional Laplacian operator with perturbation $\epsilon = 10^{-4}$, $s = 0.7$.

3. Results and discussion

In this section, we perform some numerical experiments to confirm the efficiency of the two denoising strategies. The methods are utilized on the six prototype one-dimensional signals given in Table 2, each corrupted with a well-known harmonic function and spectral profile, such as Gaussian, Lorentzian, Cauchy-Lorentz, and Voigt peaks. These profiles, which are defined by precise analytical expressions, are necessary for spectroscopy and are often used to model physical phenomena in many scientific fields. They work well with high-order numerical schemes like the FD-CN methods because they are naturally smooth. This makes them great ways to see how well denoising works. So, being able to successfully recover these signals after they have been contaminated is both important for science and useful in the real world. Here are the analytical forms of the test profiles:

- **Gaussian profile:**

$$\mathcal{G}_j(\mathrm{x};\mu_j,\sigma_j,A_j) = A_j \exp\left[-\frac{(\mathrm{x}-\mu_j)^2}{2\sigma_j^2}\right].$$

- **Lorentzian profile:**

$$L_j(x;\mu_j,\sigma_j,A_j) = \frac{A_j}{1+\left(\frac{\mathrm{x}-\mu_j}{\sigma_j}\right)^2}.$$

  In both Gaussian and Lorentzian profiles, A_j denotes the peak amplitude, μ_j the peak location, and σ_j the width parameter.
- **Cauchy–Lorentz peak:**

$$C_j(\mathrm{x};\mu_j,\gamma_j) = \frac{1}{\pi}\left[\frac{\gamma_j}{(\mathrm{x}-\mu_j)^2+\gamma_j^2}\right],$$

  where γ_j is the half-width at half-maximum.
- **Voigt profile:**

$$\mathcal{V}(x;\sigma,\gamma) = \int_{-\infty}^{\infty} \mathcal{G}(\tau;\sigma)\, C(\mathrm{x}-\tau;\gamma)\, d\tau = \frac{\Re[\omega(z)]}{\sigma\sqrt{2\pi}}, \quad z = \frac{\mathrm{x}+i\gamma}{\sigma\sqrt{2}},$$

  where the constituent functions are given by

$$\mathcal{G}(\mathrm{x};\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\mathrm{x}^2/(2\sigma^2)}, \qquad C(\mathrm{x};\gamma) = \frac{\gamma}{\pi(\mathrm{x}^2+\gamma^2)}, \qquad \omega(z) = e^{-z^2}\mathrm{erfc}(-iz).$$

We accomplish a set of numerical experiments with the six benchmark signals given in Table 2. Each profile has additive white Gaussian noise added to it in order to make it look similar to real measurement noise. These datasets serve as a rigorous framework to evaluate the denoising efficacy and stability of the numerical methods.

Table 2. Clean signal used for numerical experimentation, noised by additive white Gaussian.

fTest	Clean signal description
Signal 1	$\mathcal{G}_1(\mathrm{x}; 5, 0.7, 1) + \mathcal{G}_2(\mathrm{x}; 0, 0.8, 2) + \mathcal{G}_3(\mathrm{x}; -5, 0.4, 0.5)$
Signal 2	$L_1(\mathrm{x}; 5, 0.7, 1) + L_2(\mathrm{x}; 0, 0.8, 2) + L_3(\mathrm{x}; -5, 0.4, 0.5)$
Signal 3	$C_1(x; -0.4, 0.5) + C_2(\mathrm{x}; 0.4, 0.5)$
Signal 4	$\mathcal{V}(\mathrm{x}; 1, 0.5)$
Signal 5	$\cos(4\pi\mathrm{x}) + 0.5\cos(2\pi\mathrm{x}) + 0.25\cos(5\pi\mathrm{x})$
Signal 6	$\mathrm{S}_6(\mathrm{x}) = 3\mathrm{x} + \cos(10\pi\mathrm{x}) + \begin{cases} \cos(5\pi\mathrm{x}), & 0 \le \mathrm{x} < 0.5 \\ 0, & 0.5 \le \mathrm{x} \le 1 \end{cases}$

To see how well the proposed denoising framework works, different quantitative metrics can be used to see how close the filtered signal is to the original clean signal. In this work, we focus on two commonly used metrics: The SNR and the RMSE. These metrics offer additional perspectives on the efficacy of numerical and learning-based methods in reducing noise while maintaining critical signal characteristics. The SNR is a logarithmic measure of the ratio between the energy of the real signal and the noise that is left over after filtering. This tells us how clear the reconstructed signal is and is given by:

$$\mathrm{SNR} = 10\log_{10}\left(\frac{\sum_{i=0}^{\mathcal{N}}(\mathcal{G}_e(\mathrm{x}_i))^2}{\sum_{i=0}^{\mathcal{N}}(\mathcal{G}_e(\mathrm{x}_i) - \mathcal{S}(\mathrm{x}_i, T))^2}\right). \tag{3.1}$$

Here $\mathcal{G}_e(\mathrm{x}_i)$ give us the exact clean signal at certain points, and $\mathcal{S}((\mathrm{x}_i, T)$ provides the denoised signal at time $t = T$, which is the last level. A higher SNR value means that the denoising worked better, which means that the processed signal has more of the original structure than the noise. The RMSE tells us how different the filtered and real signals are on average. It is characterized by

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathcal{N}}\sum_{i=0}^{\mathcal{N}}(\mathcal{G}_e(\mathrm{x}_i) - \mathcal{S}(\mathrm{x}_i, T))^2}. \tag{3.2}$$

A smaller RMSE means that the reconstruction is more accurate and has less distortion. When used together, SNR and RMSE give a strong and complementary measure of denoising quality, which makes it possible to objectively compare numerical schemes and check the reliability of the suggested fractional diffusion–CNN approach.

In all of our simulations, we used $N = 256$ grid points, with $\kappa = 0.5$ for all signals. The strength and nonlocality of the diffusion process were controlled by the fractional order s. As s gets closer to 1, the model gets closer to the classical heat equation. This makes smoothing better, which helps get rid of noise, but it might hide sharp features. The diffusion effect is weaker but more nonlocal when s is smaller. This helps keep the peaks, but it could cause some noise behind. In these simulations,

we chose s by finding the best balance between improving the SNR and lowering the RMSE. Our tests show that moderate values, like $s \approx 0.7$, are a good balance between denoising performance and structural accuracy. For this reason, we chose $s = 0.7$ in our simulations a final diffusion time $T = 0.3$, and a time step $\Delta t = 2 \times 10^{-3}$. The CNN denoiser learns by predicting the difference between the clean and noisy signals, $\mathcal{S}_{\text{clean}} - \mathcal{S}_{\text{noisy}}$. The final output of the CNN is the predicted difference added to the noisy input and a short moving-average smoothing filter. We demonstrated the denoising results for Signals 1–6 using both the FD method and the CNN method for the model equation (2.1). Each signal had a lot of extra noise added to it at first (initial SNR $\approx$ 5 dB), which made it a tough test for the denoising algorithms. The SNR and RMSE metrics show how much better the quality of the signal is and how closely the reconstructed signals match the real signals. Both methods improve the SNR over the noisy input, but the CNN method always gives the best SNR for each signal. The RMSE values, which were about 10^{-2}, show that most of the important information in the signal is still there after filtering. Both methods raised the SNR for Signal 1 from 4.98 dB (noisy) to 12.27 dB (FD) and 17.30 dB (CNN). The CNN result had an SNR that was about 5 dB higher and an RMSE that was almost 44% lower than FD's (0.08108 vs. 0.14471). This means that it is better at getting rid of noise.

