IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 10 August 2022, accepted 30 August 2022, date of publication 14 September 2022, date of current version 22 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3206620

RESEARCH ARTICLE

Lightweight Denoising Autoencoder Design for Noise Removal in Electrocardiography

MING-HWA SHEU[1], (Member, IEEE), YU-SYUAN JHANG[1], YEN-CHING CHANG[1], SZU-TING WANG[2], CHUAN-YU CHANG[3], (Senior Member, IEEE), AND SHIN-CHI LAI[4,5], (Member, IEEE)

[1]Department of Electronics Engineering, National Yunlin University of Science & Technology, Douliu 64002, Taiwan
[2]Doctor's Program of Smart Industry Technology Research and Design, National Formosa University, Huwei 632301, Taiwan
[3]Department of Computer Science and Information Engineering, National Yunlin University of Science & Technology, Yunlin 640301, Taiwan
[4]Department of Automation Engineering, National Formosa University, Huwei 632301, Taiwan
[5]Smart Machinery and Intelligent Manufacturing Research Center, National Formosa University, Huwei 632301, Taiwan

Corresponding author: Shin-Chi Lai (shivan0111@nfu.edu.tw)

This work was supported in part by the Ministry of Science and Technology, Taiwan, under Grant MOST 110-2622-E-224 -006, Grant 110-2221-E-150 -045, and Grant 109-2221-E-150 -043; in part by the Smart Machinery and Intelligent Manufacturing Research Center through the Higher Education Sprout Project, National Formosa University, Yunlin, Taiwan; in part by the Ministry of Education (MOE) Female Researching Talent Cultivation Project for Science, Technology, Engineering, and Mathematics (STEM) Field; and in part by the Intelligent Recognition Industry Service Center from the Featured Areas Research Center Program within the Framework of the Higher Education Sprout Project of the Ministry of Education (MOE) in Taiwan.

ABSTRACT This study proposes two denoising autoencoder models with discrete cosine transform and discrete wavelet transform, to remove electrode motion artifacts in noisy electrocardiography. Initially, the discrete cosine transform and discrete wavelet transform efficiently removed the high-frequency noise. The six encoder layers then retain important electrocardiography features, whereas the six decoder layers reconstruct the clean electrocardiography. To improve the denoising performance, two network layers, the residual block and pixel adjustment, are added to the encoder and decoder layers to solve the vanishing gradient and improve subtle feature extraction. The proposed methods were applied to 66,000 real-recorded noisy electrocardiography fragments. The experimental result indicates that discrete wavelet transform based denoising autoencoder and discrete cosine transform based denoising autoencoder can improve the signal-to-noise ratio by 25.29 and 25.13 dB on average when the input signal-to-noise ratio is −6 dB.

INDEX TERMS Artificial neural networks, biomedical computing, biomedical signal processing, discrete cosine transforms, deep learning, discrete wavelet transforms, electrocardiography, noise cancellation, neural networks, signal reconstruction.

I. INTRODUCTION

The heart is the most important organ in humans. Only through continuous heartbeat, relaxation, and contraction can blood be transported to the entire body, completing full-body blood circulation. The World Trade Organization has a dataset that records 55.4 million deaths worldwide [1]. Cardiovascular diseases ranked first in the survey of the top ten causes of death from non-communicable diseases in 2019 and 2020 [1]. People have pinned their hopes for vaccines and hoped that the pandemic would recede as soon as possible, given that it has ravaged the world in the last two years. However, the vaccine has side effects, including pain at the point of injection, fatigue, headache, fever, and other symptoms. Additionally, the vaccine may cause blood clots, myocarditis, or pericarditis in severe cases. Severe myocarditis and pericarditis can result in death. These cardiovascular-related diseases are all diagnosed using electrocardiography (ECG), which graphically records the electrophysiological activity of the heart. From an electrocardiogram, changes in the heartbeat can be observed to determine whether the heart is functioning normally. However, ECG measured using an acquisition instrument generally contains noise. There are four main categories of noise: baseline-wander (BW)

The associate editor coordinating the review of this manuscript and approving it for publication was Gerard-Andre Capolino.

noise [2], muscle artifacts (MA) [3], electrode motion (EM) artifacts [4], and powerline interference (PLI) [5]. BW is the low-frequency artifact generated during respiration; MA is the high-frequency noise generated by muscle vibration; EM is the potential change generated by the friction between the electrode patch and skin; and PLI is caused by AC interference (e.g., 60 Hz or 50 Hz) owing to the alternating current in the power equipment. This noise may reduce quality and lead to misdiagnosis. Therefore, to improve the accuracy of the diagnosis, it is necessary to reconstruct a clean ECG signal, which preserves the characteristics of the ECG signal and removes noise interference.

Many researchers have proposed state-of-the-art methods to enhance ECG quality, which can be separated into two main categories: traditional and deep learning algorithms. In traditional algorithms, adaptive filters have been proven to eliminate simple noise, which can be categorized as either linear or nonlinear. The nonlinear adaptive filter has powerful noise cancellation ability and high computational complexity. Therefore, in practice, a linear adaptive filter is used. The noisy ECG and reference noise are fed into the adaptive filter simultaneously; then, the reference noise can be adjusted by the filter to make it similar to the noise magnitude in a noisy ECG. Finally, after subtracting the two signals, a denoised ECG was obtained. It is worth noting that the filter coefficient is updated according to the error of the noisy ECG and reference noise; therefore, it is more dynamically adapted to various noises than the fixed filter. For example, several approaches use the least mean square (LMS) [6], [7], [8], [9] and Kalman filter [10], [11] as the basis for developing denoising algorithms. The approach in [8] used the LMS algorithm to determine the filter coefficient and generate the minimum mean square of the error signal. The result demonstrates better noise suppression than recursive least squares and normalized LMS. However, determining a suitable step parameter in any scenario is a significant challenge. A study has been conducted [9] in which discrete wavelet transform (DWT) was performed to remove BW and an infinite impulse response notch filter was used to eliminate PLI. In addition, the author reported that white Gaussian noise can be effectively removed by three LMS-based approaches with adjustable step parameters: genetic algorithm (GA)-LMS; particle swarm optimization (PSO)-LMS; and GA-PSO-LMS. The approach in [11] applied two Kalman filters to remove the high-frequency noise of the QRS complex and the low-frequency noise of the P-wave and T-wave, respectively. The results demonstrated that the approach effectively removed noise at a low input SNR. The aforementioned adaptive filters exhibit the advantages of low complexity. However, because a noise reference is necessary in the adaptive filter, it is not the best practice. In another technique, the time-frequency analysis tool of the discrete wavelet transform (DWT) with a threshold is also adopted in noise filtering [12], [13], [14], [15]. The DWT decomposes the noisy ECG into different frequency components. Then, the noise could be eliminated by giving these components the threshold, such as the hard and soft thresholds. However, the hard threshold leads to oscillation of the reconstructed ECG, and the soft threshold may reduce the amplitude of the ECG features. The approaches in [12], [14], and [15] proposed an improved threshold to eliminate noise from ECG. The improved threshold not only avoided the reconstructed ECG oscillation but also effectively preserved the original features of the ECG. The approach in [13] adopted a nonlocal wavelet with a shrinkage threshold to remove the Gaussian noise of the ECG. Each matrix was further decomposed into many frequency components by a nonlocal wavelet, and the noise was removed by the shrinkage threshold. Combining the DWT with a threshold can effectively remove the noise of the ECG without a noise reference signal, which is necessary for adaptive filter approaches. However, in practice, it is difficult to fit any scenario by providing a suitable threshold. Empirical mode decomposition (EMD) [16], [17], [18], [19] on the other hand, decomposes a signal into several independent intrinsic mode functions (IMFs). Because ECG and noise have different IMFs, a clean ECG can be reconstructed by excluding noisy IMFs. The work [16] decomposed the IMFs of the noisy ECG using the Hilbert–Huang transform (HHT). The MA can then be identified by analyzing the entropy, mean, and variance of the F-IMF. The results showed that MAs could be effectively detected with a sensitivity and specificity of 96.63% and 94.73%, respectively. The approaches in [18] and [19] adopted the DWT threshold with variational mode decomposition (VMD), which is a variation of EMD. VMD is achieved by decomposing a noisy ECG concurrently into various variational mode functions (VMF). Then, lower-order modes containing noise were decomposed into several frequency components using DWT. Finally, the noise component was rejected by the threshold. Because each type of noise may exist in several discontinuous frequency bands, decomposing the noisy ECG into various independent components according to its characteristics using EMD or VMD is more effective than the frequency transform technique. However, the noise is often dispersed into several IMFs because of the mode-mixing separation problem in HHT, which makes analysis difficult.

