

An ECG Signal Denoising Method Using Conditional Generative Adversarial Net

Xiaoyu Wang , Bingchu Chen, Ming Zeng , Yuli Wang, Hui Liu , Ruixia Liu , Member, IEEE,
Lan Tian , and Xiaoshan Lu

Abstract—In this paper, a novel denoising method for electrocardiogram (ECG) signal is proposed to improve performance and availability under multiple noise cases. The method is based on the framework of conditional generative adversarial network (CGAN), and we improved the CGAN framework for ECG denoising. The proposed framework consists of two networks: a generator that is composed of the optimized convolutional auto-encoder (CAE) and a discriminator that is composed of four convolution layers and one full connection layer. As the convolutional layers of CAE can preserve spatial locality and the neighborhood relations in the latent higher-level feature representations of ECG signal, and the skip connection facilitates the gradient propagation in the denoising training process, the trained denoising model has good performance and generalization ability. The extensive experimental results on MIT-BIH databases show that for single noise and mixed noises, the average signal-to-noise ratio (SNR) of denoised ECG signal is above 39 dB, and it is better than that of the state-of-the-art methods. Furthermore, the denoised classification results of four cardiac diseases show that the average accuracy increased above 32% under multiple noises under SNR=0 dB. So, the proposed method can remove noise effectively as well as keep the details of the features of ECG signals.

Index Terms—Convolutional auto-encoder (CAE), generative adversarial network (GAN), ECG signal denoising.

I. INTRODUCTION

LECTROCARDIOGRAM (ECG) plays an irreplaceable role in the recognition [1], diagnosis, and classification [2] of cardiac diseases (e.g., atrial fibrillation [3], premature ventricular, and atrial beats, etc.). With the development of telemedicine, remote ECG monitoring provides important auxiliary information for automatic diagnosis of cardiac diseases [4]. However, ECG signals are often polluted by plenty of noises from electrode motion (EM), baseline wander (BW), and muscle artifacts (MA) [5], thus the recognition and diagnosis accuracy of ECG must be affected [6]. For this reason, noise interference should be removed from noisy ECG signals.

There are plenty of traditional denoising methods for ECG signals, for instance, adaptive filtering, Empirical Mode Decomposition (EMD), S-Transform (ST), Wavelet Transform (WT), and Fourier decomposition. Rahman *et al.* [7] denoised remote ECG signals with a number of adaptive recurrent filters. However, adaptive filtering often required reference noise as the input signal, which was hard to obtain in ECG signal acquisition system. Kabir *et al.* [8] removed the noise from the initial intrinsic mode functions (IMFs) through the windowing method integrated with EMD. This method preserved the QRS complex and produced the comparatively pure ECG signals. The EMD method used the Hilbert transform, but the Hilbert transform has no ability to separate the signals whose frequency is similar, so it is easy to accidentally filter P-waves and T-waves of ECG signals. ST was used in [9] to represent the noisy ECG signals. Then, masking and filtering techniques were used to reduce the noise in time-frequency domain. However, the spectra of MA noise and ECG signals are overlapping. So T-wave is attenuated and QRS peak becomes smaller. Gokhale *et al.* [10] proposed a discrete WT (DWT) method for removing 50 Hz power line interference (PLI) noise. And Smital *et al.* [11] proposed an adaptive wavelet Wiener filtering method for reducing broadband electromyographic (EMG) noise in ECG signals. However, the ST-wave and BW noise as well as the QRS wave group and MA noise are difficult to distinguish based on frequency domain features completely. Adaptive Fourier Decomposition (AFD) was used in [12] for denoising. This method decomposed the

Manuscript received April 29, 2021; revised October 31, 2021, January 12, 2022, and March 8, 2022; accepted April 18, 2022. Date of publication April 21, 2022; date of current version July 4, 2022. This work was supported in part by the Natural Science Foundation of Shandong Province under Grants ZR2021ZD40 and ZR2021MF065, in part by the Research Project for Graduate Education and Teaching Reform, Shandong University, China under Grant XYJG2020108, and in part by the Innovation Pilot Project for Integration of Science, Education and Industry, Qilu University of Technology (Shandong Academy of Sciences), China under Grant 2020KJC-ZD03. (Xiaoyu Wang, Bingchu Chen, and Ming Zeng are co-first authors.) (Corresponding authors: Lan Tian; Xiaoshan Lu.)

Xiaoyu Wang, Bingchu Chen, Ming Zeng, and Lan Tian are with the School of Microelectronics, Shandong University, Jinan 250100, China, and also with the Shandong Artificial Intelligence Institute, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250353, China (e-mail: 201912234@mail.sdu.edu.cn; cbc9409@foxmail.com; zengming@sdu.edu.cn; tianlansdu@foxmail.com).