Signal 2 had a similar pattern, where SNR went up from 4.98 dB to 12.83 dB (FD) and 17.16 dB (CNN). CNN again had about 4.3 dB more SNR. The RMSE of the CNN output (0.08053) was much lower than that of FD (0.13255), which means that the output was more accurate. The CNN method worked even better in Signal 3. The FD method raised the SNR to 13.32 dB, while the CNN method raised it to 22.06 dB, which is almost 9 dB more. The CNN reconstruction for Signal 3 had an RMSE of 1.78×10^{-2}, which was about one-third of the FD error, which was 4.88×10^{-2}. This huge improvement means that the CNN method is especially good at removing noise from signals that have more complicated features or higher-frequency content. For Signal 4, which was the hardest case, the FD method did not really improve the SNR (it went from 5.28 dB to 4.99 dB, which was basically the same), but the CNN method did improve the SNR to 12.56 dB. The FD's output for Signal 4 had an RMSE of 7.24×10^{-2}, which was much higher than CNN's 3.03×10^{-2}. The substantial variation in Signal 4 shows that the learning-based CNN is capable of handling specific noise patterns or signal features that the fixed FD diffusion process has trouble with. In the end, the CNN method is the best way to get rid of noise in all four signals because it always has the highest SNR and the lowest RMSE. This means that it gets rid of noise better and keeps the real signal features better. The FD method still worked well to get rid of noise in Signals 1–3, which shows that the diffusion model works well for signals that are just a small amount out of shape. However, it failed to perform as well as the CNN, especially if problems get really hard, like with Signal 4. The difference in performance is mostly due to the fractional diffusion operator's linear smoothing nature. It slowly takes away parts with high frequencies, which can make sharp shapes look unreal. The CNN, on the other hand, uses residual learning to perform nonlinear, feature-adaptive filtering, which helps keep localized peaks and irregular features better.

Due to their complicated content, Signals 5 and 6 are hard test cases: Signal 5 is made up of several sinusoids, and Signal 6 has a linear trend with high-frequency oscillations and a piecewise discontinuity at $x = 0.5$. Both signals were denoised by strong noise (initial SNR $\approx$ 5 dB), which made the task of removing the noise more difficult. Both methods made the SNR much better than the noisy baseline, almost doubling it for Signals 5 and 6. In every case, the CNN method always had the best post-denoising SNR and the lowest RMSE. With FD, the SNR for Signal 5 went from 5.1160 dB (noisy) to

10.2759 dB, and with CNN, it went up even more to 12.0012 dB. The RMSE went down from 0.0779 for FD to 0.0570 for CNN-sm. Signal 6 showed a similar pattern; the SNR went from 5.0220 dB to 10.8324 dB (FD) and 12.1411 dB (CNN), while the RMSE went from 0.0680 (FD) to 0.0526 (CNN). The CNN method had about 1.3–1.7 dB higher SNR (about 15%–17% more) and about 20%–25% lower RMSE than the FD scheme for these signals. The fact that this kept getting better shows that the CNN-based method does a better job of getting rid of noise while keeping the signal features for both complex waveforms.

Figures 4–9 describe the noisy signal, the cleaned-up outputs from the FD-CN and CNN, and the original clean signal for comparison for Signals 1 to 6, respectively. These comparisons help us understand how each method smooths things out and changes the details of the signal. These figures show that both methods can get rid of loud noises. The output curves are much smoother than the noisy input, and they are closer to the real signal. The CNN outputs are better at matching the original signal's peaks than the FD-CN outputs. For instance, the CNN method makes sure that the most important peaks in each signal are at the right level and in the right place. The FD-CN method, on the other hand, results in the peak slightly wider or shorter. Signals 3 and 4 indicate what separates them the best. The CNN reconstruction makes the peaks look sharp and clear, but the FD-CN result makes them look less sharp. In the flat areas between the peaks, the FD-CN outputs change a little bit. The CNN outputs, on the other hand, usually have a smoother baseline. These observations match the data that shows the CNN gets rid of noise more effectively without changing important signal characteristics. On the other hand, the FD-CN keeps a little more low-level noise and hides certain features. The Signal 5 plot in Figure 8 shows that both the FD-CN and CNN were able to get very close to the real sinusoidal waveform by getting rid of noise over a wide range of frequencies. The CNN output looks more like the original signal's highs and lows than the FD-CN output does. Both methods can put the oscillations back together on top of the piecewise linear trend, as seen in Figure 9 for Signal 6. This makes it easier to see the signal.

The denoising tests on Signals 1–6 show that the CNN-based smoothing method works better for both numbers and words than the FD-CN method. The CNN method always gets better SNR gains and makes fewer errors. The visual tests also show that the reconstructions are clearer and that the edges and peaks are still there. The FD-CN method is still a good place to start because it makes most signals a lot better. But when things get tough, like with Signal 4, it is clear that it has some issues. These results show that the CNN-driven method works well to get rid of noise in one-dimensional signals without changing the signal's quality. The FD-CN method that uses fractional diffusion is also good, but it might need some work. It still cannot keep up with how learned CNN filters can change to keep complex signal details.

We used the time-fractional diffusion model with $L1$ Caputo discretization ($\alpha = 0.9$) and spatial fractional order $s = 0.7$ on a grid of $N = 256$ points to get the denoising results for Signals 1 and 2, which are shown in Figures 10 and 11. The time-fractional derivative adds memory effects to the diffusion process, which makes the signals that are reconstructed more stable and better at regularizing. The discrepancy principle chooses the best value for the parameter $\kappa = 0.15$ for Signal 1. The algorithm stops at $T = 1.2$, and the reconstructed signal has an $\mathrm{SNR}_{out} = 4.23\,\mathrm{dB}$ and an $\mathrm{RMSE} = 1.83 \times 10^{-1}$ from an input $\mathrm{SNR}_{in} = 5.11\,\mathrm{dB}$. When using $\kappa = 0.5$, final diffusion time $T = 0.3$, and time step $\Delta t = 2 \times 10^{-3}$ for Signal 2, the method gives $\mathrm{SNR}_{out} = 3.85\,\mathrm{dB}$ and $\mathrm{RMSE} = 1.85 \times 10^{-1}$ from an input $\mathrm{SNR}_{in} = 5\,\mathrm{dB}$.

We ran additional tests on real multichannel ECG data from the MIT-BIH Arrhythmia Database (records: 100, two leads, 360 Hz) to improve the performance of our proposed method. The model was trained and evaluated using real physiological signals with correlated noise, which is different from the previous simplified scenario [45, 46]. For $SNR_{in} = 5$ dB, the CNN improved Lead 1 from 5.00 dB to 8.49 ± 0.51 dB, while the FD-CN ($s = 0.70$) achieved only 5.39 ± 0.02 dB. On Lead 2, which exhibits higher variability, the CNN achieved 4.32 ± 1.44 dB compared to 5.28 ± 0.11 dB for the FD-CN. These results confirm that the proposed framework has been validated on real multichannel data under nontrivial noise conditions, shown in Figures 12 and 13, respectively. We then performed an additional real-world, two-lead denoising experiment (records: 100) under correlated ECG-type noise at $SNR_{in} = 5$ dB, and compared the proposed shallow residual CNN against a small one-dimensional U-shaped convolutional neural network (1D U-Net) baseline, where both networks were trained using balanced lead sampling and per-segment normalization to avoid amplitude mismatch between leads. The results shown in Figures 14 and 15 show that both deep learning models significantly improve the denoising quality compared to the noisy input, while the U-Net baseline yields slightly higher average SNR/RMSE improvement than the proposed CNN in this setting, confirming the competitiveness of the proposed architecture. Overall, this experiment demonstrates that the proposed CNN achieves strong performance with a lighter architecture, whereas the U-Net provides a stronger but more complex baseline. We also implemented the classical diffusion model corresponding to the integer-order Laplacian ($s = 1$) and compared it with the fractional-order case ($s = 0.70$) under identical multichannel real ECG conditions. Numerically, both operators provide comparable smoothing effects, with the classical model yielding an SNR $= 5.33 \pm 0.06$ dB and the FD-CN acquiring an SNR $= 5.34 \pm 0.06$ dB (average over two leads, 30 segments), shown in Figures 16 and 17, respectively. This confirms that the improvement observed in the hybrid framework is not solely due to diffusion strength, but to the structural difference between fractional and classical operators combined with the CNN refinement stage. The FD–CN diffusion operator performs best with additive white Gaussian noise. In contrast, the CNN refinement stage improves robustness to structured, correlated, or non-Gaussian noise, as shown in real multichannel ECG experiments.

