LiWave: A Lightweight 1D Wavelet-Residual U-Net for ECG Signal Denoising

Ngoc-Son Nguyen[1], Duc Nguyen-Minh[1], Phi-Thuc Vo[1],
Huy-Dung Han[1], Loan Pham-Nguyen[1], Tai Le[2], Hung Cao[2,3]
[1]*School of Electrical and Electronic Engineering (Hanoi University of Science and Technology), Hanoi, Vietnam*
[2]*Department of Biomedical Engineering, University of California Irvine, CA, USA*
[3]*Department of Electrical Engineering and Computer Science, University of California Irvine, CA, USA*
Email: son.nn6303@gmail.com, duc.nm2414980@sis.hust.edu.vn, thuc.vp224158@sis.hust.edu.vn,
dung.hanhuy@hust.edu.vn, loan.phamnguyenthanh@hust.edu.vn, tail3@uci.edu, hungcao@uci.edu

***Abstract*—Robust electrocardiogram (ECG) denoising is essential for accurate clinical diagnosis, yet preserving fine morphological details under non-stationary noise remains a challenge. This paper proposes a hybrid deep learning framework, LiWave, which integrates Discrete Wavelet Transform (DWT) into a 1D Convolutional Autoencoder enhanced by Squeeze-and-Excitation (SE) blocks. Our architecture explicitly separates signals into low- and high-frequency components through a multi-level wavelet decomposition, processing them via residual learning paths to ensure high-fidelity reconstruction. The model is trained on the MIT-BIH Arrhythmia Database and NSTDB using the mean squared error loss function and evaluated across a range of SNRs (0, 1.25, and 5 dB). Experimental results demonstrate superior performance, achieving an output SNR of up to 12.33 ± 4.55 dB and reducing RMSE to 0.15 ± 0.08 under typical noise conditions. Furthermore, external validation on the real-world BKIC dataset confirms the model's robust generalization and its ability to preserve critical P-QRS-T waveforms for downstream arrhythmia detection.**

***Index Terms*—ECG denoising, convolutional autoencoder, biomedical signal processing, SNR and morphology improvement**

I. Introduction

The electrocardiogram (ECG) is a fundamental non-invasive tool for monitoring cardiac health and detecting arrhythmias. However, ECG signals are frequently corrupted by diverse noise sources, including baseline wander (BW), electromyographic (EMG) interference, and electrode motion artifacts (MA), which can distort the clinical morphology of P-QRS-T complexes. While traditional filtering and standard deep learning models have been employed, they often struggle with a trade-off between noise suppression and the preservation of high-frequency components or transient structures.

To address these limitations, we introduce LiWave, a denoising framework that leverages the multi-resolution advantages of the Discrete Wavelet Transform (DWT) coupled with residual convolutional hierarchies. Unlike conventional autoencoders that rely on lossy pooling layers, our method utilizes wavelet-based downsampling and upsampling to maintain spectral-temporal integrity throughout the encoding-decoding process. The integration of Squeeze-and-Excitation (SE) modules further allows the network to adaptively recalibrate channel-wise features, focusing on the most informative components of the denoised signal.

The main contributions of this study are summarized as follows: Wavelet-Integrated Architecture - a hybrid design that embeds DWT/IDWT operators directly into the latent space as downsampling and upsampling layers to enhance feature separation while preserving spectral-temporal integrity; Attention-Driven Recalibration - implementation of Squeeze-and-Excitation (SE) blocks within the residual bottleneck to adaptively emphasize informative cardiac features and suppress non-stationary noise; Robust Multi-Dataset Validation - comprehensive testing on the MIT-BIH and NSTDB databases, covering representative noise levels $\{0, 1.25, 5\}$ dB to ensure stability in severe contamination scenarios [1]; Real-World Generalization - verification of the model's efficacy using the in-house BKIC dataset, demonstrating the system's readiness for practical ambulatory monitoring with high morphological fidelity [2].

II. Methodology

A. Preprocessing

Segmentation: Following the preprocessing protocol in the original article, the input ECG signal is not subjected to any additional normalization, in order to avoid distorting amplitude information and to preserve the original characteristics of the ECG waveform. The original ECG recordings are uniformly segmented into fixed-length windows of 4096 samples, and these segments are treated as clean ECG signals for the subsequent noise synthesis stage and model training [3].