Deep learning-based approaches have recently reported excellent results in noise filtering [20], [21], [22], [23], particularly the well-known denoising algorithm based on the denoising autoencoder (DAE) [24], [25], [26], [27], [28], [29], [30], [31]. Xiong *et al.* first eliminated the noise in the ECG by DWT, and the remaining noise was further removed using a deep neural network (DNN)–DAE [24]. In [25] and [26], a DAE architecture with a convolution neural network (CNN) was applied to remove noise in the ECG. The result reveals that the convolution layer extracts the clean feature effectivity more than the DNN and reduces the number of parameters. The literature in [27] added the adaptive parameter (AP), rectified linear unit (ReLU) and dual attention module (DAM) to the CNN–DAE. AP–ReLU can retain all data better than ReLU. DAM improves noise suppression by focusing on the information in each channel and space

using two different types of attention layers. The work in [28] consisted of a long short-term memory cell (LSTM) in a CNN–DAE. Because the ECG is a type of sequential data, the LSTM cell compares the relational surnames over time and provides the sequential characteristics of the CNN–DAE. The results [25], [26], [27], [28] revealed that traditional approaches performed better at noise removal. However, because DNN–DAE, CNN–DAE, and CNN–LSTM–DAE contain several fully connected layers, a larger amount of memory is required during the training phase. In [29], a fully convolutional network (FCN)–DAE was applied to reconstruct a clean ECG from a noisy ECG. The FCN-DAE requires significantly less memory than the approaches in because it only consists of convolution layers [25], [26], [27], [28]. However, the large kernel in the convolution layer loses its detailed features and incurs enormous computational complexity. In addition, with the significant achievement of generative adversarial networks (GAN) in generating fake data, some scholars have recently applied this concept to ECG noise suppression [32]. The GAN consists of a generator and a discriminator. The generator generates a reconstructed ECG by inputting a noisy ECG. The discriminator determines whether the input ECG is from the generator or real-world recorded data. If the ECG reconstructed by the generator can completely fool the discriminator, the reconstructed ECG will be very similar to a clean ECG. Therefore, it is possible to use a trained generator to eliminate the ECG noise. The results showed effective noise reduction under various noise conditions. However, because most of the ECG patterns in the database are normal rhythms, the GAN destroys the features under abnormal heart rhythms, and the overly complex network architecture makes it a huge challenge to build a lightweight model.

To solve the above problems, this study aims to provide a high-efficiency, high-noise-reduction performance and a DAE-based method that guarantees the quality of the reconstructed signal. We propose two DAE methods with DWT and a discrete cosine transform (DCT), called DWT–DAE and DCT–DAE, respectively. Both DAEs can effectively eliminate EM signals using lightweight architectures, and their low complexity enables them to be executed on low-cost CUDA devices. Because high-frequency signals are generally not the main feature of ECG, DCT and DWT are used for signal preprocessing to suppress high-frequency noise in the input layer of DAEs. The Residual block (Res.) is then applied to both the encoding and decoding ends to extract the features of the signal and avoid gradient vanishing. Meanwhile, pixel adjustment (PA) with pointwise convolution (PW Conv.) is used to expand features, merge channels without losing features, and increase the importance of the detailed features. In addition, detailed features can increase the accuracy of the reconstructed signal. The experiment shows that DWT–DAE and DCT–DAE can effectively remove noise from noisy ECG signals, and maintain full ECG characteristics at low noise levels. Compared to the methods mentioned above, the contributions of the new methods proposed herein are as follows:

- Using DCT and DWT to effectively eliminate high-frequency noise significantly reduces computation and memory usage, and using DAE to extract features from the remaining low-frequency signals and reconstruct clean ECG signals.
- A residual block and pixel adjustment layer are added to the encoder and decoder layers of each layer in the DAE to minimize the problem of gradient vanishing and improve subtle feature extraction.

II. METHODOLOGY

A. AUTOENCODER AND DENOISING AUTOENCODER

The DAE [33] is an extension of the AutoEncoder (AE, [34]), and its primary function is to remove noise from the input signal. The AE comprises two main parts: an encoder and a decoder. First, high-dimensional data ($\boldsymbol{x}$) is first input into the encoder to automatically learn the features, discard the unimportant information, and output the low-dimensional feature code ($\mathbf{z}$). Subsequently, the code was used as the input of the decoder to reconstruct the original signal ($\tilde{\boldsymbol{x}}$). Because unimportant features may be lost, AE is commonly used in lossy data compression. The formulae for the encoder and decoder can be expressed as (1) and (2):

$$\mathbf{z} = \phi(\mathbf{w}\mathbf{x} + \mathbf{b}) \tag{1}$$

$$\tilde{\mathbf{x}} = \varphi(\tilde{\mathbf{w}}\mathbf{z} + \tilde{\mathbf{b}}) \tag{2}$$

where $\mathbf{x}$ is the input signal, $\mathbf{z}$ is the output of the encoder (code), and $\tilde{\mathbf{x}}$ is the reconstructed signal. In particular, $\mathbf{x}$and $\tilde{\mathbf{x}}$ have the same signal length. $\mathbf{w}$ and $\mathbf{b}$ represent the weight and bias of the encoder, respectively. Similarly, $\tilde{\mathbf{w}}$ and $\tilde{\mathbf{b}}$ represent the weights and biases of the decoder, respectively. ϕ and φ are the activation functions of the encoder and decoder.

The DAE applies the AE structure to perform noise cancellation. The relationship between the interfered signal ($\hat{\boldsymbol{x}}$), the noise ($\boldsymbol{n}$), and the original signal can be written as $\hat{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{n}$. In contrast to AE, $\hat{\boldsymbol{x}}$ is the input of DAE, and features of the original signal are obtained from the interfered signal through the encoder. Then, the decoder reconstructs the signal $\tilde{\mathbf{x}}$ according to the features. If the DAE is sufficiently trained, the code ($\mathbf{z}$) of the encoder only contains the features of the original signal. The reconstructed signal was then similar to the original signal. For the DAE or AE to effectively train parameters, the mean square error (MSE) is usually used to calculate the loss, as shown in (3). where M denotes the signal length. DAE and AE can adjust their weights and biases using a suitable optimizer according to the loss value.

$$Loss = \frac{1}{M}\sum_{i=0}^{M-1}(x_i - \tilde{x}_i)^2 \tag{3}$$

B. DISCRETE WAVELET TRANSFORM

DWT [35] is a widely used variable time-frequency analysis tool in signal processing. It has lower complexity and better time-frequency analysis characteristics. In particular, multi-resolution can be achieved because the frequency resolution can be changed by changing the window length.

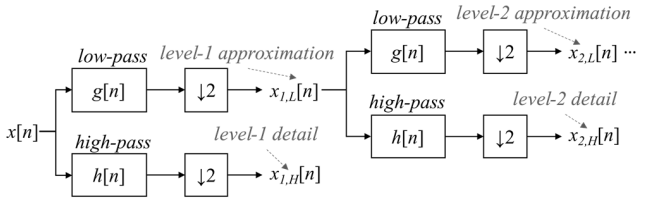

FIGURE 1. Block diagram of the DWT analysis.

First, the DWT inputs the signal into a series of low-pass and high-pass filters. Low-frequency signals are retained after passing through the low-pass filter, and then half of the signals are removed through down-sampling. Similarly, after passing through the high-pass filter, the high-frequency signal also undergoes downsampling for half of the signals to be removed. After repeating this process, the signal of the desired frequency band was retained. Accordingly, subsequent analyses of signals at different frequencies can be performed individually. The final output low- and high-frequency component signals are called approximation coefficients and detail coefficients, respectively, and the process is shown in Fig. 1.

Here, $g[n]$ is the filter coefficient of the low-pass filter, $h[n]$ is the filter coefficient of the high-pass filter, $x_{1,L}[n]$ and $x_{1,H}[n]$ are the level-1 approximation coefficients and detail coefficients, respectively. $x_{2,L}[n]$ and $x_{2,H}[n]$ are the level-2 approximation and detail coefficients, respectively.

C. DISCRETE COSINE TRANSFORM

DCT [36] is similar to the Fourier transform and is widely used in lossy compression or noise reduction; however, it only uses the real part, which is equivalent to performing a discrete Fourier transform (DFT) on even functions and discarding the imaginary part. Although DFT can reduce the computation complexity by using fast Fourier transform (FFT), such as Cooley–Tukey FFT, prime-factor FFT, and so on, the imaginary part must be calculated, making the signal and image processing more complicated. The DCT and inverse DCT (IDCT) formulas are defined in equations (4) and (5), respectively, whereas x_m and X_m are the signals in the time and frequency domains, respectively. n and m represent the signal lengths in the time and frequency domains, respectively. Cm is the coefficient.

$$X_m = \sum_{k=0}^{n-1} C_m x_k \cos\left[\frac{\pi}{n} m(k+\frac{1}{2})\right], \quad C_m = \begin{cases} \sqrt{1/n}, & m = 0 \\ \sqrt{2/n}, & else \end{cases} \tag{4}$$

$$x_k = \sum_{m=0}^{n-1} C_k X_m \cos\left[\frac{\pi}{n} m(k+\frac{1}{2})\right], \quad C_k = \begin{cases} \sqrt{1/n}, & m = 0 \\ \sqrt{2/n}, & \text{else} \end{cases} \tag{5}$$

III. PROPOSED METHODS

The structures of the proposed DCT–DAE and DWT–DAE systems are shown in Fig. 2. The DCT–DAE and DWT–DAE only have different operations in the input and output layers, whereas the encoder and decoder are essentially the same. As for the execution sequence, DWT–DAE and DCT–DAE first input the ECG signal with noise (noisy ECG) into the specified input layer to eliminate part of the high-frequency noise and amplify the features. The encoder then compresses and encodes the ECG signal features into Code (z). In contrast, the decoder restores the signal and amplifies the features according to Code (z). Finally, the DWT–DAE and DCT–DAE are input into the exclusive output layer to reconstruct a clean ECG.

A. INPUT LAYER

The noisy ECG first filters out high-frequency signals and then amplifies the number of channels through the input layer to meet the channel size required by the encoder. The noisy ECG first performs DWT with the db1 mother wavelet to obtain the level-1 approximation in DWT-DAE (Fig. 3 (a)). Pointwise convolution (PW Conv.) expanded the channel to the number of channels required by the encoder. Finally, the features were learned using residual blocks (Res.).

In the DCT–DAE, there are six steps in the input layer, and the process is shown in Fig. 3 (b). The DCT was calculated by using (4) to obtain the frequency magnitude of the noisy ECG. Then, zero padding (ZP) sets the latter half of the frequency magnitude to zero, retaining only the low-frequency energy of the first half. Finally, after converting the signal back to the time domain, the high-frequency noise energy can be suppressed by using (5) to calculating the inverse-DCT operation. Next, PW Conv and Res. also need to execute channels that meet the encoder requirements at the end of the input layer of the DCT–DAE.