The proposed framework integrates fractional nonlocal diffusion with residual deep learning refinement. This differs from variational and level-set-based segmentation-denoising algorithms, which primarily focus on region partitioning and edge detection. This hybrid method guarantees spectral stability while still enabling adaptive feature preservation. The FD–CN scheme requires stepping through time and solving linear systems for each new signal. The CNN, on the other hand, requires only one forward pass to denoise. CNN training is computationally expensive initially, but for tasks that require repeated denoising, inference is much faster.

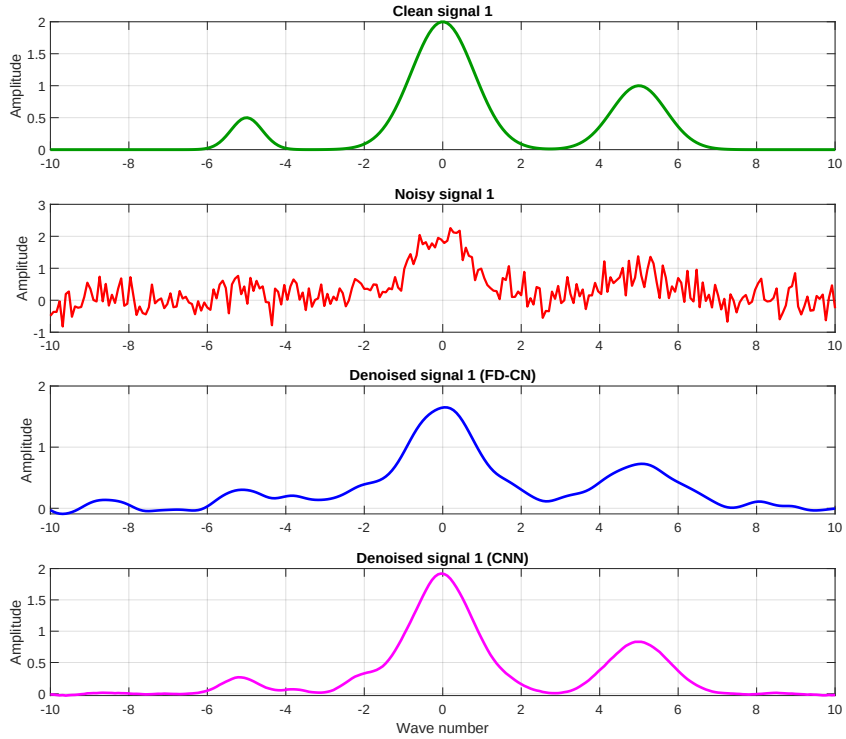

Figure 4. Signal 1 outputs of the numerical and deep learning methods illustrating the effectiveness of each approach in noise reduction and signal smoothing.

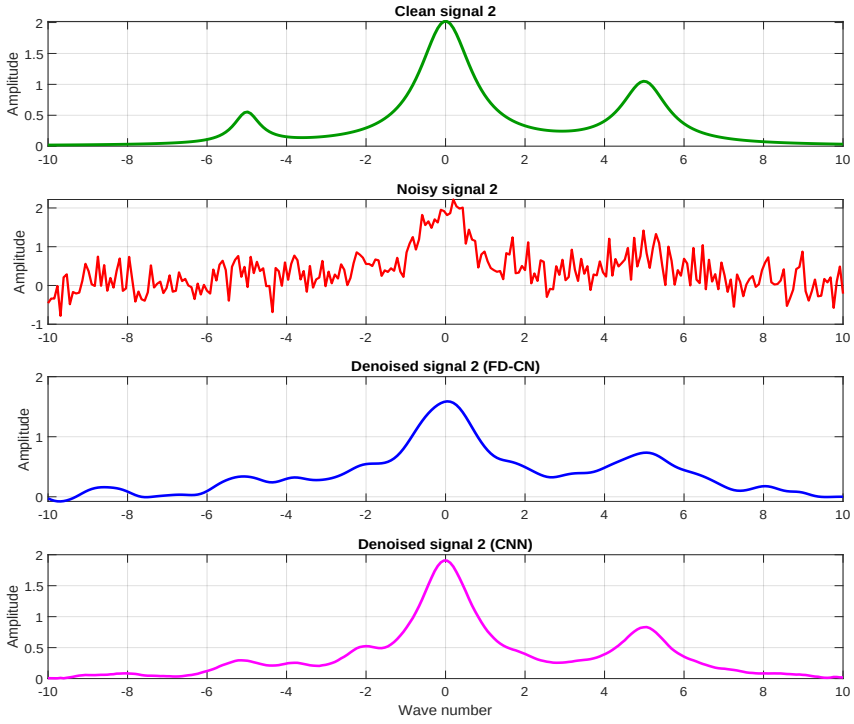

Figure 5. Signal 2 outputs of the numerical and deep learning methods illustrating the effectiveness of each approach in noise reduction and signal smoothing.

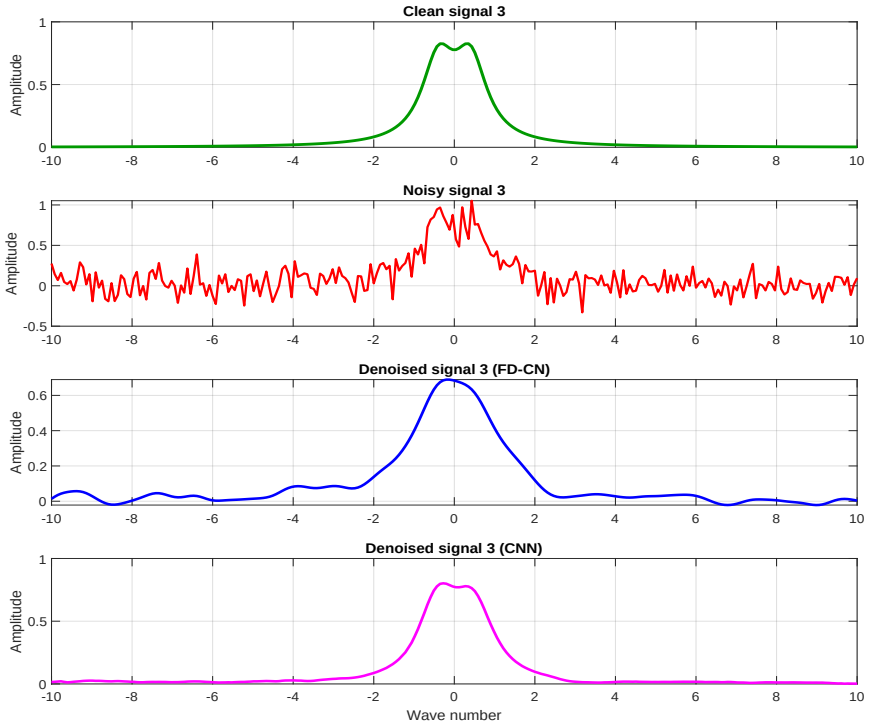

Figure 6. Signal 3 outputs of the numerical and deep learning methods illustrating the effectiveness of each approach in noise reduction and signal smoothing.