Xiaoshan Lu is with the School of Information Science and Engineering, Shandong University, Qingdao 266237, China (e-mail: luxiaoshan@126.com).

Yuli Wang, Hui Liu, and Ruixia Liu are with the Shandong Artificial Intelligence Institute, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250353, China (e-mail: wyuli@sdas.org; liuhui@sdas.org; liurx@sdas.org).

Digital Object Identifier 10.1109/JBHI.2022.3169325

signal in accordance with the energy distribution and separated the ECG signals and noise which have the same frequency range but different energy distribution. However, noise with the same energy distribution cannot be removed. To sum up, in the above traditional denoising methods based on different transformations, it is difficult to distinguish ECG signals from noise, when the spectra or energy distribution of noise and ECG signals are overlapping. Therefore, they cannot effectively remove noise in ECG signals under the above conditions.

Cardiac monitoring system and a large amount of telemedicine data provide database for ECG researches, and the denoising methods using deep learning have also been studied. Suranai *et al.* [13] proposed a denoising method based on wavelet neural network (WNN), which integrated the adaptive learning ability of the neural network and the multi-resolution feature of the wavelet, but it only removed high-frequency noise. In his later work [14], he proposed an adaptive filtering method for ECG signal denoising based on DWT and neural network. This method can remove EM, MA, BW, and mixed noises. The stacked contractive denoising auto-encoder (CDAE) proposed by Peng *et al.* [15] and its improved version [16] denoising auto-encoder (DAE) can also remove EM, MA, BW, and mixed noises in ECG signals. The DAE can capture as much information as possible in each given sample, when the sample is simply corrupting the input distribution. Therefore, the more similar the waveform of the training sample to the test sample, the better the denoising effect. So, this method relies on sample selection to achieve a better denoising effect.

However, the learning method based on generative adversarial networks (GAN) is different from that of DAE. It uses adversarial training mode to update generator (G) to generate data that meets the requirements of learning purposes. In this way, G can automatically learn the complex distribution of raw ECG signals. GAN was introduced by Ian Goodfellow *et al.* in [17] for generating samples via an adversarial process. They trained two sub-networks: a generator G to generate a realistic sample, and a discriminator D to estimate the probability that a sample came from the real sample rather than that of G. The network corresponded to a minimax two-player game. Wang *et al.* [18] proposed an ECG denoising method based on an adversarial method. They used a GAN model to remove single noise and mixed noise effectively and this method achieved the highest SNR of denoised signal. But we think that this structure may achieve a better denoising effect after adding condition to direct the data generation process. In 2020, Pratik *et al.* [19] proposed convolutional neural network (CNN) based on GAN model, which can carry out end-to-end noise reduction for EM, BW, and MA noises. However, the noise reduction effect of mixed noise is not studied, and the influence of the input variables z on the denoising task was not explored. In 2021, Xu *et al.* [20] used a combination of GAN and residual network for ECG denoising. Their denoising results were about 29 dB, 32 dB, and 60 dB when SNR=0 dB, 1.25 dB, and 5 dB, respectively. We found this method is not ideal at low SNR. In the above methods based on GAN, conditional generative adversarial network (CGAN) is not explored for ECG denoising.

CGAN proposed by Mirza *et al.* [21] in 2014 and it proposed a method of conditioning additional information to control data generation process in a supervised manner. CGAN has witnessed great progress in the field of image processing, such as style transfer [22], super-resolution [23], and image inpainting [24]. It is worth mentioning that CGAN is also used in image enhancement tasks, for instance, image de-raining [25], image dehazing [26], and infrared image enhancement [27]. Qin *et al.* [28] proposed a model based on a one-dimensional CNN with CGAN. CGAN was used for sample augmentation of plant electrical signals and then one-dimensional CNN was used for classification. Inspired by this, we apply the CGAN to our ECG denoising task. The convolutional auto-encoder (CAE) is also adopted in our method.

Based on the above analysis, we propose a new CGAN based on CAE (i.e., CAE-CGAN). The CAE-CGAN is a GAN-based framework, which is composed of a generator and a discriminator. The generator is constructed with optimized CAE, which is used to generate the denoised ECG signals. And the discriminator is an auxiliary network that helps the generator to produce denoised ECG signals that are more similar to the raw signals. Once the training is done, the noisy ECG signals can be denoised by forward propagation of the trained generator. And our experimental results show that the proposed method can obviously improve ECG denoising performance.