Here, PW Conv. is typically used to amplify or combine channel features. By providing a kernel size of 1×1, convolution only captures the single-point feature of each channel, as shown in Fig. 4.

In convolution architecture, it is difficult to update the weight when the gradient vanishes, which means it cannot efficiently pass from the deeper layer. To solve this problem, a residual block (Res.) is widely used in deep neural networks [37]. The Res. consists of two convolutions with an additional shortcut path, as illustrated in Fig. 5. If the gradient of the convolution layer is too small, it can still be passed to the shallower layer via the shortcut [38].

B. ENCODER

After the operations in the input layer are completed, the encoder is used to reduce the dimensions of the high-dimensional feature signals. In this study, the encoders of DWT–DAE and DCT–DAE consist of five encoder layers. They are used to extract the key features of the ECG in noisy ECG and to compress them into low-dimensional features (Code Z). As shown in Fig. 6, each encoder layer consists of the Res., PA, and PW Conv, where a^n is the input signal of the n^{th} layer. a^{n+1} is the output signal of the nth layer and the input signal of the $n + 1^{th}$ layer. C represents the number

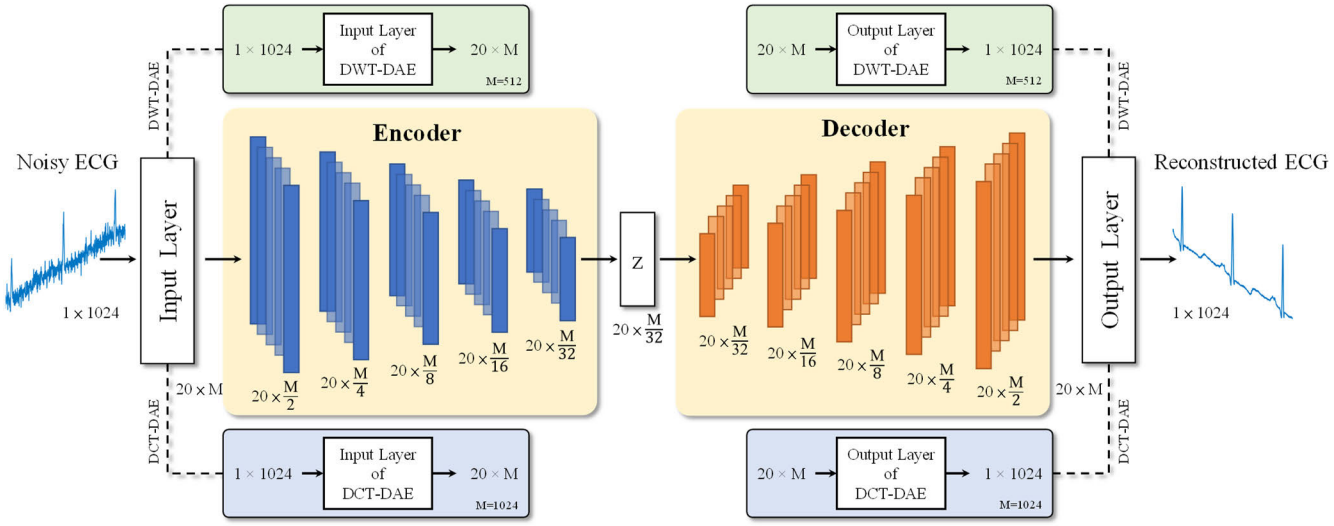

FIGURE 2. System Architecture Diagram of DCT–DAE and DWT–DAE. Two proposed DAE models have different operations in the input and output layers. For the Encoder and the Decoder, the same network structurals are adopted in the proposed two DAEs.

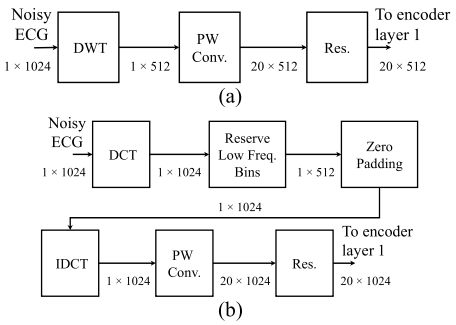

FIGURE 3. Operation of the input layer. (a) DWT–DAE. (b) DCT–DAE.

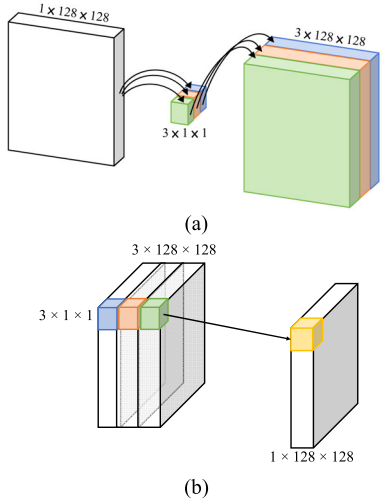

FIGURE 4. The Scenario of Pointwise Convolution (PW Conv.). (a) one channel extends to three channels through 3 × 1 × 1 convolutional kernel. (b) three channels combine into one channel through 1 × 1 × 1 convolutional kernel.

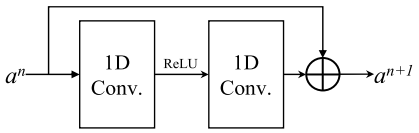

FIGURE 5. Architecture of residual block (Res.). The identity mapping is added between the start and end of the convolutional layer to solve the gradient vanishing.

of channels, and N is the number of signal samples. The Res. was used to extract the features. PA and PW Conv. were used to reduce the feature lengths for encoding. After passing through the five encoder layers, DWT–DAE outputs 20 × 16 features, whereas DCT–DAE outputs 20 × 32 features.

The PA block rearranges the signal such that the channel is doubled and the signal length is halved. As shown in (6), P' is acquired after mixing the features P, whereas the channel size × feature size of P' and P are $2C \times N/2$ and $C \times N$.

$$P'[i,j] = P\left[\lfloor i/2 \rfloor, 2j + (i \bmod 2)\right],$$
$$\text{where } i = 0, 1, \cdots, 2C-1;\ \ j = 0, 1, \cdots, \frac{N}{2}-1;\ \ \frac{N}{2} \in \mathbb{N} \tag{6}$$

C. DECODER

When the encoder obtains ECG features from the noisy ECG, the decoder can be used to reconstruct a clean ECG based on the features. The decoder consists of five decoder layers, and the architecture is completely symmetrical with that of the encoder. As shown in Fig. 7, the decoder layer consists of the Res., PW Conv., and inverse pixel adjustment (IPA). When the C×N feature (an) is input to the decoder layer, it is reconstructed into the C × 2N feature a^{n+1}, and the

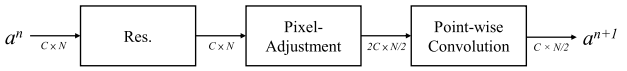

FIGURE 6. Block diagram of encoder layer. The feature size decreases to half while passing one encoder layer.

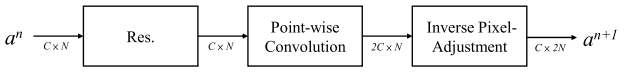

FIGURE 7. Block diagram of decoder layer. The feature size doubles while passing one decoder layer.

reconstructed feature is used as the input feature of the next decoder layer. Similar to the encoder operation, Res. was used to reconstruct the features, followed by PW Conv. and IPA, to double the size of the reconstructed features (C × 2N). The IPA equation is shown in (7), where P' and P represent the feature values before and after the adjustment, respectively, and their sizes are C × N and C/2 × 2N, respectively. i and j represent the index values of the channel and feature, respectively. Finally, after five decoder layers, DWT–DAE outputs 20 × 512 features, whereas DCT–DAE outputs 20 × 1024 features.

$$P[i,j] = P'\left[2i + (j \bmod 2)\,,\ \left\lfloor \frac{j}{2} \right\rfloor\right],$$
$$\text{where } i = 0, 1, \cdots, \frac{C}{2} - 1;\ \ j = 0, 1, \cdots, 2N-1;\ \ \frac{C}{2} \in \mathbb{N} \tag{7}$$

D. OUTPUT LAYER

Because the ECG features are scattered in different channels, the output layer can obtain the reconstructed ECG signal by combining the signal features into one channel through Res. and PW Conv. However, because half of the features of DWT–DAE have already been lost in the input layer of DWT–DAE, it needs to perform additional interpolation to fill the reconstructed ECG back to the original size, as shown in Fig. 8 (a).

E. DETAIL LAYER INFORMATION AND PARAMETERS

The detailed layer information for DWT–DAE and DCT–DAE is listed in Tables I and II. When the noisy ECG of 1 × 1024 is input into the DWT–DAE, a low-frequency signal of size 1 × 512 is obtained after passing through the DWT. The output size of the encoder was 20 × 16 pixels, and that of the decoder was 20 × 512 pixels. Finally, the output layer performs channel merging and patching, and the final output signal is 1 × 1024. In DCT–DAE, the noisy ECG with an input size of 1 × 1024 is input, and after the input layer, 1 × 1024 features are output. The output of the encoder was 20 × 32, and that of the decoder was 20 × 1024. Finally, PW. Conv. was used to generate the reconstructed ECG data. Because DWT–DAE and DCT–DAE use the same network layers, the number of trainable parameters is 56,841. However, because the feature size of DCT–DAE after the input layer is double that of DWT–DAE, the operations in multiply accumulate operations (MACs) are also doubled. In this study, we set the number of channels to 20 to conform to the lightweight network design, and the kernel size of the two convolutions in all Res. was set to 5, and the paddings were set to 2 to ensure the same size of the input feature. To retain only the ECG features for DAEs, the ReLU nonlinear function was inserted in the convolution of Res. to enhance the ECG features and discard noise.