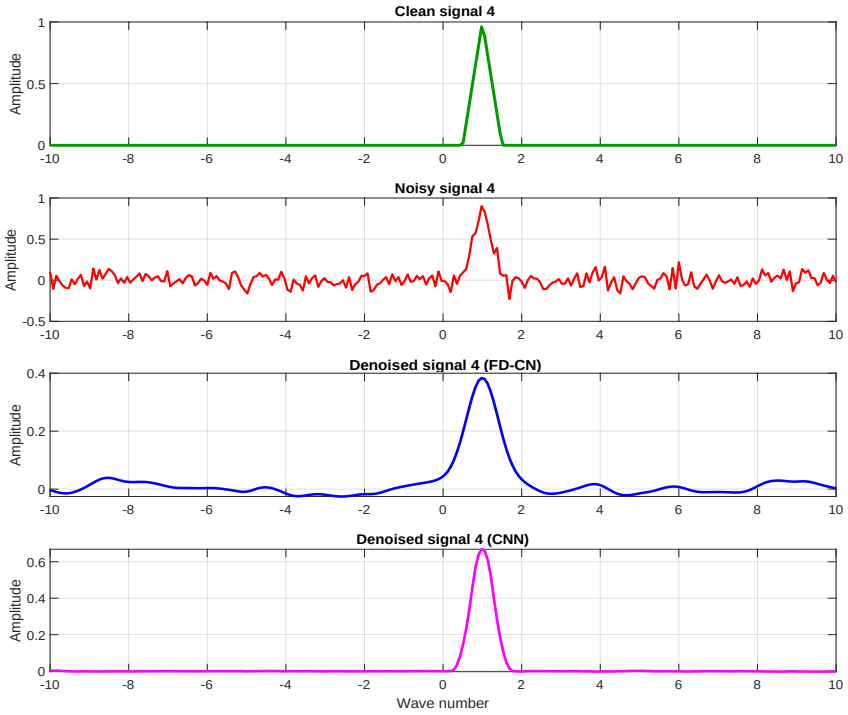

Figure 7. Signal 4 outputs of the numerical and deep learning methods illustrating the effectiveness of each approach in noise reduction and signal smoothing.

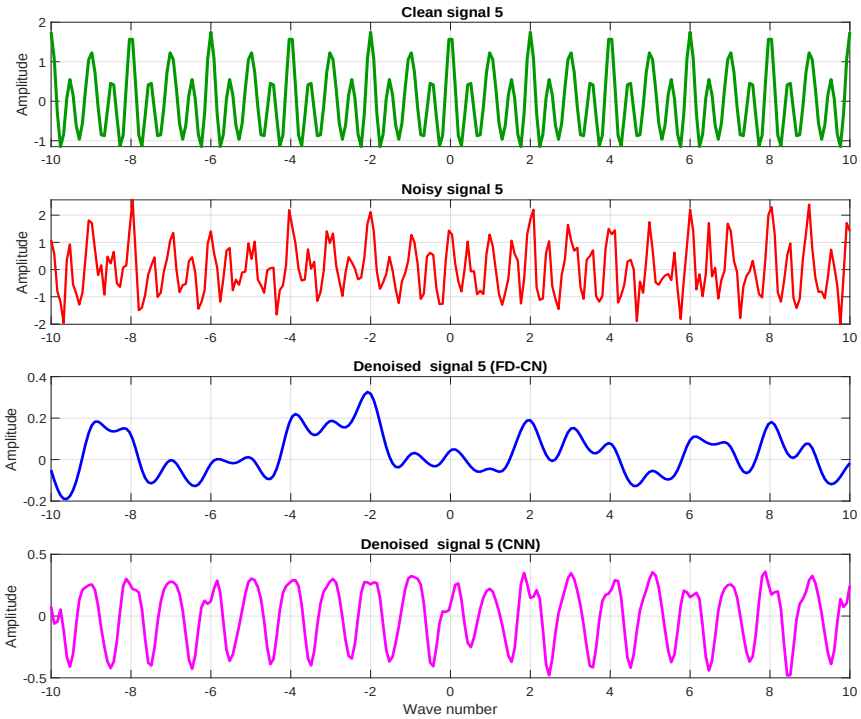

Figure 8. Signal 5 outputs of the numerical and deep learning methods illustrating the effectiveness of each approach in noise reduction and signal smoothing.

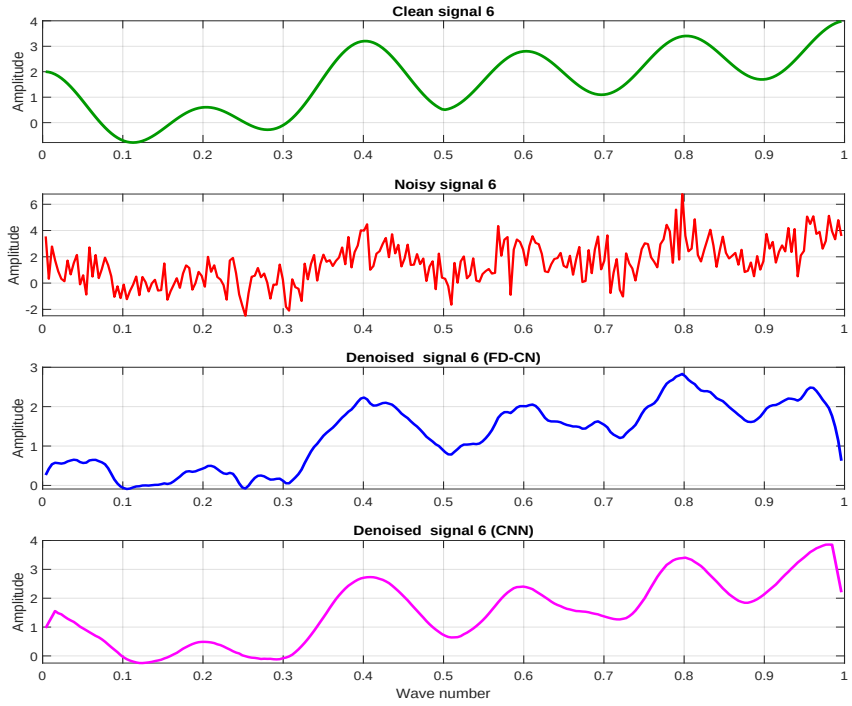

Figure 9. Signal 6 outputs of the numerical and deep learning methods illustrating the effectiveness of each approach in noise reduction and signal smoothing.