To sum up, the contributions of this paper are as follows:

- An end-to-end ECG denoising method based on CAE-CGAN is proposed. We conducted a comparative experimental study on how CGAN is used in ECG noise reduction task, and obtained the optimized CGAN structure.
- The CAE structure is applied as the generator of CGAN framework. The optimized CAE structure is determined by optimizing the parameters and designing the structure. The experimental results show that the optimized structure has better noise reduction effect than other structures.
- Extensive experiments are conducted on the MITBIH datasets, compared with the existing methods, our performance metrics are significantly improved for various cases of single noise and mixed noise. And for multiple noises cases under SNR = 0 dB, the average accuracy of the denoised classification results for four cardiac diseases increased above 32%.
- The denoising results for new records and new lead show that our method has good generalization ability.

The organization of this paper is as follows. The related knowledge of GAN is introduced in Section II. The details of the proposed method are presented in Section III. Section IV analyzes the experiments results, and Section V gives a conclusion of whole work.

II. THE RELATED KNOWLEDGE OF GAN

GAN is a generative model proposed by Ian Goodfellow [17]. It uses adversarial training mode to update generator (G) to generate data that meets the requirements of specific purposes. Discriminator (D) is usually a binary classifier, whose task is to

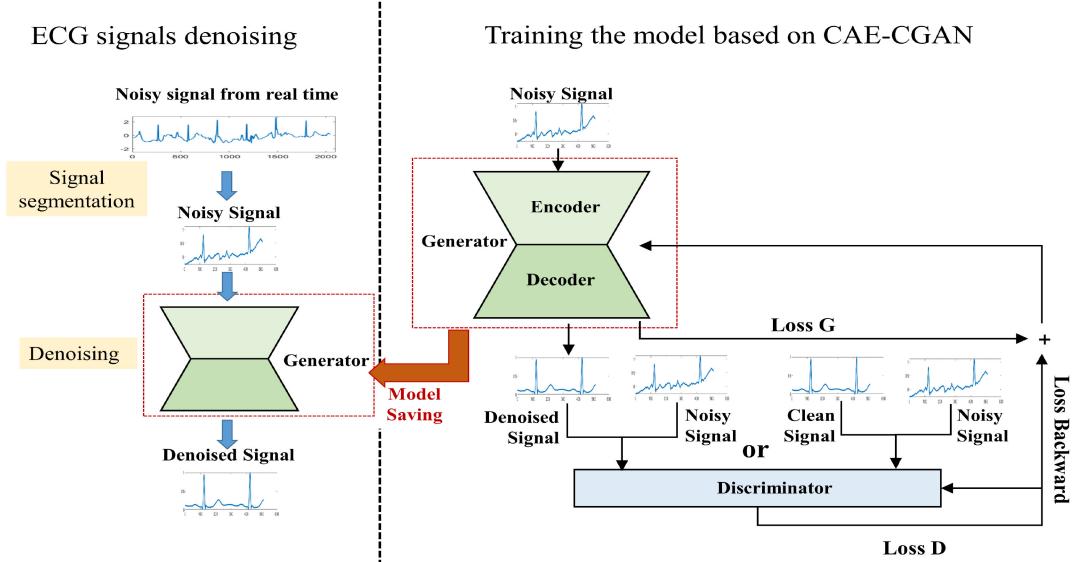


Fig. 1. The overall structure of CAE-CGAN for ECG denoising.

distinguish the true sample of the training data from the false sample generated by G . G learns the training data distribution and simulates the training data to generate false samples. G is committed to generating more vivid sample data to fool D . This adversarial learning process is modeled as a minimax game between G and D . That is, the following objective function is optimized:

$$\min_G \max_D = \mathbb{E}_{x \sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))] \quad (1)$$

where $p_{data}(x)$ is the distribution of the real data x , $p_z(z)$ is the prior distribution of input variables z .

After the emergence of GAN, a variety of improved versions have been proposed, among which one of the most important is the CGAN proposed by Mirza *et al.* [21]. GAN can be improved to a conditional version by adding additional condition information y . y can be any additional information like class labels or data from other distributions. The objective function of CGAN is:

$$\min_G \max_D = \mathbb{E}_{x \sim p_{data}(x)}[\log D(x, y)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z, y), y))] \quad (2)$$

In the denoising task, y is set to the noisy signal \tilde{x} .