Energy-Controlled Noise Injection: From each retained clean ECG segment, we generate paired clean-noisy training data via controlled additive noise injection. A noise sequence $\eta[n]$ of length N is sampled from the MIT-BIH Noise Stress Test Database (NSTDB) [1], and the noisy observation is constructed as

$$x[n] \;=\; x_0[n] + a\,\eta[n] + b, \tag{1}$$

where a is a gain factor controlling the noise strength (thus determining the target SNR), and b is a constant DC offset. The result is discrete-time sequence $x[n]$ of length N, with

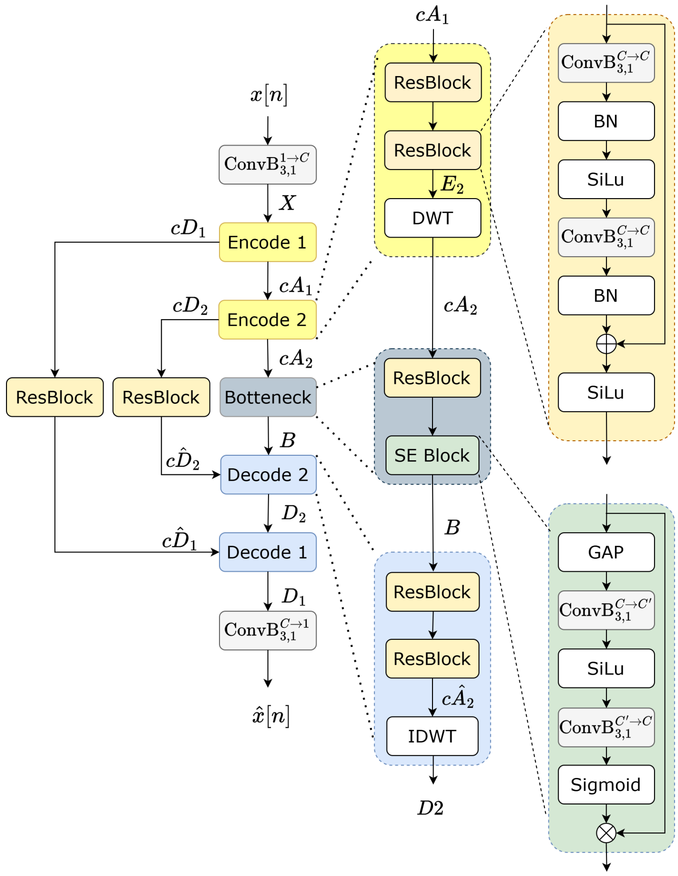

Fig. 1: Overall architecture of the proposed wavelet-based ECG denoising network.

$n = 0, 1, \ldots, N-1$. Under this construction, each segment corresponds to a single noise environment defined by one noise type and one SNR level. This formulation is consistent with the additive noise model in [3].

B. Neural Approximation of the Inverse Mapping

1) Notation and layer definitions:

a) 1D convolution: We denote a 1D convolution by

$$Q = \mathrm{Conv}_{k,\, s}^{C\to C'}(P), \tag{2}$$

where C and C' are the numbers of input and output channels, k is the kernel size, and s is the stride. Given an input tensor $P \in \mathbb{R}^{C\times N}$, the output is $Q \in \mathbb{R}^{C'\times\lfloor N/s\rfloor}$. In the proposed model, all intermediate convolutions use stride $s = 1$ with padding chosen to preserve the temporal length.

b) Residual block: Since the dropout rate is set to 0.0 in the current implementation, each residual block contains only two convolutional layers with batch normalization and SiLU activation. Let $H \in \mathbb{R}^{C\times N}$ be an intermediate feature tensor. A 1D residual block is defined as

$$Z_1 = \mathrm{SiLU}\big(\mathrm{BN}\big(\mathrm{Conv}_{3,\,1}^{C\to C}(H)\big)\big), \tag{3}$$

$$Z_2 = \mathrm{BN}\big(\mathrm{Conv}_{3,\,1}^{C\to C}(Z_1)\big), \tag{4}$$

$$\mathrm{Res}(H) = \mathrm{SiLU}(Z_2 + H), \tag{5}$$

where $\mathrm{BN}(\cdot)$ denotes batch normalization and $\mathrm{SiLU}(\cdot)$ denotes the sigmoid linear unit activation.

c) Squeeze-and-excitation (SE) block: Given an input feature tensor $X \in \mathbb{R}^{C\times N}$, the squeeze-and-excitation block first compresses the temporal dimension by global average pooling. For the c-th channel, let $x_c \in \mathbb{R}^{1\times N}$ denote the temporal feature vector of that channel. The pooled descriptor is computed as

$$z_c = \frac{1}{N}\sum_{t=1}^{N} x_c(t), \qquad c = 1, 2, \ldots, C. \tag{6}$$

By stacking all channel descriptors, we obtain

$$Z = [z_1, z_2, \ldots, z_C]^\top \in \mathbb{R}^{C\times 1}. \tag{7}$$

The descriptor is then passed through a pointwise convolution to reduce the channel dimension, followed by a SiLU activation:

$$U = \mathrm{SiLU}\Big(\mathrm{Conv}_{1,\,1}^{C\to C'}(Z)\Big), \qquad U \in \mathbb{R}^{C'\times 1}, \tag{8}$$

where C' denotes the reduced channel dimension. The compressed representation is then mapped back to the original channel dimension:

$$Y = X \odot \sigma\Big(\mathrm{Conv}_{1,\,1}^{C'\to C}(U)\Big), \qquad Y \in \mathbb{R}^{C\times N}. \tag{9}$$