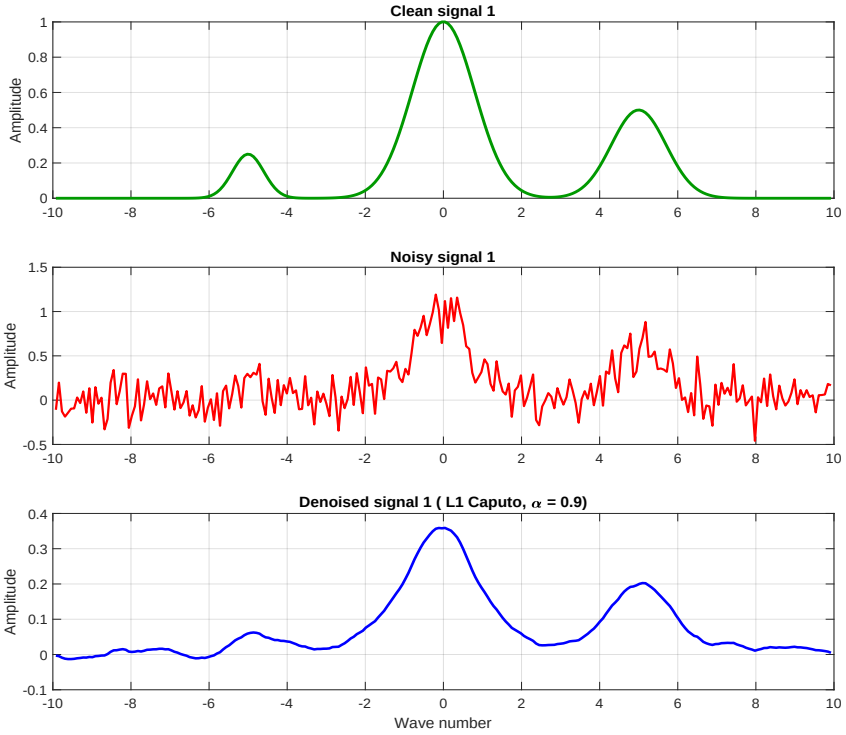

Figure 10. Signal 1 output of the time-fractional diffusion model illustrating the effectiveness of the proposed approach in noise reduction and signal reconstruction.

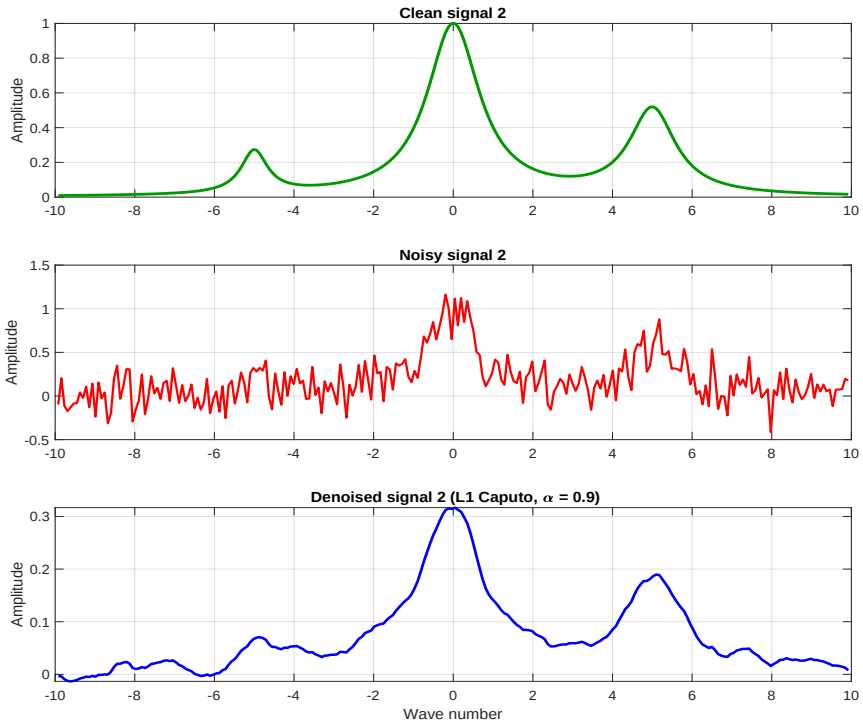

Figure 11. Signal 2 output of the time-fractional diffusion model illustrating the effectiveness of the proposed approach in noise reduction and signal reconstruction.

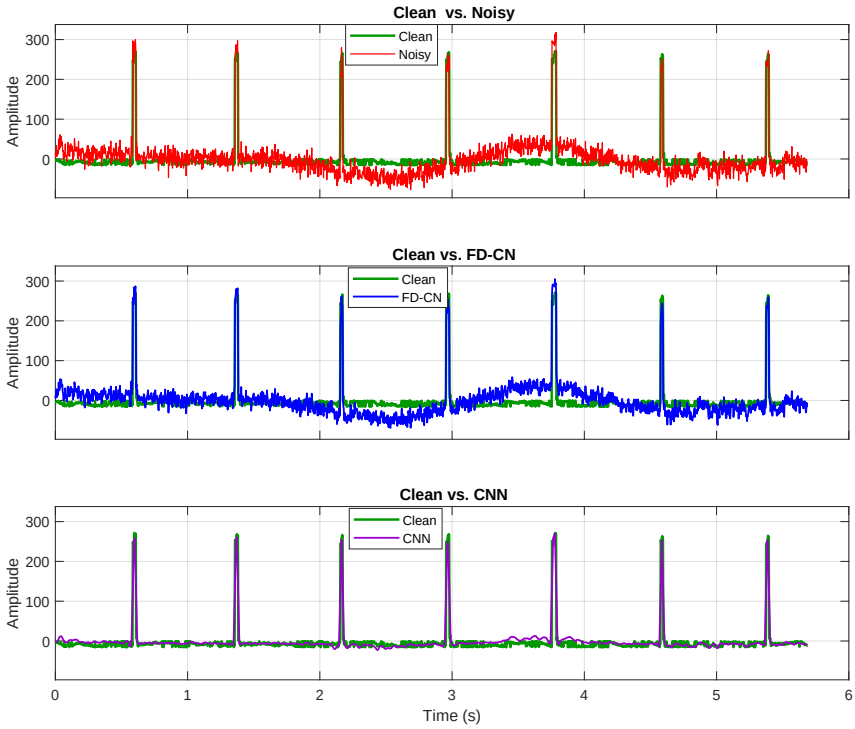

Figure 12. Denoising results for Lead 1 under realistic noisy conditions.

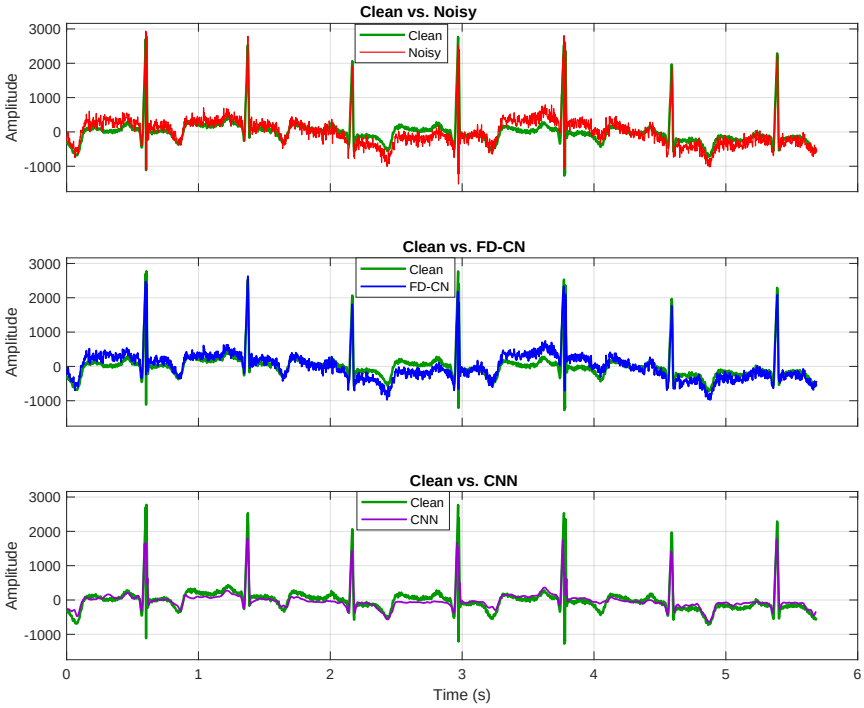

Figure 13. Denoising results for Lead 2 under multichannel real-world conditions.