IV. EXPERIMENT RESULT

A. DATASET SELECTION

Two ECG databases were used to train and evaluate the performance of noise reduction, namely, the MIT-BIH Noise Stress Test Database (NSTDB, [39]) and the MIT-BIH Normal Sinus Rhythm Database (NSRDB, [40]). The NSTDB has been widely used to test the denoising performance under different noise sources. The database contains three actual recorded noises, namely BW, EM, and MA, and 12 premixed noisy ECG recordings. These signals were digitized by an 11-bit ADC at 360 Hz sampling points per second to digitize the voltage value. In this study, we used 12 premixed noisy ECG recordings, which mixed six levels of EM noise (−6, 0, 6, 12, 18, and 24 dB) with the two ECG signals of record 118 and record 119 of the MIT-BIH arrhythmia database (MITDB, [41]). Because these data have been published on the PhysioNet official website, the same noisy ECG and experimental environment can be recreated. In another dataset, the NSRDB included 18 long-term ECG data points, all of which were normal heart rhythms. In this study, we mixed the 13-minute data of each record in the NSRDB with EM noise of six intensities (−6, 0, 6, 12, 18, and 24 dB) in the NSTDB so that 108 noisy ECG recordings were produced. Finally, 66,000 ECG segments of length 1024 were generated in this experiment, 80% of which were divided into a training set to train the parameters of the DAEs (52,800), and the remaining 20% were used as the testing set to test the evaluation indicators of the denoising performance of the proposed DAEs.

B. PERFORMANCE CRITERIA

In this study, improvement in the signal-to-noise ratio (SNR_{imp}), root mean square error (RMSE), and percentage root-mean-square difference (PRD) are the three criteria used to quantify the noise reduction effect of various DAEs.

SNR_{imp} shows an improvement in the SNR performance of noisy ECG and filtered ECG. When the index is high, the algorithm can suppress more noise. The equation can be written as

$$SNR_{imp} = SNR_{out} - SNR_{in} \tag{8}$$
$$SNR_{in} = 10 \times \log_{10}\left(\frac{\sum_{i=0}^{M-1} x_i^2}{\sum_{i=0}^{M-1} (\tilde{x}_i - x_i)^2}\right) \tag{9}$$
$$SNR_{out} = 10 \times \log_{10}\left(\frac{\sum_{i=0}^{M-1} x_i^2}{\sum_{i=0}^{M-1} (\hat{x}_i - x_i)^2}\right) \tag{10}$$

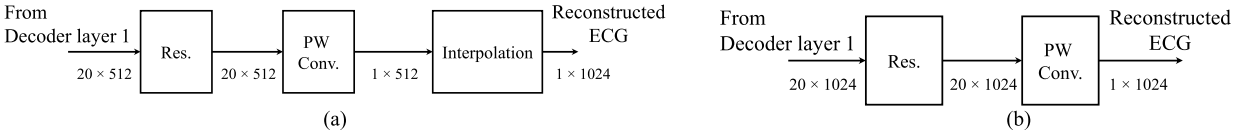

FIGURE 8. Operation of the output layer. (a) DWT–DAE. (b) DCT–DAE.

TABLE 1. Information of each layer in the Proposed DWT-DAE.

Annotation – Type	1D NN Layer name	No. filter × kernel size	Paddings	Region / unit size	*AF	No. trainable parameter	Output size
Input – Noisy ECG							1 × 1024
Input Layer	DWT	–	–	↓ 2	–	–	1 × 512
	PW Conv.	20 × 1	0	↑ 20	–	40	20 × 512
	Res.	20 × 5	2	–	ReLU	4,040	20 × 512
Encoder Layer 1	Res. + PA + PW Conv.	20 × 5	2	↓ 2	ReLU	4,860	20 × 256
Encoder Layer 2	Res. + PA + PW Conv.	20 × 5	2	↓ 2	ReLU	4,860	20 × 128
Encoder Layer 3	Res. + PA + PW Conv.	20 × 5	2	↓ 2	ReLU	4,860	20 × 64
Encoder Layer 4	Res. + PA + PW Conv.	20 × 5	2	↓ 2	ReLU	4,860	20 × 32
Encoder Layer 5	Res. + PA + PW Conv.	20 × 5	2	↓ 2	ReLU	4,860	20 × 16
Decoder Layer 5	Res. + PW Conv. + IPA	20 × 5	2	↑ 2	ReLU	4,880	20 × 32
Decoder Layer 4	Res. + PW Conv. + IPA	20 × 5	2	↑ 2	ReLU	4,880	20 × 64
Decoder Layer 3	Res. + PW Conv. + IPA	20 × 5	2	↑ 2	ReLU	4,880	20 × 128
Decoder Layer 2	Res. + PW Conv. + IPA	20 × 5	2	↑ 2	ReLU	4,880	20 × 256
Decoder Layer 1	Res. + PW Conv. + IPA	20 × 5	2	↑ 2	ReLU	4,880	20 × 512
Output Layer	Res.	20 × 5	2	↓ 20	ReLU	4,040	20 × 512
	PW Conv.	20 × 1	0	–	–	21	1 × 512
	Interpolation	–	–	–	–	–	1 × 1024
Output – Reconstructed ECG							1 × 1024
Total parameters: 56,841		**Total MACs: 11.05 M**			**Forward/Backward memory size: 1.13 MB**		

*AF: activation function; ↓: down-sampling; ↑: up-sampling

TABLE 2. Information of each layer in the Proposed DCT-DAE.

Annotation – Type	1D NN Layer name	No. filter × kernel size	Paddings	Region / unit size	*AF	No. trainable parameter	Output size
Input – Noisy ECG							1 × 1024
Input Layer	DCT	–	–	–	–	–	1 × 1024
	Reserve low freq. bins + ZP	–	–	–	–	–	1 × 1024
	IDCT	–	–	–	–	–	1 × 1024
	PW Conv.	20 × 1	0	↑ 20	–	40	20 × 1024
	Res.	20 × 5	2	–	ReLU	4,040	20 × 1024
Encoder Layer 1	Res. + PA + PW Conv.	20 × 5	2	↓ 2	ReLU	4,860	20 × 512
Encoder Layer 2	Res. + PA + PW Conv.	20 × 5	2	↓ 2	ReLU	4,860	20 × 256
Encoder Layer 3	Res. + PA + PW Conv.	20 × 5	2	↓ 2	ReLU	4,860	20 × 128
Encoder Layer 4	Res. + PA + PW Conv.	20 × 5	2	↓ 2	ReLU	4,860	20 × 64
Encoder Layer 5	Res. + PA + PW Conv.	20 × 5	2	↓ 2	ReLU	4,860	20 × 32
Decoder Layer 5	Res. + PW Conv. + IPA	20 × 5	2	↑ 2	ReLU	4,880	20 × 64
Decoder Layer 4	Res. + PW Conv. + IPA	20 × 5	2	↑ 2	ReLU	4,880	20 × 128
Decoder Layer 3	Res. + PW Conv. + IPA	20 × 5	2	↑ 2	ReLU	4,880	20 × 256
Decoder Layer 2	Res. + PW Conv. + IPA	20 × 5	2	↑ 2	ReLU	4,880	20 × 512
Decoder Layer 1	Res. + PW Conv. + IPA	20 × 5	2	↑ 2	ReLU	4,880	20 × 1024
Output Layer	Res.	20 × 5	2	↓ 20	ReLU	4,040	20 × 1024
	PW Conv.	20 × 1	0	–	–	21	1 × 1024
Output – Reconstructed ECG							1 × 1024
Total parameters: 56,841		**Total MACs: 22.1 M**			**Forward/Backward memory size: 2.26 MB**		

*AF: activation function; ↓: down-sampling; ↑: up-sampling

where SNR_{in} and SNR_{out} represent the SNR of noisy and reconstructed ECG, respectively. x_i is the voltage value of the clean ECG at sampling point i. Similarly, $\tilde{x}_i$ and $\hat{x}_i$ are the values of the noisy ECG and reconstructed ECG at sampling point (i), respectively. Finally, M is the length of the input fragment, which was fixed at 1024 in this experiment.

RMSE represents the error value between the reconstructed ECG and the clean ECG. This equation can be written as (11).