This mechanism explicitly models inter-channel dependencies and adaptively emphasizes informative channels while suppressing less useful ones [4].

d) Wavelet down/up operators.: For an input tensor $H \in \mathbb{R}^{C\times N}$, a one-level discrete wavelet transform produces the low-frequency and high-frequency components:

$$cA = \mathrm{DWT}_{\mathrm{low}}(H) = f_L \otimes H \downarrow 2, \qquad cA \in \mathbb{R}^{C\times N/2}, \tag{10}$$

$$cD = \mathrm{DWT}_{\mathrm{high}}(H) = f_H \otimes H \downarrow 2, \qquad cD \in \mathbb{R}^{C\times N/2}, \tag{11}$$

The low-frequency branch is propagated through the encoder–decoder pathway, while the high-frequency components are refined via residual skip branches and later fused through inverse wavelet reconstruction. The corresponding inverse wavelet transform reconstructs the higher-resolution representation from the low- and high-frequency inputs:

$$\tilde{H} = \mathrm{IDWT}(cA, cD), \qquad \tilde{H} \in \mathbb{R}^{C\times N}. \tag{12}$$

The DWT and IDWT layers are implemented as fixed wavelet filter banks using convolutional and inverse-convolutional operations. Since these operations are differentiable in PyTorch, gradients can be propagated through the entire network during backpropagation.

e) Initial convolutional projection: Before entering the encoder, the noisy ECG signal is first mapped to a higher-dimensional feature space using a 1D convolutional projection layer. Let x denote the noisy ECG signal. The initial feature tensor is computed as

$$X = \mathrm{SiLU}\big(\mathrm{BN}\big(\mathrm{Conv}_{3,1}^{1\to C}(x)\big)\big), \qquad X \in \mathbb{R}^{C\times N}, \tag{13}$$

where C denotes the base number of feature channels and the convolution uses kernel size 3 with stride 1 and appropriate padding to preserve the temporal length.

2) Encoder 1: Let $X \in \mathbb{R}^{C\times N}$ denote the feature tensor produced by the initial convolutional projection layer applied to the noisy ECG input signal, where C is the base number of channels. In the current implementation, $C = 32$. The first encoder stage contains two consecutive residual blocks:

$$E_1 = \mathrm{Res}(\mathrm{Res}(X)), \qquad E_1 \in \mathbb{R}^{C\times N}. \tag{14}$$

A one-level discrete wavelet transform is then applied to E_1:

$$cA_1 = \mathrm{DWT}_{\mathrm{low}}(E_1), \qquad cA_1 \in \mathbb{R}^{C\times N/2}, \tag{15}$$

$$cD_1 = \mathrm{DWT}_{\mathrm{high}}(E_1), \qquad cD_1 \in \mathbb{R}^{C\times N/2}. \tag{16}$$

The high-frequency branch is refined by the first skip block:

$$c\hat{D}_1 = \mathrm{Res}(cD_1), \qquad c\hat{D}_1 \in \mathbb{R}^{C\times N/2}. \tag{17}$$

3) Encoder 2: The second encoder stage operates on the low-frequency branch from Encoder 1 and again contains two consecutive residual blocks:

$$E_2 = \mathrm{Res}(\mathrm{Res}(cA_1)), \qquad E_2 \in \mathbb{R}^{C\times N/2}. \tag{18}$$

A second wavelet decomposition is then applied:

$$cA_2 = \mathrm{DWT}_{\mathrm{low}}(E_2), \qquad cA_2 \in \mathbb{R}^{C\times N/4}, \tag{19}$$

$$cD_2 = \mathrm{DWT}_{\mathrm{high}}(E_2), \qquad cD_2 \in \mathbb{R}^{C\times N/4}. \tag{20}$$

The second skip branch is defined by

$$c\hat{D}_2 = \mathrm{Res}(cD_2), \qquad c\hat{D}_2 \in \mathbb{R}^{C\times N/4}. \tag{21}$$

4) Bottleneck: At the bottleneck, the deepest low-frequency representation is refined by a residual block and then recalibrated by the squeeze-and-excitation module:

$$B = SE(\mathrm{Res}(cA_2)), \qquad B \in \mathbb{R}^{C\times N/4}, \tag{22}$$

5) Decoder 2: The second decoder stage takes the bottleneck representation as its low-frequency input and contains two consecutive residual blocks:

$$c\hat{A}_2 = \mathrm{Res}(\mathrm{Res}(B)), \qquad c\hat{A}_2 \in \mathbb{R}^{C\times N/4}. \tag{23}$$