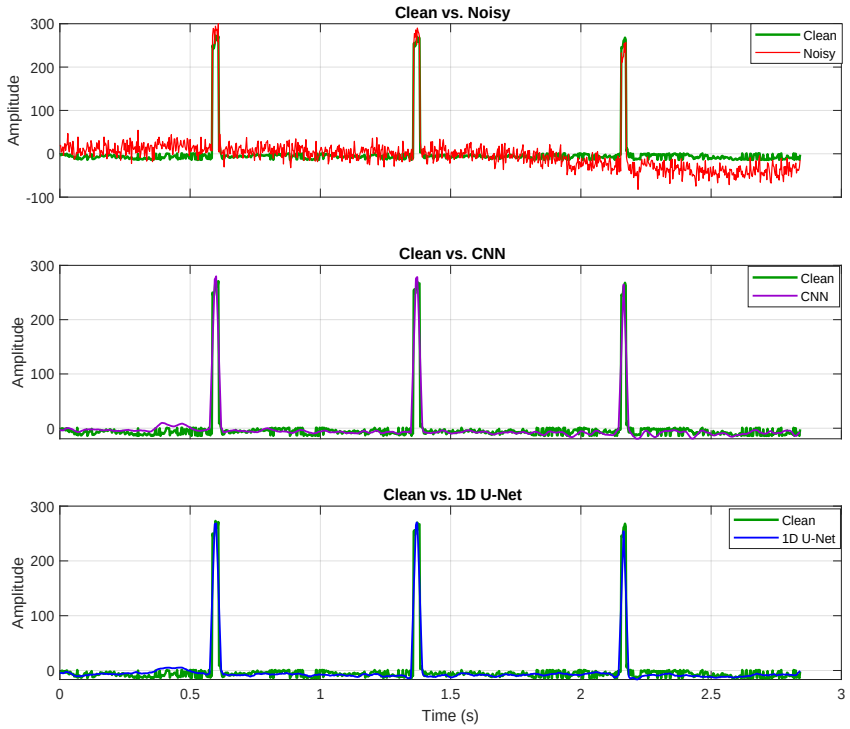

Figure 14. Comparison of CNN and 1D U-Net denoising for Lead 1.

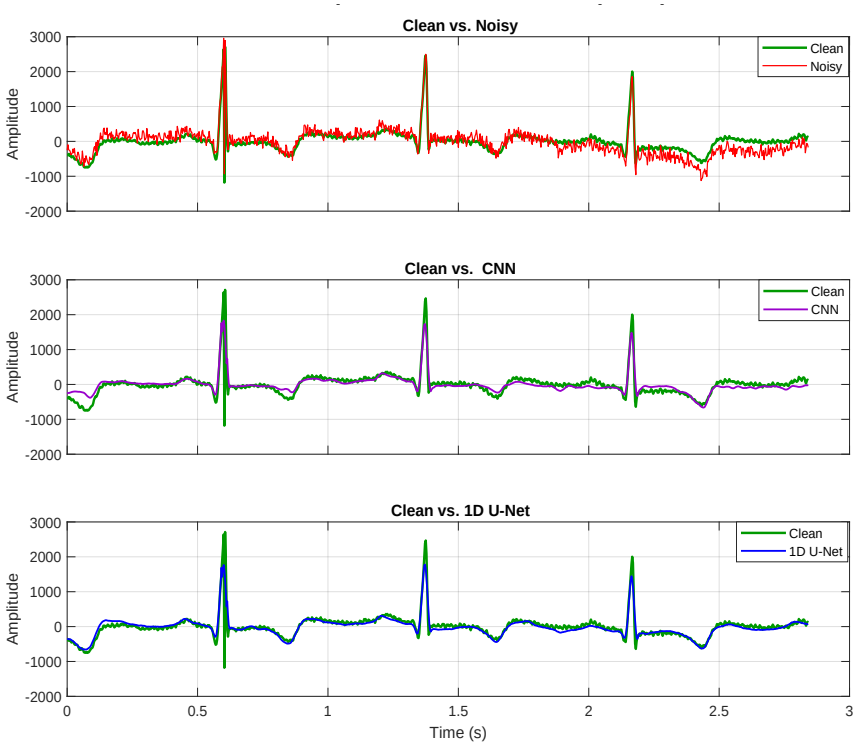

Figure 15. Comparison of CNN and 1D U-Net denoising for Lead 2.

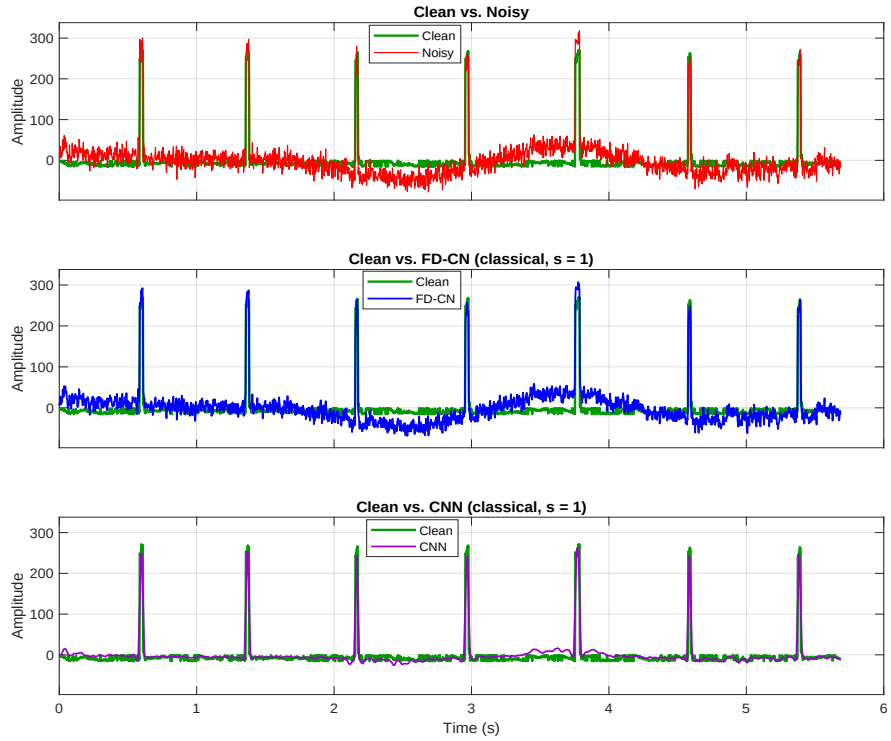

Figure 16. Classical diffusion denoising results for Lead 1 under correlated noise conditions.