When the value of RMSE is lower, the signals of the two were more similar.

$$RMSE = \sqrt{\sum_{i=0}^{M-1} (\hat{x}_i - x_i)^2 \Big/ M} \tag{11}$$

PRD can be used to evaluate the restoration quality of the ECG signal. The lower the PRD value, the closer the reconstructed ECG and clean ECG signals are, and the equation can be written as (12). Compared with RMSE, the PRD also considers the energy of the clean ECG and presents the values using percentages. However, because the DC value of the clean ECG in each fragment may differ, it is necessary to remove the direct current of the clean ECG before calculating the PRD to achieve a fair PRD evaluation [42].

$$PRD = \sqrt{\sum_{i=0}^{M-1} (\hat{x}_i - x_i)^2 \Big/ \sum_{i=0}^{M-1} x_i^2} \times 100\% \tag{12}$$

C. EVALUATION METHODS

Four DAE algorithms, DNN–DAE, CNN–DAE, FCN–DAE [29], and CNN–LSTM–DAE, were [28] used to evaluate the proposed DAEs under the same test conditions. The FCN–DAE and CNN–LSTM–DAE are state-of-the-art algorithms, and the rest are widely used DAE architectures. DNN–DAE has 10 fully connected layers, and the number of nodes is 512, 256, 128, 64, 32, 64, 128, 256, 512, and 1024, respectively. In each fully connected layer, a ReLU nonlinear function is inserted between them to eliminate non-ECG features. The CNN–DAE has a total of six convolution layers in the encoder, which are used to extract and compress ECG features and finally output 32 features. Next, six transpose convolution layers and two fully connected layers were used to reconstruct the features of the ECG, and the ReLU was inserted into each layer to preserve the features. FCN–DAE [29] has an architecture similar to that of CNN–DAE, using six convolution layers for the encoder and seven transpose convolution layers for the decoder to reconstruct a clean ECG, and an exponential linear unit (ELU) is inserted between every layer to preserve ECG features. The CNN–LSTM–DAE [28] uses eight convolution layers and five maximum pooling layers to obtain the ECG features for the encoder. Because ECG is a sequential signal, the CNN–LSTM–DAE adds 8 LSTM cells and a fully connected layer at the end of the encoder to learn the relevant features of the sequential signal. For the decoder, eight convolutions, six up-sampling, and one fully connected layer were used to reconstruct the ECG signal. A hyperbolic tangent was used as the activation function to limit the features between −1–1.

D. EXPERIMENT DESIGN AND RESULTS

All DAEs were trained and validated using Python 3.6.9 with Pytorch 1.9.1, a framework for machine learning. The adopted CPU, RAM, and GPU in the experiment were AMD R9-5950x, 96 GB, and Nvidia RTX3090, 24 GB, respectively. The detailed training parameters are presented in Table 3. The loss function is commonly used in unsupervised learning with mean square error (MSE), and the batch size is set to 32 to speed up training. The learning rate (LR) was initially 1×10^{-4} and was halved every 200 epochs. The epochs were set to 1000 to ensure that the training parameters of all DAEs were stable. Fig. 9 shows the loss chart for each epoch of each algorithm during the training phase. The results show that the loss performances of DWT–DAE and DCT–DAE during training and testing are equivalent, CNN–DAE has the lowest loss value, and DNN–DAE and CNN–LSTM–DAE cannot effectively remove EM noise. Finally, it can be observed from the loss trend of the training phase that all algorithms tend to converge after 600 epochs. Therefore, in this experiment, we stopped the training at epochs = 1000 and verified the denoising performance during the testing phase in the last epoch.

TABLE 3. Hyperparameters in the experiment.

Hyperparameters	Value
Loss function	Mean Square Error (MSE)
Batch Size	32
Learning Rate (LR)	1×10^{-4}
Learning Rate Scheduler	Step-LR ($LR/2^{\lfloor \#\text{ of epoch} / 200 \rfloor}$)
Epochs	1000

In the last epoch, we recorded the SNR_{imp}, PRD, and RMSE of all DAEs in the testing phase, and we presented the distribution of each index of each DAE through a block diagram. The results are presented in Figs. 10 – 12. The x-axis represents the SNR of the noisy ECG signal. Fig. 10 shows the block diagram of the SNR_{imp}. When the noise level is much stronger than that of the ECG ($SNR_{in} = -6$ dB), each DAE has a good noise-reduction capability, and the average has an improvement of more than 22 dB. However, when SNR_{in} gradually increases, it is observed from the results that the two DAEs, DNN–DAE and CNN–LSTM–DAE, cannot effectively remove EM noise. Moreover, when SNR_{in} is greater than 18 dB, nearly half of SNR_{in} has negative values of SNR_{imp}, which means that the reconstructed ECG has more noise than the noisy ECG. On average, CNN–DAE, FCN–DAE, DWT–DAE, and DCT–DAE all had positive values for any SNR_{in}. However, the FCN–DAE has twice the number of outliers as other DAEs, indicating that the FCN-DAE has poor stability.

The reconstruction quality and error value of the two signals, reconstructed ECG and clean ECG, can be determined using the indicators of PRD and RMSE. Fig. 11 shows a box plot of RMSE. The RMSE distributions of DNN–DAE and CNN–LSTM–DAE were higher, and the interquartile range (IQR) was relatively wider than that of the other four DAEs. The RMSE did not decrease significantly with an increase in SNR_{in}. The RMSE distributions of DCT–DAE and DWT–DAE were similar to those of CNN–DAE, but the IQR of the proposed two DAEs was wider than that of CNN–DAE, and CNN–DAE had fewer outliers after $SNR_{in} > 6$ dB. Therefore, in the RMSE evaluation, the CNN–DAE had more advantages.

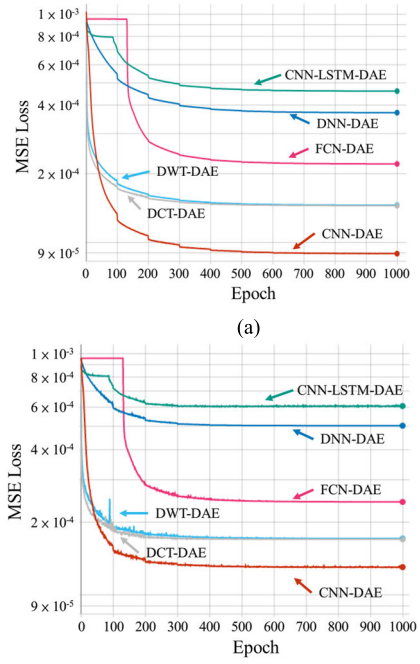

FIGURE 9. Average loss per epoch of all evaluated methods. (a) Training Phase. (b) Testing Phase.

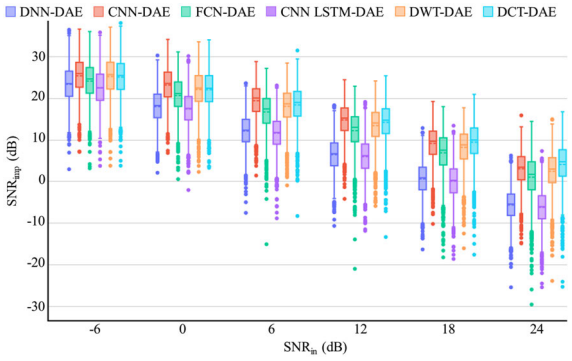

FIGURE 10. Box plots for SNR_{imp} group by six levels of SNRin. The box plots include outliers (dot), minimum, interquartile range, median, maximum, and average (dotted line). All DAEs significantly improve in $SNR_{in} = -6 - 12$ dB. However, DNN–DAE and CNN–LSTM–DAE cannot effectively improve SNR at 18 and 24 dB.

Fig. 12 shows the box plot of the PRD. When the noise level exceeds the ECG ($SNR_{in} = -6$ dB), the subtle features of the ECG are covered by noise, and all DAEs have difficulty retaining the characteristics of the ECG from the noisy ECG, resulting in high PRD values. Conversely, if the noise is much smaller than that of the ECG ($SNR_{in} = 24$ dB), the features of the ECG are very obvious, and all DAEs easily capture the features, resulting in the reconstructed ECG being closer to the clean ECG. The results reveal that quartile 1 (Q1) of DNN–DAE and CNN–LSTM–DAE is significantly higher than the others when $SNR_{in} = -6$ dB, which means that 75% of the denoising results of the PRD are greater than 85%. Furthermore, even if the SNR_{in} is increased to 24 dB, the PRD of these two algorithms is still not reduced, which means that DNN–DAE and CNN–LSTM–DAE cannot

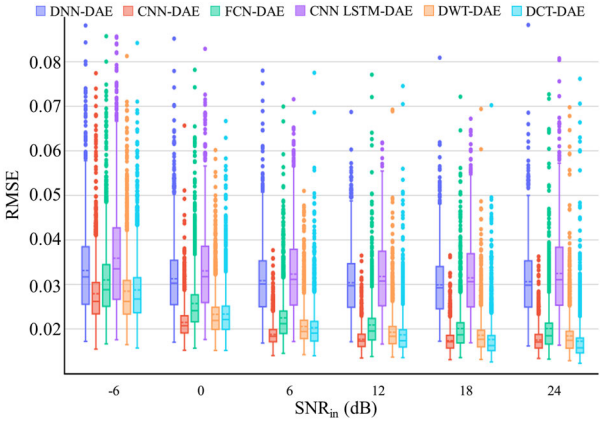

FIGURE 11. Box plots for RMSE group by six levels of SNR_{in}. The box plots include outliers (dot), minimum, interquartile range, median, maximum, and average (dotted line). When $SNR_{in} = -6$ dB, all reconstructed ECGs are significantly different than clean ECG. However, when SNR_{in} is 6 – 24 dB, the quartile 3 (Q3) of CNN–DAE, FCN–DAE, DCT–DAE and DWT–DAE have declined.