This feature is then combined with the refined high-frequency branch through inverse wavelet reconstruction:

$$D_2 = \mathrm{IDWT}(c\hat{A}_2, c\hat{D}_2), \qquad D_2 \in \mathbb{R}^{C\times N/2}. \tag{24}$$

6) Decoder 1: The first decoder stage again consists of two consecutive residual blocks:

$$c\hat{A}_1 = \mathrm{Res}(\mathrm{Res}(D_2)), \qquad c\hat{A}_1 \in \mathbb{R}^{C\times N/2}. \tag{25}$$

The second inverse wavelet transform restores the original temporal resolution:

$$D_1 = \mathrm{IDWT}(c\hat{A}_1, c\hat{D}_1), \qquad D_1 \in \mathbb{R}^{C\times N}. \tag{26}$$

Finally, the reconstructed feature tensor is mapped back to a denoised ECG waveform by the output convolution:

$$\hat{x}[n] = \mathrm{Conv}_{3,\,1}^{C\to 1}(D_1), \qquad \hat{x}[n] \in \mathbb{R}^{1\times N}. \tag{27}$$

C. Loss Function

The model is trained using the Mean Squared Error (MSE) loss between the denoised output and the corresponding clean ECG signal. Given the fact that $x_0[n]$ denotes the clean reference signal and $\hat{x}[n]$ denotes the model's output discrete-time signal, the loss is defined as:

$$L_{MSE} = \frac{1}{N}\sum_{n=0}^{N-1} (x_0[n] - \hat{x}[n])^2 \tag{28}$$

where N denotes the number of samples in the ECG segment. This objective penalizes reconstruction errors and encourages the network to recover the underlying cardiac morphology while reducing noise artifacts.

III. Experiments

A. Data Description

To ensure the robustness and reproducibility of the proposed model across diverse noise conditions, we aggregated data from two public databases and one locally collected dataset.

MIT-BIH Arrhythmia Database: The MIT-BIH Arrhythmia Database [5], [6] serves as the primary source of clean ECG signals. In this study, ten specific records (103, 105, 111, 116, 122, 205, 213, 219, 223, and 230) were selected to provide a balanced representation of various cardiac morphologies and rhythms. The MLII lead was selected due to its superior morphological consistency. All signals were resampled to 360 Hz and processed through a bandpass filter (0.67–100 Hz) to eliminate baseline drift and high-frequency artifacts.

MIT-BIH Noise Stress Test Database (NSTDB): To simulate real-world contamination, we utilized the MIT-BIH NSTDB [1], which contains three representative noise types: baseline wander (BW), muscle artifact (MA), and electrode motion (EM). These noise profiles were additively injected with the clean ECG segments to generate synthetic noisy inputs at SNR levels $\{0, 1.25, 5\}$ dB

BKIC Dataset: The BKIC dataset [2], collected at the BKIC Laboratory, Hanoi University of Science and Technology, was employed for external validation. This dataset consists of single-lead (MLII) ECG recordings captured under non-stationary real-world conditions, including motion and respiration artifacts. The signals were collected simultaneously with other physiological modalities such as phonocardiogram (PCG) and photoplethysmogram (PPG) using a custom non-invasive wearable acquisition system with non-contact electrodes.

B. Dataset Splitting

The integrated dataset was partitioned at the segment level within each noisy record, following the protocol described above, which is consistent with the data partitioning strategy in [7]. Subject-wise evaluation will be investigated in future studies to further assess model generalization.

- **Training and Validation (90%):** For each noise record, 80% of the segments were used for training and 10% for validation. To ensure stable yet representative noise conditions, only SNR levels of $\{0, 1.25, 5\}$ dB were considered during these phases.

- **Internal testing (10%):** The remaining 10% of the segments from each noise record were reserved for testing. Evaluation was conducted at the same representative input SNR levels $\{0, 1.25, 5\}$ dB to maintain consistency with the training conditions.
- **External Testing:** The entire BKIC dataset was used exclusively for testing to evaluate the model's performance on independent, real-world noise distributions that differ from the synthetic noise used during training.

C. Training Setup

The proposed LiWave model was implemented using the PyTorch framework and trained on an NVIDIA GPU. The training process employed a supervised learning approach, optimizing the network to minimize the Mean Squared Error (MSE) between the denoised output and the clean ground-truth signal.

The model was trained for 100 epochs with a batch size of 512. We utilized the Adam optimizer with an initial learning rate of 1×10^{-3}. The input ECG segments were fixed at a length of 4096 samples, sampled at 360 Hz. The architecture configuration used a base channel width of 32, depth parameters of 2 for the encoder/decoder stages, and a bottleneck depth of 1. SE blocks were integrated with a reduction ratio of 8 to enhance feature recalibration. The model was evaluated across multiple mother wavelets, including *haar, db2, db4, db6, sym2, sym4, sym6, coif2, coif4,* and *bior3.3*. A safe-resume mechanism was implemented to handle training interruptions and ensure architectural compatibility between checkpoints and the model.