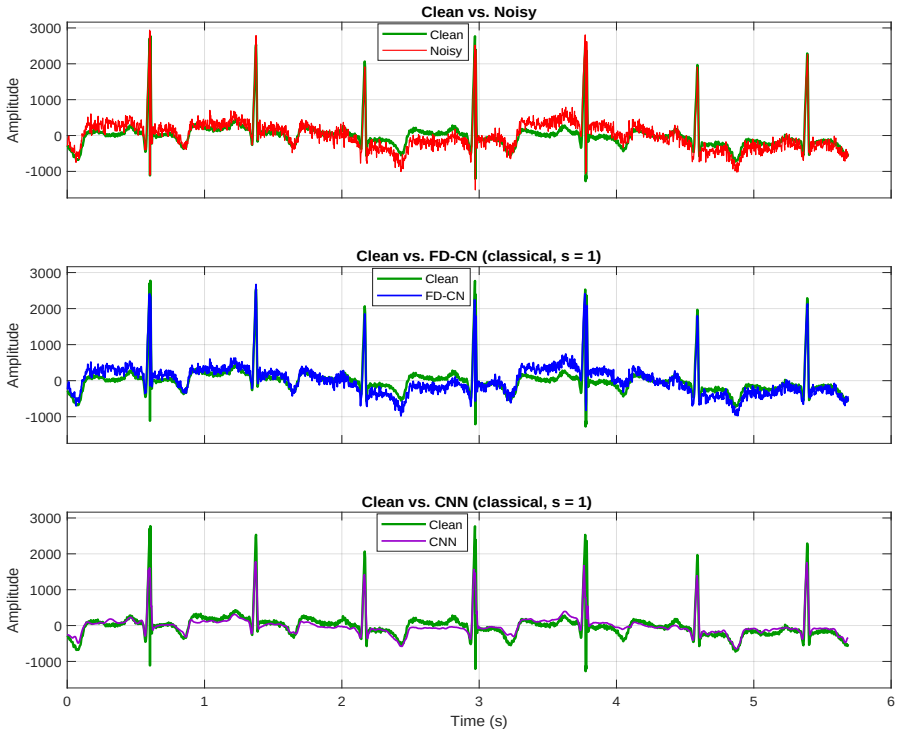

Figure 17. Classical diffusion denoising results for Lead 2 under correlated noise conditions.

4. Conclusions

In this work, we introduced a fully discrete numerical framework for denoising one-dimensional signals utilizing the fractional diffusion equation. A centered finite difference scheme was used to discretize the fractional Laplacian, and the CN method was employed to construct a second-order, unconditionally stable temporal integrator. An ε-pseudospectral analysis of the discrete diffusion operator further confirmed stability, demonstrating its robustness under perturbations. To enhance the denoising capability, a CNN was incorporated as a supplementary data-driven module, enabling nonlinear noise suppression beyond the capability of the linear PDE model. Numerical experiments on synthetic noisy signals showed that the proposed hybrid fractional diffusion-CNN approach outperforms diffusion-only methods in terms of SNR and RMSE, consistently recovering the underlying signal with higher accuracy. Furthermore, the framework was extended to multichannel real-world ECG data, where correlated noise across multiple leads was considered. The results confirmed that both the fractional diffusion model and the hybrid diffusion CNN approach remain robust under realistic data conditions, with the hybrid strategy achieving improved reconstruction quality. These findings demonstrate that the combined numerical-learning methodology provides a reliable and effective tool for signal restoration, with strong potential for applications to multichannel biomedical data and more advanced fractional models. The outlined fractional diffusion–CNN framework can be adapted for two- and three-dimensional datasets, including images and volumetric signals.

Author contributions

Ishtiaq Ali: Conceptualization, methodology, software, validation, formal analysis, resources, writing–original draft preparation, supervision, project administration, funding acquisition; Muneerah AL Nawairan: Conceptualization, validation, investigation, writing–review and editing, visualization. All authors have read and approved the final version of the manuscript for publication.

Use of Generative-AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Funding

This work was supported by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [Grant No. KFU261285].

Conflict of interest

All authors declare no conflict of interest in this paper.

References

1. P. W. Bates, *On some nonlocal evolution equations arising in materials science*, In: Nonlinear Dynamics and Evolution Equations, Fields Institute Communications, Providence, RI: American Mathematical Society, **48** (2006), 13–52. http://dx.doi.org/10.1090/fic/048/02
2. P. Carr, H. Geman, D. Madan, M. Yor, The fine structure of asset returns: An empirical investigation, *J. Bus.*, **75** (2002), 305–332. http://dx.doi.org/10.1086/338705
3. J. Blackledge, M. Blackledge, Fractional anisotropic diffusion for noise reduction in magnetic resonance images, *ISAST T. Electron. Signal Process.*, **4** (2010), 44–57. http://dx.doi.org/10.21427/D7491X
4. J. L. Vázquez, Recent progress in the theory of nonlinear diffusion with fractional Laplacian operators, *Discrete Cont. Dyn.-S*, **7** (2014), 857–885. http://dx.doi.org/10.3934/dcdss.2014.7.857
5. R. L. Magin, C. Ingo, L. C. Perez, W. Triplett, T. H. Mareci, Characterization of anomalous diffusion in porous biological tissues using fractional order derivatives and entropy, *Micropor. Mesopor. Mat.*, **178** (2013), 39–43. http://dx.doi.org/10.1016/j.micromeso.2013.02.054
6. L. Caffarelli, L. Silvestre, An extension problem related to the fractional Laplacian, *Commun. Part. Diff. Eq.*, **32** (2007), 1245–1260. http://dx.doi.org/10.1080/03605300600987306
7. C. Brändle, E. Colorado, A. de Pablo, U. Sánchez, A concave–convex elliptic problem involving the fractional Laplacian, *P. Roy. Soc. Edinb. A*, **143** (2013), 39–71. http://dx.doi.org/10.1017/S0308210511000175
8. P. R. Stinga, J. L. Torrea, Extension problem and Harnack's inequality for some fractional operators, *Commun. Part. Diff. Eq.*, **35** (2010), 2092–2122. http://dx.doi.org/10.1080/03605301003735680
9. M. Portnoff, Time-frequency representation of digital signals and systems based on short-time Fourier analysis, *IEEE T. Acoust. Speech Signal Process.*, **28** (1980), 55–69. http://dx.doi.org/10.1109/TASSP.1980.1163359
10. I. S. Reed, D. W. Tufts, X. Yu, T. K. Truong, M. T. Shih, X. Yin, Fourier analysis and signal processing by use of the Mobius inversion formula, *IEEE T. Acoust. Speech Signal Process.*, **38** (1990), 458–470. http://dx.doi.org/10.1109/29.106864
11. G. U. Reddy, M. Muralidhar, S. Varadarajan, ECG de-noising using improved thresholding based on wavelet transforms, *Int. J. Comput. Sci. Net.*, **9** (2009), 221–225.
12. I. Houamed, L. Saidi, F. Srairi, ECG signal denoising by fractional wavelet transform thresholding, *Res. Biomed. Eng.*, **36** (2020), 349–360. http://dx.doi.org/10.1007/s42600-020-00075-7
13. J. Lian, Z. Liu, H. Wang, X. Dong, Adaptive variational mode decomposition method for signal processing based on mode characteristic, *Mech. Syst. Signal Pr.*, **107** (2018), 53–77. http://dx.doi.org/10.1016/j.ymssp.2018.01.019
14. Y. Wang, F. Liu, Z. Jiang, S. He, Q. Mo, Complex variational mode decomposition for signal processing applications, *Mech. Syst. Signal Pr.*, **86** (2017), 75–85. http://dx.doi.org/10.1016/j.ymssp.2016.09.032
15. G. Rilling, P. Flandrin, P. Gonçalvès, Bivariate empirical mode decomposition, *IEEE Signal Proc. Let.*, **14** (2007), 936–939. http://dx.doi.org/10.1109/LSP.2007.904710