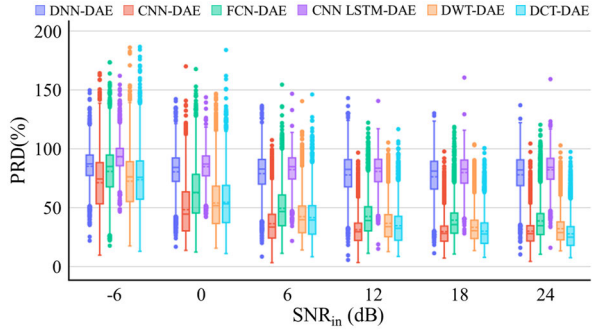

FIGURE 12. Box plots for PRD group by six levels of SNR_{in}. The box plots include outliers (dot), minimum, interquartile range, median, maximum, and average (dotted line). Similar to the PRD metrics, the Q3 of CNN–DAE, FCN–DAE, DCT–DAE, and DWT–DAE are significantly reduced when $SNR_{in} = 6 - 24$ dB. However, DNN–DAE and CNN–LSTM–DAE are substantially different from the clean ECG in any SNR_{in}.

effectively remove the noise in noisy ECG. On the other hand, when SNR_{in} gradually increases, the PRD distributions of CNN–DAE, FCN–DAE, DCT–DAE, and DWT–DAE gradually decrease, which means that the reconstructed signal quality of these four DAEs is higher when there is lower noise. The performances of CNN–DAE, DWT–DAE, and DCT–DAE were similar. With $SNR_{in} = 24$ dB, half of the PRD values of CNN–DAE, DWT–DAE, and DCT–DAE between 21.58% – 34.51%, 22.41% – 33.84%, and 17.82% – 33.74%, respectively. When the range of IQR is narrow, it means that the quality of reconstruction is more stable; therefore, DCT–DAE has the best performance in this evaluation. In summary, the DAE and CNN–LSTM–LDAE cannot effectively remove the EM noise of ECG signals, and the performance of the FCN–DAE [29] is balanced. However, the RMSE index of CNN–DAE shows that its reconstructed ECG signal is closer to the clean ECG, and it is also similar to the proposed DWT–DAET and DCT–DAE. It is worth mentioning that the proposed DCT–DAE has significantly better reconstruction quality than CNN–DAE in terms of PRD.

TABLE 4. Performance comparison for proposed DAEs and state-of-the-art.

DAE Method	SNR_{imp} (dB)						PRD (%)					
	SNR_{in} -6 dB	SNR_{in} 0 dB	SNR_{in} 6 dB	SNR_{in} 12 dB	SNR_{in} 18 dB	SNR_{in} 24 dB	SNR_{in} -6 dB	SNR_{in} 0 dB	SNR_{in} 6 dB	SNR_{in} 12 dB	SNR_{in} 18 dB	SNR_{in} 24 dB
DNN	23.41	18.05	12.15	6.47	0.66	-5.62	85.07	80.39	78.86	77.78	76.17	77.95
CNN	25.45	23.21	19.47	14.83	9.19	3.09	71.12	48.22	36.20	31.01	29.54	29.82
FCN [29]	24.16	20.69	16.75	12.23	6.93	1.06	80.79	62.50	49.20	42.32	39.15	38.58
CNN–LSTM [28]	22.55	17.46	11.65	5.98	0.10	-6.21	93.05	84.75	82.24	80.79	79.79	82.10
DWT	25.29	23.13	18.80	13.90	8.23	2.50	72.53	53.84	42.16	36.53	33.18	32.09
DCT	25.13	22.90	19.24	14.72	9.44	4.74	72.71	54.40	41.45	34.51	29.30	24.85

Table 4 lists the average SNR_{imp} and average PRD of all DAEs. When $SNR_{in} = -6 - 12$ dB, CNN–DAE has the best performance, and DWT–DAE is about 1 dB behind CNN–DAE; DCT–DAE has the best performance at $SNR_{in} =$ 18 and 24 dB; FCN–DAE has an improvement with any SNR_{in}. DNN–DAE and CNN–LSTM–DAE have no significant reduction in PRD with any SNR_{in}. When SNR_{in} is small, CNN–DAE is slightly better than DWT–DAE and DCT–DAE.

Table 5 lists the Trainable Parameters (TPs), MACs, memory usage, and average runtimes of all the compared DAEs. The first and second values for each metric are highlighted in red and blue, respectively. DNN–DAE, CNN–DAE, and CNN–LSTM–DAE have much larger TPs than FCN–DAE, DWT–DAE, and DCT–DAE because they have fully connected layers. DNN–DAE has far smaller MACs than the other DAEs because it does not have a convolution layer. The DWT–DAE and DCT–DAE use fewer convolution layers, and thus the MACs are smaller than those of FCN–DAE [29] and CNN–LSTM–DAE [28] Although DCT–DAE has the same number of parameters as DWT–DAE, in the input layer, the number of features output by DWT is only half that of DCT, so the MACs of DWT–DAE are half that of DCT–DAE. Regarding memory usage, only 1.06 MB of memory is required for one-time inference in FCN–DAE. The memory usage of DWT–DAE and DCT–DAE is slightly higher than that of FCN–DAE. Because Res. is adopted in every encoder and decoder layer, it requires additional memory to retain the features of the short path in Res. The memories of DNN-DAE, CNN–DAE, and CNN–LSTM–DAE are all greater than 5 MB because of the large number of TPs in the fully connected layer. In terms of the average runtime, DNN-DAE required the shortest average runtime of the two phases. DWT–DAE and DCT–DAE were 0.1342 ms and 0.1482 ms, respectively, during the testing phase. The remaining DAEs are all greater than 0.2ms. Although This indicates that the CNN–DAE performs better in terms of average SNR_{imp} and PRD in Table 4. However, CNN–DAE uses more than 1.1 million parameters, which is 19 times that of the proposed DWT–DAE and DCT–DAE methods. Moreover, compared to FCN–DAE, both DWT–DAE and DCT–DAE can use fewer parameters and MACs to achieve better results.

Fig. 13 shows the experimental results for the three ECG fragments. The three noisy ECGs show that it could be of great help in the diagnosis of abnormal rhythms if the clean ECG could be restored through DAE. When $SNR_{in} = 0$ dB

TABLE 5. Computational comparison for proposed DAEs and state-of-the-art.

DAE Method	# of Trainable Parameters	# of MACs	Memory usage	Average Runtime (ms)	
				Training phase	Testing phase
DNN	1,399,712	1.4 M	5.63 MB	0.5034	0.1151
CNN	1,116,478	13.27 M	5.32 MB	0.6537	0.1790
FCN [29]	78,444	25.08 M	1.06 MB	0.6437	0.2006
CNN-LSTM [28]	10,920,532	46.69 M	45.02 MB	0.8329	0.3935
DWT	56,841	11.10 M	1.36 MB	0.5934	0.1342
DCT	56,841	22.20 M	2.49 MB	0.6241	0.1482

(Fig. 13 (a)), the CNN–DAE is the closest to the clean ECG, but one QRS complex is distorted. Neither DNN–DAE nor CNN–LSTM–DAE can effectively obtain ECG features; FCN-DAE can restore R peak features, but there is a slight level shift and T wave distortion; DWT–DAE and DCT–DAE can effectively reconstruct all ECG features, but the distortion is found in the first QRS complex; and DCT–DAE can effectively reconstruct all ECG features. The experimental results in Fig. 13 (b) show that DNN–DAE can restore ECG features due to the weakening of noise, but there is still a significant difference between the reconstructed ECG signal and clean ECG; CNN–DAE can effectively suppress noise, but the reconstructed signal has some high-frequency noise; the reconstructed ECG by FCN–DAE has slightly suppressed the amplitude of T waves; CNN–LSTM–DAE and DNN–DAE can only restore part of the ECG features, but high-frequency noise is still corrupted the reconstructed ECG. DWT–DAE and clean ECG almost fully overlap, almost perfectly removing EM noise. DCT–DAE is also very similar to clean ECG, and the energy at the R peak can be completely restored. With a SNR_{in} of 12 dB (Fig. 13 (c)), the ECG reconstructed by DNN–DAE is broken, and only the features of the QRS complex can be seen; CNN–DAE performs well in terms of noise reduction, but high-frequency noise still exists; The level of FCN–DAE is slightly shifted, and the first T wave is not correctly restored. CNN–LSTM–DAE can only correctly restore one QRS complex, and other waveforms show obvious destruction; DWT–DAE and DCT–DAE both provide reasonable reconstructions.

V. DISCUSSION AND CONCLUSION

In the overall evaluation, the ECG reconstructed using DNN–DAE, CNN–DAE, and CNN–LSTM–DAE showed high-frequency signal jitter. According to the ECG features, because the ECG only exhibits significant changes in the QRS