D. Evaluation Setup

The evaluation framework was designed to compare the proposed LiWave against four state-of-the-art baselines: DW-CNN [7], ModelNet1 [7], DNN-DAN [8] and FCN [9]. The test set remained strictly isolated, consisting of 10% of the records from the MIT-BIH Arrhythmia Database.

The robustness of each model was tested against three real-world noise types: baseline wander (BW), electrode motion (EM), and muscle artifacts (MA) provided by the NSTDB. Simulations were conducted at input SNR levels of $\{0, 1.25, 5\}$ dB. The evaluation was performed across all ten trained wavelets to identify the optimal basis function for ECG denoising. Before inference, a warmup phase of 5 iterations was executed to ensure stable latency measurements. Performance was aggregated across samples and records to provide statistical mean and standard deviation for all results.

Figure 2 presents qualitative denoising results on a MIT-BIH recording under severe noise contamination (SNR_{in} = 0 dB). While all baseline methods suppress part of the noise, residual artifacts and waveform distortions remain visible, particularly around the P and T waves. FCN and ModelNet1 generate smoother outputs but still exhibit slight amplitude attenuation and temporal misalignment. In contrast, LiWave more closely follows the clean reference signal and better preserves the morphology of the P–QRS–T complexes. Similar visual trends were consistently observed across BW, EM, and MA noise types, indicating that the proposed model does not rely on characteristics specific to a single noise source. Instead, LiWave demonstrates robust denoising behavior under different interference patterns while preserving essential ECG morphology. These observations suggest that the proposed wavelet-based decomposition effectively separates noise from clinically relevant ECG structures, resulting in improved reconstruction fidelity under real-world conditions.

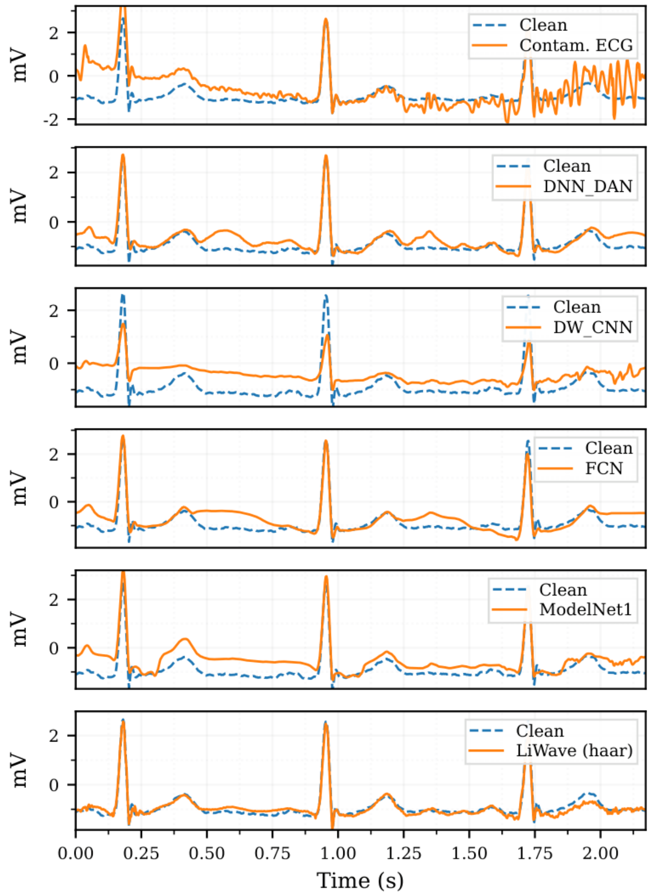

Fig. 2: Qualitative comparison of ECG denoising results on a real-world recording from the MIT-BIH noise-injected dataset under severe noise contamination (MA noise, input SNR = 0 dB). The dashed blue curve denotes the clean reference signal, while the solid orange curve represents the output of each model. From top to bottom: contaminated ECG, DNN-DAN, DW-CNN, FCN, ModelNet1, and the proposed LiWave model (haar wavelet).

Latency was measured on an NVIDIA GeForce RTX 3090 Ti using PyTorch in FP32 mode. After 5 warmup iterations, each model was evaluated for 100 timed runs with CUDA synchronization before and after each measurement. The reported latency corresponds to the average per-sample inference time at batch size 512. Although latency was evaluated on an RTX 3090 Ti for reproducibility, LiWave contains only 0.07M parameters and achieves millisecond-level inference on 4096-sample ECG segments. This lightweight design suggests potential suitability for edge and wearable devices. Future work will evaluate the model on embedded platforms and wearable SoCs.