16. M. Alfaouri, K. Daqrouq, ECG signal denoising by wavelet transform thresholding, *Am. J. Appl. Sci.*, **5** (2008), 276–281. http://dx.doi.org/10.3844/ajassp.2008.276.281
17. A. Dixit, P. Sharma, A comparative study of wavelet thresholding for image denoising, *Int. J. Image Graph. Signal Process.*, **6** (2014), 39–46. http://dx.doi.org/10.5815/ijigsp.2014.12.06
18. M. Nazari, S. M. Sakhaei, Successive variational mode decomposition, *Signal Process.*, **174** (2020), 107610. http://dx.doi.org/10.1016/j.sigpro.2020.107610
19. N. ur Rehman, D. P. Mandic, Empirical mode decomposition for trivariate signals, *IEEE T. Signal Proces.*, **58** (2010), 1059–1068. http://dx.doi.org/10.1109/TSP.2009.2033730
20. Z. Wu, N. E. Huang, Ensemble empirical mode decomposition: A noise-assisted data analysis method, *Adv. Adapt. Data Anal.*, **1** (2009), 1–41. http://dx.doi.org/10.1142/S1793536909000047
21. F. Castells, P. Laguna, L. Sörnmo, A. Bollmann, J. M. Roig, Principal component analysis in ECG signal processing, *EURASIP J. Adv. Sig. Pr.*, **2007** (2007), 1–21. http://dx.doi.org/10.1155/2007/74580
22. J. Huang, L. Cui, Tensor singular spectrum decomposition: Multisensor denoising algorithm and application, *IEEE T. Instrum. Meas.*, **72** (2023), 3510015. http://dx.doi.org/10.1109/TIM.2023.3249249
23. M. Atemkeng, S. Perkins, E. Seck, S. Makhathini, O. Smirnov, L. Bester, B. Hugo, Lossy compression of large-scale radio interferometric data, *arXiv Preprint*, 2023. https://doi.org/10.48550/arXiv.2304.07050
24. Y. Li, F. Liu, I. W. Turner, T. Li, Time-fractional diffusion equation for signal smoothing, *Appl. Math. Comput.*, **326** (2018), 108–116. http://dx.doi.org/10.1016/j.amc.2018.01.007
25. H. K. Rafsanjani, M. H. Sedaaghi, S. Saryazdi, Efficient diffusion coefficient for image denoising, *Comput. Math. Appl.*, **72** (2016), 893–903. http://dx.doi.org/10.1016/j.camwa.2016.06.005
26. B. A. Jacobs, T. Celik, Unsupervised document image binarization using a system of nonlinear partial differential equations, *Appl. Math. Comput.*, **418** (2022), 126806. http://dx.doi.org/10.1016/j.amc.2021.126806
27. J. Blackledge, Application of the fractional diffusion equation for predicting market behaviour, *Int. J. Appl. Math.*, **40** (2010), 130–158.
28. L. Silvestre, Regularity of the obstacle problem for a fractional power of the Laplace operator, *Commun. Pur. Appl. Math.*, **60** (2007), 67–112. http://dx.doi.org/10.1002/cpa.20153
29. L. Caffarelli, L. Silvestre, An extension problem related to the fractional Laplacian, *Commun. Part. Diff. Eq.*, **32** (2007), 1245–1260. http://dx.doi.org/10.1080/03605300600987306
30. M. Kwaśnicki, Ten equivalent definitions of the fractional Laplace operator, *Fract. Calc. Appl. Anal.*, **20** (2017), 7–51. http://dx.doi.org/10.1515/fca-2017-0002
31. Q. Du, M. Gunzburger, R. B. Lehoucq, K. Zhou, Analysis and approximation of nonlocal diffusion problems with volume constraints, *SIAM Rev.*, **54** (2012), 667–696. http://dx.doi.org/10.1137/110833294
32. X. R. Oton, J. Serra, The Dirichlet problem for the fractional Laplacian: Regularity up to the boundary, *J. Math. Pure. Appl.*, **101** (2014), 275–302. http://dx.doi.org/10.1016/j.matpur.2013.06.003
33. K. M. Owolabi, E. Pindza, Modeling anomalous diffusion with Riesz fractional derivatives: Applications to pattern formation, *Numer. Meth. Part. D. E.*, **41** (2025), e70041. http://dx.doi.org/10.1002/num.70041

34. Y. Jin, S. Kwak, S. Ham, J. Kim, A fast and efficient numerical algorithm for image segmentation and denoising, *AIMS Math.*, **9** (2024), 5015–5027. http://dx.doi.org/10.3934/math.2024243
35. C. Chen, Z. Chen, Y. Zhou, Y. Hao, B. Peng, X. Xie, H. Xie, A reliable evaluation approach for multichannel signal denoising algorithms based on a novel arterial pulse acquisition system, *Heliyon*, **10** (2024), e26140. http://dx.doi.org/10.1016/j.heliyon.2024.e26140
36. A. Lanza, A. Leaci, S. Morigi, F. Tomarelli, Symmetrised fractional variation with L1 fidelity for signal denoising via Grünwald-Letnikov scheme, *Appl. Math. Comput.*, **500** (2025), 129429. http://dx.doi.org/10.1016/j.amc.2025.129429
37. A. Rifai, M. N. Rachmatullah, W. K. Sari, ECG signal denoising using 1D convolutional neural network, *Comput. Eng. Appl. J.*, **13** (2024), 60–68. http://dx.doi.org/10.18495/comengapp.v13i2.482
38. L. Qiu, W. Cai, M. Zhang, W. Zhu, L. Wang, Two-stage ECG signal denoising based on deep convolutional network, *Physiol. Meas.*, **42** (2021), 115002. http://dx.doi.org/10.1088/1361-6579/ac34ea
39. H. T. Chiang, Y. Y. Hsieh, S. W. Fu, K. H. Hung, Y. Tsao, S. Y. Chien, Noise reduction in ECG signals using fully convolutional denoising autoencoders, *IEEE Access*, **7** (2019), 60806–60813. http://dx.doi.org/10.1109/ACCESS.2019.2912036
40. X. Dong, T. Zhong, Y. Li, A deep-learning-based denoising method for multiarea surface seismic data, *IEEE Geosci. Remote S.*, **18** (2021), 925–929. http://dx.doi.org/10.1109/LGRS.2020.2989450
41. W. Zhu, S. M. Mousavi, G. C. Beroza, Seismic signal denoising and decomposition using deep neural networks, *IEEE T. Geosci. Remote*, **57** (2019), 9476–9488. http://dx.doi.org/10.1109/TGRS.2019.2926772
42. Y. Pan, Q. Luo, Y. Fan, H. Chen, D. Zhou, H. Luo, et al., Deep learning-based denoising of noisy vibration signals from wavefront sensors using BiL-DCAE, *Sensors*, **25** (2025), 5012. http://dx.doi.org/10.3390/s25165012
43. T. Li, Z. Liu, Q. Sui, C. Lu, J. Han, S. Chen, et al., Deep learning-based blind denoising for distributed acoustic sensing seismic data with self-supervised and transfer learning, *Photonic Sens.*, **15** (2025), 250434. http://dx.doi.org/10.1007/s13320-025-0769-x
44. L. N. Trefethen, M. Embree, *Spectra and pseudospectra: The behavior of nonnormal matrices and operators*, Princeton, NJ: Princeton University Press, 2005.
45. G. B. Moody, R. G. Mark, The impact of the MIT-BIH arrhythmia database, *IEEE Eng. Med. Biol.*, **20** (2001), 45–50. http://dx.doi.org/10.1109/51.932724
46. A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. Ch. Ivanov, R. G. Mark, et al., PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals, *Circulation*, **101** (2000), e215–e220. http://dx.doi.org/10.1161/01.CIR.101.23.e215