To overcome disadvantages like gradient vanishing, the least squares GAN (LSGAN) method [29] replaces the cross-entropy loss with the least square loss. The LSGAN's objective functions of D and G are as follows:

$$\begin{aligned} \min_D V_{LSGAN}(D) &= \frac{1}{2} \mathbb{E}_{x \sim p_{data}(x)}[(D(x, y) - 1)^2] \\ &+ \frac{1}{2} \mathbb{E}_{z \sim p_z(z)}[(D(G(z, y), y))^2] \end{aligned} \quad (3)$$

$$\min_G V_{LSGAN}(G) = \frac{1}{2} \mathbb{E}_{z \sim p_z(z)}[(D(G(z, y), y) - 1)^2] \quad (4)$$

III. PROPOSED METHOD

In this section, the architecture of the proposed network including the generator and the discriminator, and loss function are analyzed, respectively.

A. The Overall Structure

The overall architecture of CAE-GAN is shown in Fig. 1. The left side shows the noise reduction process of the ECG signal, and the right side shows the training process of the model. On the right, the proposed CAE-CGAN contains a generator G and a discriminator D . We adopt convolutional auto-encoder (CAE) to build a generator that contains an encoding stage and a decoding stage. Its input is the noisy ECG signal \tilde{x} , and its output is the denoised signal \hat{x} . The discriminator receives a pair of signals as input, which is composed of raw data and noisy data (x, \tilde{x}) or denoised data and noisy data (\hat{x}, \tilde{x}). The discriminator updates the D 's parameters through the discriminator loss function (loss D) formed by its own output. The generator updates the G 's parameters through the generator loss function (Loss G) composed of its own output and the discriminator's output. When the model training is finished, save the generator model. On the left side of Fig. 1, we slice the noisy signals into one sample which has M sampling points and send them to the saved generator model for denoising. Then the denoised signal can be output.

The adversarial training process of G and D is shown in Fig. 2. The training process is divided into two steps. First, D should be trained. We input (x, \tilde{x}) and (\hat{x}, \tilde{x}) into D , and then feedback their output results to the D through the loss function to complete the updating of D 's parameters. During this process, the parameters of G are frozen, as shown in Fig. 2(a) and (b). Second, the generator should be trained. We fix the parameters of D , input (\hat{x}, \tilde{x}) into D and feed the result back to G through the loss function. Then G parameters are updated, as shown in Fig. 2(c). G and D are trained alternatively until Nash equilibrium is reached [30], and the training is completed.

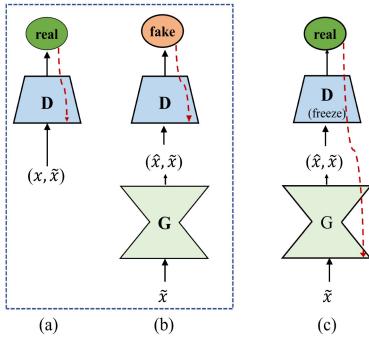


Fig. 2. Adversarial training for ECG denoising. Dashed lines represent the flow of gradient back-prop.

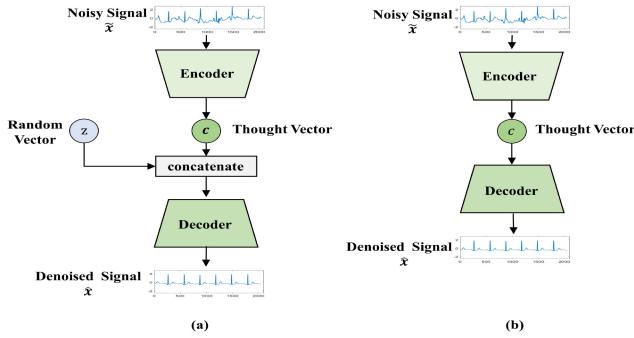


Fig. 3. Two generator models. (a) generator with z . (b) generator without z .

B. Generator and Its Loss Function

We adopt convolutional auto-encoder (CAE) to build a generator. Convolutional layers can preserve spatial locality and the neighborhood relations of the input in their potential higher-level feature representations [31].

GANs are generative models that learn a mapping from input variables z to real data x ($G : z \Rightarrow x$). In contrast, conditional GANs learn a mapping from condition y and input variables z to real data x ($G : \{z, y\} \Rightarrow x$). In [32] they used CGAN with input variables z for image-to-image translation. Correspondingly, in the image enhancement task [27], the input variables z was removed. In order to explore whether the input variables z is needed for the ECG denoising task based on CGAN, we design two models. The first model is shown in Fig. 3(a). In the encoding process, the features of the input data are extracted and compressed through the convolutional layer. The compressed features are called the thought vector c , which is connected to the input variables z . The second model, as shown in Fig. 3(b), removes the input variables z .