E. Evaluation Metrics

To quantitatively assess the proposed LiWave model, RMSE and SNR are used as objective metrics to evaluate denoising quality and signal fidelity [7]. To further examine computational efficiency and real-time applicability, we report End-to-End time ($E2E$), including host-to-device data transfer:

$$E2E = \frac{t_{finish} - t_{start}}{Batch\_Size} \times 1000\,(ms) \tag{29}$$

IV. RESULTS

TABLE I: ECG denoising performance comparison across models at three input SNR levels. Results are reported as mean ± standard deviation for BW, EM, and MA noise types. Lower RMSE and higher SNR indicate better performance. Best results are in bold and second-best are underlined.

Metric	SNR$_{in}$	Model	BW	EM	MA
RMSE (↓)	0	DW_CNN	0.28 ± 0.11	0.30 ± 0.10	0.30 ± 0.11
		ModelNet1	0.30 ± 0.11	0.31 ± 0.10	0.30 ± 0.11
		DNN_DAN	$\underline{0.23 \pm 0.10}$	$\underline{0.29 \pm 0.09}$	$\underline{0.25 \pm 0.12}$
		FCN	$\underline{0.24 \pm 0.09}$	0.30 ± 0.09	$\underline{0.25 \pm 0.12}$
		LiWave	$\mathbf{0.18 \pm 0.10}$	$\mathbf{0.19 \pm 0.09}$	$\mathbf{0.20 \pm 0.11}$
	1.25	DW_CNN	0.28 ± 0.11	0.28 ± 0.09	0.28 ± 0.10
		ModelNet1	0.30 ± 0.11	0.29 ± 0.09	0.29 ± 0.12
		DNN_DAN	0.20 ± 0.12	$\underline{0.25 \pm 0.09}$	$\underline{0.24 \pm 0.10}$
		FCN	$\underline{0.19 \pm 0.12}$	$\underline{0.25 \pm 0.08}$	0.26 ± 0.11
		LiWave	$\mathbf{0.17 \pm 0.09}$	$\mathbf{0.19 \pm 0.08}$	$\mathbf{0.18 \pm 0.11}$
	5	DW_CNN	0.23 ± 0.11	0.23 ± 0.06	0.23 ± 0.09
		ModelNet1	0.25 ± 0.11	0.23 ± 0.06	0.23 ± 0.09
		DNN_DAN	$\underline{0.22 \pm 0.09}$	$\underline{0.21 \pm 0.07}$	$\underline{0.20 \pm 0.09}$
		FCN	$\underline{0.22 \pm 0.10}$	$\underline{0.21 \pm 0.06}$	0.22 ± 0.09
		LiWave	$\mathbf{0.15 \pm 0.08}$	$\mathbf{0.16 \pm 0.06}$	$\mathbf{0.16 \pm 0.09}$
SNR (↑)	0	DW_CNN	6.39 ± 3.50	5.65 ± 3.12	5.98 ± 3.50
		ModelNet1	5.84 ± 3.79	5.43 ± 2.91	5.86 ± 3.56
		DNN_DAN	$\underline{8.41 \pm 3.43}$	$\underline{5.94 \pm 2.88}$	$\underline{7.60 \pm 3.11}$
		FCN	$\underline{7.77 \pm 4.02}$	5.50 ± 3.10	$\underline{7.50 \pm 3.16}$
		LiWave	$\mathbf{10.74 \pm 4.54}$	$\mathbf{9.76 \pm 3.84}$	$\mathbf{9.98 \pm 4.11}$
	1.25	DW_CNN	6.32 ± 3.90	6.36 ± 2.77	6.26 ± 3.81
		ModelNet1	5.94 ± 4.25	5.87 ± 2.99	6.37 ± 3.33
		DNN_DAN	9.46 ± 3.78	$\underline{7.31 \pm 3.02}$	$\underline{7.88 \pm 3.58}$
		FCN	$\underline{10.05 \pm 3.81}$	7.28 ± 3.18	7.22 ± 3.24
		LiWave	$\mathbf{11.55 \pm 4.77}$	$\mathbf{9.92 \pm 3.75}$	$\mathbf{10.66 \pm 4.20}$
	5	DW_CNN	8.24 ± 4.37	7.85 ± 3.37	8.33 ± 3.50
		ModelNet1	7.78 ± 4.61	7.73 ± 3.34	8.23 ± 3.56
		DNN_DAN	8.70 ± 4.24	$\underline{8.79 \pm 3.04}$	$\underline{9.54 \pm 3.48}$
		FCN	$\underline{8.75 \pm 4.02}$	8.59 ± 3.22	8.57 ± 3.58
		LiWave	$\mathbf{12.33 \pm 4.55}$	$\mathbf{11.32 \pm 3.79}$	$\mathbf{11.74 \pm 4.34}$