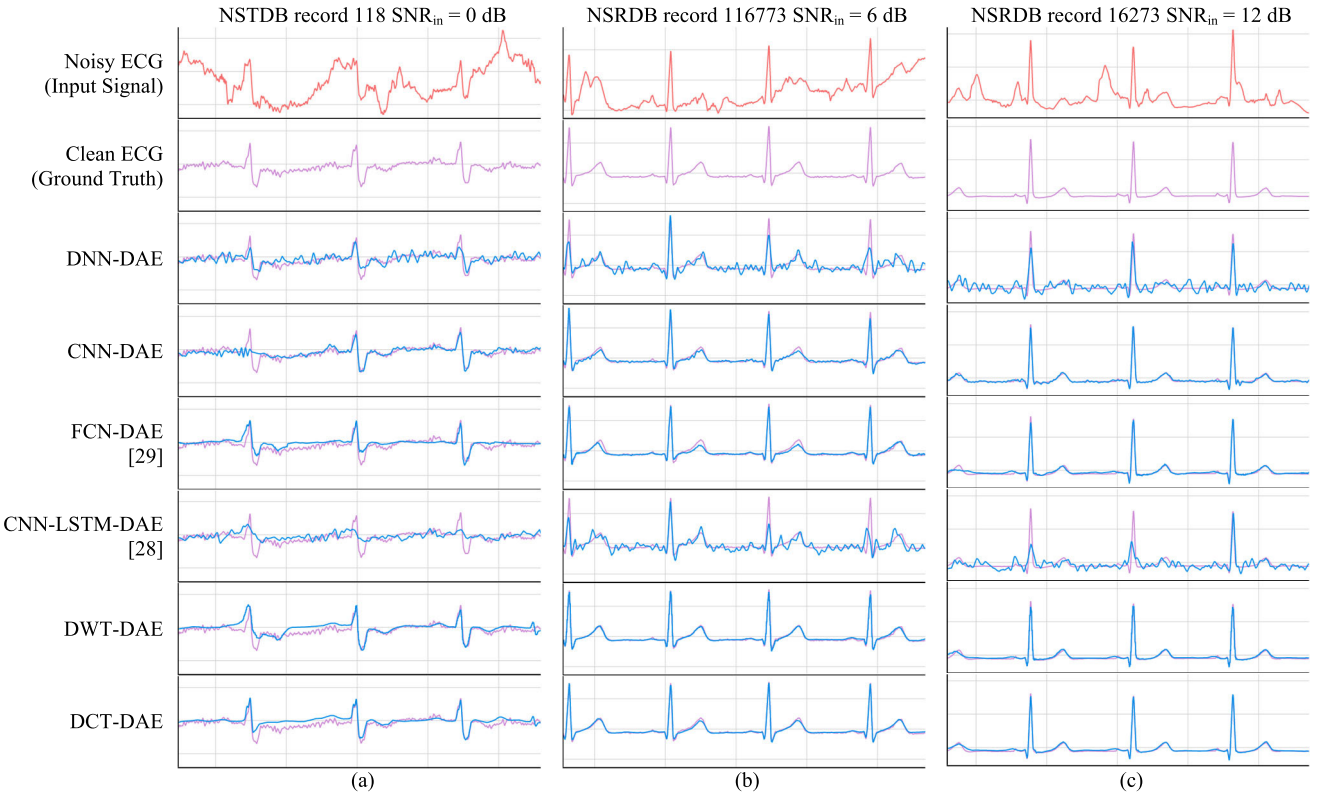

FIGURE 13. The reconstructed ECG of the evaluated models under. (a) $SNR_{in} = 0$ dB. Due to the intense noise of noisy ECG, the ECG features are obscured. FCN-DAE, DWT-DAE, and DCT-DAE are able to distinguish the QRS complex. (b)In $SNR_{in} = 6$ dB, noisy ECG, the QRS Complex feature is more prominent, but there is baseline drift, and the P wave and T wave have interfered. The ECG reconstructed by CNN-DAE, DWT-DAE, and DCT-DAE is the most similar to Clean ECG. (c) With less noise at SNRin = 12 dB, CNN-DAE, FCN-DAE, DWT-DAE, and DCT-DAE can effectively reconstruct ECG signals.

complex, the rest of the time is stable. Therefore, if a fully connected layer is used, the strong features of the R-peak may generate high-frequency noise in the reconstructed ECG. Similarly, the amplitude of the predicted R peak is lower than that of the original R peak owing to the reference to the remaining lower features (e.g., Fig. 13 (b)). In contrast, FCN–DAE, DWT–DAE, and DCT–DAE only consist of the convolution layer, so the reconstructed ECG only refers to the signal features of the region. However, owing to the lack of a fully connected layer, DAEs, which only consist of a convolution layer, can only use a very few parameters to reconstruct the features of ECG details. This indicates that FCN–DAE, DWT–DAE, and DCT–DAE can not efficiently reconstruct the ECG compared to CNN–DAE in the scenario of high-level noise.

In this study, because high-frequency signals are often not the main source of ECG features, DCT and DWT, in which only a small amount of calculation is required, are initially used to effectively reduce the high-frequency noise of noisy ECG. Subsequently, we use the encoder to preserve the ECG features and the decoder to reconstruct the ECG signal. To enhance the effect of feature extraction, we used techniques such as residual block and pixel adjustment to avoid gradient vanishing and to enhance the effect of detailed feature extraction. The results demonstrate that the proposed DAE can effectively eliminate MA noise. When $SNR_{in} = -6$ and 0 dB, DWT–DAE and DCT–DAE use fewer parameters that only decrease 0.5 dB with CNN–DAE in SNR_{imp}, and the PRD error falls within 6%. It is worth noting that when the level of noise is small, the proposed DCT–DAE and DWT–DAE can perfectly retain the clean ECG signal. Because the level of noise is unknown during the measurement process, noise removal methods for practical applications should not destroy the ECG signal during low-level noise and should suppress the noise during high-level noise. Hence, the proposed DWT–DAE and DCT–DAE can more effectively handle noise at any moment in time.

REFERENCES

[1] WHO Global Health Estimates. (2020). *The Top 10 Causes of Death*. Accessed: Mar. 15, 2022. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death

[2] Y. Luo, R. H. Hargraves, A. Belle, O. Bai, X. Qi, K. R. Ward, M. P. Pfaffenberger, and K. Najarian, "A hierarchical method for removal of baseline drift from biomedical signals: Application in ECG analysis," *Sci. World J.*, vol. 2013, pp. 1–10, Apr. 2013, doi: 10.1155/2013/896056.

[3] V. de Pinto, "Filters for the reduction of baseline wander and muscle artifact in the ECG," *J. Electrocardiol.*, vol. 25, pp. 40–48, Jan. 1992, doi: 10.1016/0022-0736(92)90060-D.

[4] J. G. Webster, "Reducing motion artifacts and interference in biopotential recording," *IEEE Trans. Biomed. Eng.*, vol. BME-31, no. 12, pp. 823–826, Dec. 1984.

[5] J. C. Huhta and J. G. Webster, "60-Hz interference in electrocardiography," *IEEE Trans. Biomed. Eng.*, vol. BME-20, no. 2, pp. 91–101, Mar. 1973.

[6] M. Z. U. Rahman, R. A. Shaik, and D. V. R. K. Reddy, "Efficient and simplified adaptive noise cancelers for ECG sensor based remote health monitoring," *IEEE Sensors J.*, vol. 12, no. 3, pp. 566–573, Mar. 2012.

[7] Z. Islam, G. M. S. Sajjad, H. Rahman, A. Dey, M. Biswas, and A. Hoque, "Performance comparison of modified LMS and RLS algorithms in denoising of ECG signals," *Int. J. Eng. Technol.*, vol. 2, pp. 466–468, May 2012.

[8] I. Sharma, R. Mehra, and M. Singh, "Adaptive filter design for ECG noise reduction using LMS algorithm," in *Proc. 4th Int. Conf. Rel., Infocom Technol. Optim. (ICRITO)*, Sep. 2015, pp. 1–6, doi: 10.1109/ICRITO.2015.7359333.

[9] R. Shaddeli, N. Yazdanjue, S. Ebadollahi, M. M. Saberi, and B. Gill, "Noise removal from ECG signals by adaptive filter based on variable step size LMS using evolutionary algorithms," in *Proc. IEEE Can. Conf. Electr. Comput. Eng. (CCECE)*, Sep. 2021, pp. 1–7, doi: 10.1109/CCECE53047.2021.9569149.

[10] M. H. Moradi, M. A. Rad, and R. B. Khezerloo, "ECG signal enhancement using adaptive Kalman filter and signal averaging," *Int. J. Cardiol.*, vol. 173, no. 3, pp. 553–555, May 2014, doi: 10.1016/J.IJCARD.2014.03.128.

[11] H. D. Hesar and M. Mohebbi, "An adaptive Kalman filter bank for ECG denoising," *IEEE J. Biomed. Health Informat.*, vol. 25, no. 1, pp. 13–21, Jan. 2021, doi: 10.1109/JBHI.2020.2982935.

[12] V. Supraja and S. Safiya, "ECG de-noising using thresholding based on wavelet transforms," *Int. J. Eng. Res. Appl.*, vol. 3, no. 2, pp. 253–257, May 2013.

[13] S. K. Yadav, R. Sinha, and P. K. Bora, "Electrocardiogram signal denoising using non-local wavelet transform domain filtering," *IET Signal Process.*, vol. 9, no. 1, pp. 88–96, Feb. 2015.

[14] P. Divya and B. A. Devi, "Epileptic EEG signal denoising enhancement using improved threshold based wavelet method," in *Proc. Int. Conf. Syst., Comput., Autom. Netw. (ICSCAN)*, Jul. 2021, pp. 1–4, doi: 10.1109/ICSCAN53069.2021.9526343.

[15] B. Dai, D. Yang, and D. Feng, "Denoising ECG by a new wavelet threshold function," in *Proc. 14th Int. Congr. Image Signal Process., Biomed. Eng. Informat. (CISP-BMEI)*, Oct. 2021, pp. 1–6, doi: 10.1109/CISP-BMEI53629.2021.9624454.

[16] J. Lee, D. D. McManus, S. Merchant, and K. H. Chon, "Automatic motion and noise artifact detection in Holter ECG data using empirical mode decomposition and statistical approaches," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 6, pp. 1499–1506, Jun. 2012.

[17] G. Li, S. M. Wali Ullah, B. Li, J. Lin, and H. Wang, "Baseline wander removal for ECG signals based on improved EMD," in *Proc. 15th IEEE Int. Conf. Signal Process. (ICSP)*, Dec. 2020, pp. 484–487, doi: 10.1109/ICSP48669.2020.9320927.

[18] Y. Wang and D. Bai, "Application of wavelet threshold method based on optimized VMD to ECG denoising," in *Proc. IEEE 3rd Int. Conf. Frontiers Technol. Inf. Comput. (ICFTIC)*, Nov. 2021, pp. 741–744, doi: 10.1109/ICFTIC54370.2021.9647050.

[19] S. A. Malik, S. A. Parah, and B. A. Malik, "A VMD-SWT based ECG denoising technique," in *Proc. 6th Int. Conf. Image Inf. Process. (ICIIP)*, Nov. 2021, pp. 158–163, doi: 10.1109/ICIIP53038.2021.9702571.

[20] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Speech enhancement based on deep denoising autoencoder," in *Proc. Interspeech*, Aug. 2013, pp. 436–440.