Table I presents the quantitative denoising performance of the evaluated models under different noise levels and artifact types. Overall, the proposed LiWave model consistently achieves the lowest RMSE values and the highest output SNR across all input SNR conditions. The performance improvement is particularly noticeable under severe noise contamination (SNR$_{in}$ = 0 dB), where LiWave reduces the reconstruction error by a significant margin compared to the baseline models. This trend remains consistent across all three noise types (BW, EM, and MA), suggesting that the proposed architecture generalizes well to different physiological noise characteristics. As the input SNR increases, all models show improved performance; however, LiWave maintains a clear advantage in both RMSE reduction and SNR improvement. These results indicate that the wavelet-based feature decomposition effectively enhances noise suppression while preserving the morphological structure of ECG signals. Notably, the performance gap becomes more pronounced at lower input SNR levels, indicating that the proposed LiWave architecture is particularly effective in challenging denoising scenarios.

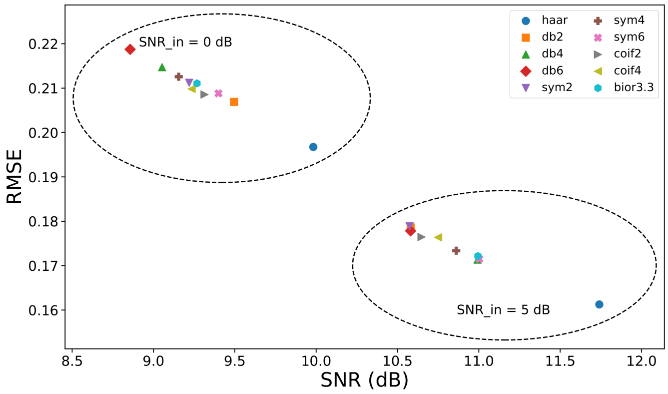

Fig. 3: Comparison of denoising performance across mother wavelets in LiWave. Each marker shows the average performance in terms of output SNR (x-axis) and RMSE (y-axis) at SNR$_{in}$ = 0 dB and 5 dB. Dashed ellipses indicate result groupings under the two noise conditions.

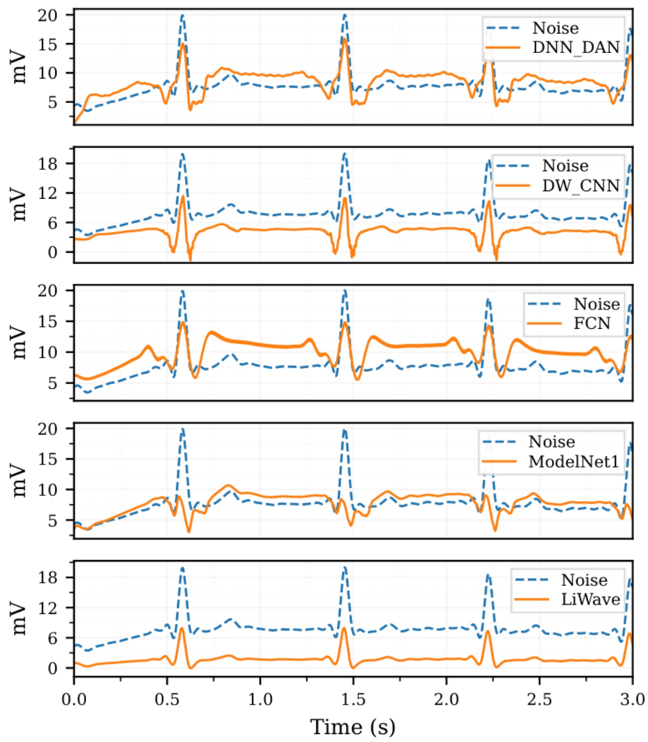

Fig. 4: The figure presents a qualitative comparison of ECG denoising results on the BKIC dataset using five models: DNN-DAN, DW-CNN, FCN, ModelNet1, and LiWave. In each subplot, the dashed line denotes the noisy input ECG, while the solid line denotes the denoised output of the corresponding model.

Figure 3 compares the reconstruction performance obtained using different mother wavelets within the LiWave framework.

Despite the variation in wavelet bases, most configurations achieve comparable output SNR values with similar RMSE levels. This indicates that the proposed architecture is relatively robust to the choice of wavelet family. In particular, several wavelets such as sym4, coif4, and bior3.3 exhibit slightly better performance, suggesting a favorable balance between noise suppression and signal reconstruction fidelity.

Figure 4 clearly shows that LiWave preserves the morphology of the P-QRS-T complexes more effectively while introducing fewer residual artifacts than the baseline models. In particular, sharper QRS complexes and smoother baseline reconstruction are observed under real-world noisy conditions.

Figure 5 illustrates the trade-off between model complexity and computational efficiency for the evaluated architectures. While larger models such as FCN contain substantially more parameters, they do not necessarily achieve lower inference latency. In contrast, the proposed LiWave model maintains a relatively small parameter count while providing competitive end-to-end processing time. This indicates that the integration of wavelet-based decomposition enables efficient feature representation without requiring a large number of learnable parameters.