[21] X.-J. Mao, C. Shen, and Y.-B. Yang, "Image restoration using convolutional auto-encoders with symmetric skip connections," 2016, *arXiv:1606.08921*.

[22] J. Zhou, Z. He, X. Liu, Y. Wang, S. Wang, and Q. Liu, "Transformed denoising autoencoder prior for image restoration," *J. Vis. Commun. Image Represent.*, vol. 72, Oct. 2020, Art. no. 102927, doi: 10.1016/J.JVCIR.2020.102927.

[23] C.-C. Chuang, C.-C. Lee, C.-H. Yeng, E.-C. So, B.-S. Lin, and Y.-J. Chen, "Convolutional denoising autoencoder based SSVEP signal enhancement to SSVEP-based BCIs," *Microsyst. Technol.*, vol. 28, no. 1, pp. 237–244, Oct. 2019, doi: 10.1007/S00542-019-04654-2.

[24] P. Xiong, H. Wang, M. Liu, S. Zhou, Z. Hou, and X. Liu, "ECG signal enhancement based on improved denoising auto-encoder," *Eng. Appl. Artif. Intell.*, vol. 52, pp. 194–202, Jun. 2016.

[25] L. El Bouny, M. Khalil, and A. Adib, "Convolutional denoising auto-encoder based AWGN removal from ECG signal," in *Proc. Int. Conf. Innov. Intell. Syst. Appl. (INISTA)*, Aug. 2021, pp. 1–6.

[26] P. Bing, W. Liu, and Z. Zhang, "DeepCEDNet: An efficient deep convolutional encoder–decoder networks for ECG signal enhancement," *IEEE Access*, vol. 9, pp. 56699–56708, 2021.

[27] Z. He, X. Liu, H. He, and H. Wang, "Dual attention convolutional neural network based on adaptive parametric ReLU for denoising ECG signals with strong noise," in *Proc. 43rd Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Nov. 2021, pp. 779–782.

[28] E. Dasan and I. Panneerselvam, "A novel dimensionality reduction approach for ECG signal via convolutional denoising autoencoder with LSTM," *Biomed. Signal Process. Control*, vol. 63, Jan. 2021, Art. no. 102225.

[29] 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.

[30] S. Nurmaini, A. Darmawahyuni, A. N. S. Mukti, M. N. Rachmatullah, F. Firdaus, and B. Tutuko, "Deep learning-based stacked denoising and autoencoder for ECG heartbeat classification," *Electronics*, vol. 9, no. 1, p. 135, Jan. 2020.

[31] P. Xiong, H. Wang, M. Liu, and X. Liu, "Denoising autoencoder for eletrocardiogram signal enhancement," *J. Med. Imaging Health Inform.*, vol. 5, no. 8, pp. 1804–1810, Dec. 2015.

[32] P. Singh and G. Pradhan, "A new ECG denoising framework using generative adversarial network," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 18, no. 2, pp. 759–764, Mar. 2021, doi: 10.1109/TCBB.2020.2976981.

[33] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, 2008, pp. 1096–1103.

[34] T. Liu, J. Wang, Q. Liu, S. Alibhai, and T. Lu, "High-ratio lossy compression: Exploring the autoencoder to compress scientific data," *IEEE Trans. Big Data*, early access, Mar. 17, 2021, doi: 10.1109/TBDATA.2021.3066151.

[35] J. T. Olkkonen, *Discrete Wavelet Transforms: Theory and Applications*. Rijeka, Croatia: STeP Ri, Apr. 2011, doi: 10.5772/649.

[36] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Comput.*, vol. C-23, no. 1, pp. 90–93, Jan. 1974, doi: 10.1109/T-C.1974.223784.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[38] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 6, no. 2, pp. 107–116, Jun. 1998, doi: 10.1142/S0218488598000094.

[39] G. Moody and R. Mark. *MIT-BIH Noise Stress Test Database v1.0.0*. Accessed: Apr. 1, 2022. [Online]. Available: https://physionet.org/content/nstdb/1.0.0/

[40] G. Moody and R. Mark. *MIT-BIH Normal Sinus Rhythm Database v1.0.0*. Accessed: Apr. 1, 2022. [Online]. Available: https://physionet.org/content/nsrdb/1.0.0/

[41] G. Moody and R. Mark. *MIT-BIH Arrhythmia Database v1.0.0*. Accessed: Apr. 1, 2022. [Online]. Available: https://physionet.org/content/mitdb/1.0.0/

[42] A. Němcová, R. Smíšek, L. Maršánová, L. Smital, and M. Vítek, "A comparative analysis of methods for evaluation of ECG signal quality after compression," *BioMed Res. Int.*, vol. 2018, pp. 1–26, Jul. 2018.

MING-HWA SHEU (Member, IEEE) received the M.S. and Ph.D. degrees in electrical engineering from the National Cheng Kung University, Tainan, Taiwan, in 1989 and 1993, respectively. He is currently a Full Professor with the Department of Electronic Engineering, National Yunlin University of Science & Technology, Yunlin, Taiwan. From 2015 to 2018, he has worked as a Supervisor of Taiwan IC Design Association. He has served as a Committee Chair of E.E. course planning for the Technical High School, Ministry of Education, Taiwan. He has worked as a Review Committee of the Engineering Department, Ministry of Science & Technology (MOST). From 2008 to 2011, he served as the Chairperson of the Department of Electronic Engineering. His research interests include CAD/VLSI, digital signal process, algorithm analysis, and embedded systems.

YU-SYUAN JHANG was born in Keelung, Taiwan, in 1993. He received the B.S. degree from the Department of Computer Science and Information Engineering, Nanhua University, Chiayi, Taiwan, in 2014, and the M.S. degree from the Institute of Medical Science and Technology, National Sun Yat-sen University, Kaohsiung, Taiwan. He is currently pursuing the Ph.D. degree with the Department of Electronic Engineering, National Yunlin University of Science & Technology, Yunlin, Taiwan. His research interests include digital signal processing, embedded system application, FPGA application design, android application design, deep learning, and adaptive filter algorithm.

YEN-CHING CHANG was born in Changhua, Taiwan, in 1997. She received the B.S. degree from the Department of Computer Science and Information Engineering, Nanhua University, Chiayi, Taiwan, in 2019, and the M.S. degree from the Department of Green Technology for Sustainability, Nanhua University, in 2020. She is currently pursuing the Ph.D. degree in electronic engineering with the National Yunlin University of Science & Technology, Yunlin, Taiwan. Her research interests include digital signal processing and data compression.

SZU-TING WANG received the B.S. degree from the Department of Computer Science and Information Management, Providence University, Taichung, Taiwan, and the M.S. degree from the Department of Information Engineering and Computer Science, Feng Chia University, Taichung, Taiwan. She is currently pursuing the Ph.D. degree in smart industry technology research and design with the National Formosa University, Yunlin, Taiwan. Her research interests include image processing, digital signal processing, and deep learning.

CHUAN-YU CHANG (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the National Cheng Kung University, Tainan, Taiwan, in 2000. From 2009 to 2011, he was the Chair of the Department of Computer Science and Information Engineering, National Yunlin University of Science & Technology (YunTech), Taiwan, where he was the Dean of the Research and Development and the Director of the Incubation Center for Academia-Industry Collaboration and Intellectual Property, from 2011 to 2019. He is currently the Deputy General Director of the Service Systems Technology Center, Industrial Technology Research Institute, Taiwan. He is also a Distinguished Professor with the Department of Computer Science and Information Engineering, YunTech. He has authored or coauthored more than 200 publications in journals and conference proceedings in his research fields. His research interests include computational intelligence and their applications to medical image processing, automated optical inspection, emotion recognition, and pattern recognition. He is a fellow of IET and a Life Member of IPPR and TAAI. He was the Program Co-Chair of TAAI 2007, CVGIP 2009, (2010–2019) International Workshop on Intelligent Sensors and Smart Environments, and the Third International Conference on Robot, Vision and Signal Processing (RVSP 2015). He was the General Co-Chair of 2012 International Conference on Information Security and Intelligent Control, from 2011 to 2013, Workshop on Digital Life Technologies, CVGIP 2017, WIC 2018, ICS 2018, and WIC 2019. From 2015 to 2017, he was the Chair of the IEEE Signal Processing Society Tainan Chapter and the Representative for Region 10 of IEEE SPS Chapters Committee. He received the National Award for Distinguished Contribution to Industry-Academia Cooperation from Ministry of Education and the Outstanding Electrical Engineering Professor Award from the Chinese Institute of Electrical Engineering, Taiwan, in 2021. He is also the President of the Chinese Image Processing and Pattern Recognition Society.

SHIN-CHI LAI (Member, IEEE) was born in Taichung, Taiwan, in 1980. He received the B.S. degree in electronic engineering from Chienkuo Technology University, Changhua, Taiwan, in 2002, the M.S. degree in electronic engineering from the National Yunlin University of Science & Technology, Yunlin, Taiwan, in 2005, and the Ph.D. degree from the National Cheng Kung University, Tainan, Taiwan, in 2011. From October 2011 to July 2013, he had been an Assistant Research Fellow of the Department of Electrical Engineering, National Cheng Kung University, Tainan, Taiwan. From August 2013 to July 2016, he had been an Assistant Professor at the Department of Computer Science and Information Engineering, Nanhua University. From August 2016 to July 2019, he was an Associate Professor with the Department of Computer Science and Information Engineering. From August 2019 to July 2021, he had been a Full Professor of the Department of Computer Science and Information Engineering, Nanhua University, Chiayi, Taiwan. He is currently a Professor with the Department of Automation Engineering, National Formosa University. His research interests include signal processing and its circuit design, especially for speech, audio, biomedical, and multimedia applications.