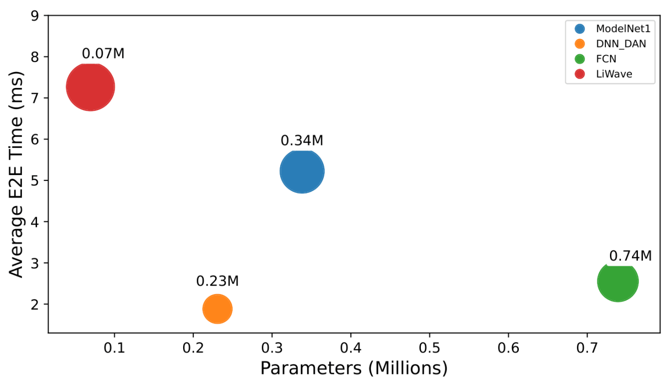

Fig. 5: Comparison of evaluated models in terms of parameter count (horizontal axis) and average end-to-end inference time (vertical axis). The bubble size represents the variation in inference time.

V. Conclusion

This paper presents LiWave, a lightweight wavelet-residual framework for robust ECG signal denoising. By integrating the Discrete Wavelet Transform (DWT) directly into a convolutional autoencoder architecture with residual connections and Squeeze-and-Excitation (SE) attention mechanisms, the proposed model effectively separates noise components from clinically relevant ECG morphology. The wavelet-based multi-resolution representation enables the network to suppress high-frequency noise while preserving the spectral and structural characteristics of the P-QRS-T complexes, consistent with the advantages of wavelet-domain analysis reported in [10], [11].

Experimental results demonstrate that LiWave consistently achieves lower RMSE and higher output SNR compared to baseline CNN-based denoising models across evaluated noise levels (0, 1.25, and 5 dB). The combination of residual learning and channel-wise feature recalibration allows the model to focus on informative signal components while reducing residual noise artifacts. The use of the MSE loss further promotes stable optimization and encourages the preservation of global ECG waveform morphology during reconstruction [3].

Additional validation on the BKIC dataset [2] further indicates that the proposed framework generalizes effectively to real-world non-stationary noise conditions, including motion artifacts and muscle interference commonly encountered in ambulatory monitoring environments, similar to the noise characteristics described in the NSTDB dataset [1].

Although the current study focuses on single-lead ECG recordings, the proposed architecture demonstrates promising potential for practical deployment due to its compact design and efficient inference capability. Since DWT and residual feature extraction are channel-agnostic, the framework can be naturally extended to multi-lead ECG through independent lead processing or multi-channel feature fusion. Future work will explore extending the framework to multi-lead clinical datasets and integrating the denoising module with downstream ECG analysis tasks such as arrhythmia classification.

References

[1] G. B. Moody and R. G. Mark, "The mit-bih noise stress test database (nstdb)." https://physionet.org/content/nstdb/1.0.0/, 1995. Accessed: January 2026.

[2] D.-A. Truong, K.-T. Hoang, H.-H. Nguyen, M.-T. Nguyen, H.-D. Han, and L. Pham-Nguyen, "Baseline wander removal in biomedical signal acquisition system for heart rate monitoring," in *2025 2nd International Conference on Health Science and Technology (ICHST)*, pp. 1–6, IEEE, 2025.

[3] E. Liu, Y. Liu, Z. Wang, H. Zhou, and X. Wang, "A noise prediction method for paper-based grayscale ecg denoising," *Biomedical Signal Processing and Control*, vol. 111, p. 108408, 2026.

[4] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141, 2018.

[5] G. B. Moody and R. G. Mark, "The impact of the mit-bih arrhythmia database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.

[6] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[7] Y. Jin, C. Qin, J. Liu, Y. Liu, Z. Li, and C. Liu, "A novel deep wavelet convolutional neural network for actual ecg signal denoising," *Biomedical Signal Processing and Control*, vol. 87, p. 105480, 2024.

[8] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning*, pp. 1096–1103, 2008.

[9] H.-T. Chiang, Y.-Y. Hsieh, S.-W. Fu, K.-H. Hung, Y. Tsao, and S.-Y. Chien, "Noise reduction in ecg signals using fully convolutional denoising autoencoders," *Ieee Access*, vol. 7, pp. 60806–60813, 2019.

[10] P. S. Addison, "Wavelet transforms and the ecg: A review," *Physiological Measurement*, vol. 26, no. 5, pp. R155–R199, 2005.

[11] T. Terada and M. Toyoura, "Wavelet integrated convolutional neural network for ecg signal denoising," in *Multimedia Modeling (MMM) 2025, LNCS 15523*, pp. 311–324, 2025.