Experiments are carried out in Section IV(B) to compare the denoising performance of the two models in Fig. 3(a) and Fig. 3(b). The experimental results confirm that the module without input variables z can not only improve the denoising performance but also retain more waveform details. Therefore, we selected the model without input variables z .

The network architecture of the generator is shown in Fig. 4. The encoder is composed of 7 convolutional layers. We conduct an optimization experiment on dimensions (see IV.C for details),

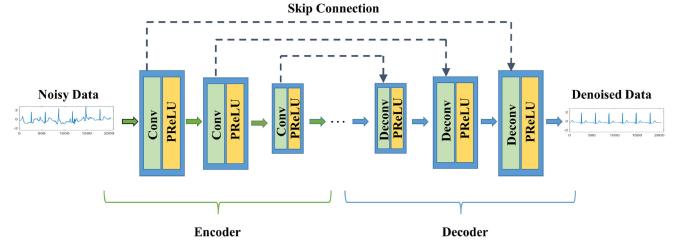


Fig. 4. Network architecture for generator of CAE-CGAN.

the resulting dimensions for each layer are 512×1 , 512×16 , 512×32 , 512×64 , 256×128 , 128×256 , 64×512 , and 32×1024 . The decoder is a mirroring of the encoder, and their corresponding layers have the same filter widths and numbers. The PReLU activation function layer is set between each convolution layer. In addition, in order to facilitate gradient propagation, we introduce skip connection between each coding layer and corresponding decoding layer. The pooling layer will lose details, so there is no pooling layer in the generator.

Based on LSGAN objective function, the input variables z is removed, and the distance function l_{dist} and the maximum local difference l_{max} [18] are added. l_{dist} is used to measure the differences between the denoised signal \hat{x} and the raw signal x . The smaller l_{dist} , the better the quality of the denoised signal. l_{max} keeps track of the maximum difference between the denoised and raw signals. The smaller l_{max} is, the more details of the ECG signal are preserved, and the greater the medical value of the denoised signal is. Calculations of l_{dist} and l_{max} are as follows:

$$l_{dist} = \sqrt{\sum_{n=1}^N |\hat{x}_n - x_n|} \quad (5)$$

$$l_{max} = \max(|\hat{x}_1 - x_1|, |\hat{x}_2 - x_2|, \dots, |\hat{x}_N - x_N|) \quad (6)$$

where N represents the number of samples; \hat{x}_n represents the n-th sample of denoised signals; x_n represents the n-th sample of raw signals.

The loss function of G is as follow:

$$\min_G V(G) = \mathbb{E}_{\tilde{x} \sim p_{noisy}(\tilde{x})} [(D(G(\tilde{x}), \tilde{x})) - 1]^2 + \lambda_1 l_{dist} + \lambda_2 l_{max} \quad (7)$$

where $p_{noisy}(\tilde{x})$ is the distribution of the noisy data \tilde{x} . λ_1 and λ_2 are weight coefficients used to adjust the weights of l_{dist} and l_{max} in the objective function, which are set to 0.7 and 0.2 respectively through experiments.

C. Discriminator and Its Loss Function

The discriminator has two-channel input. In other words, the discriminator receives a pair of signals as input. This pair of signals may be a pair of raw signal and noisy signal (x, \tilde{x}) or a pair of denoised signal and noisy signal (\hat{x}, \tilde{x}). The noisy signal in the input is the additional conditional information used to direct the data generation process. As shown in Fig. 5, the

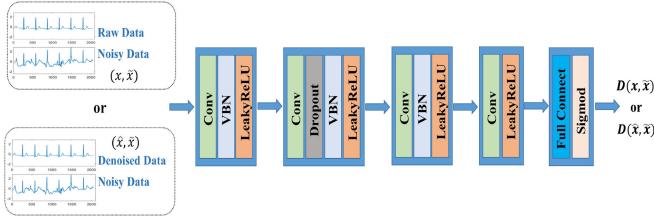


Fig. 5. Network architecture for discriminator of CAE-CGAN.

discriminator uses four convolution layers and one full connection layer, and each convolution layer is followed by the virtual batch normalization (VBN) layer and the LeakyReLU activation function layer. After the final full connection layer, the sigmoid function outputs a probability value between 0 and 1.

At the beginning of training, the discriminator is trained, as shown in Fig. 2(a) and (b). The loss function of the discriminator is used to update the discriminator so that the discriminator considers $(\mathbf{x}, \tilde{\mathbf{x}})$ as true and $(\tilde{\mathbf{x}}, \tilde{\mathbf{x}})$ as false.

The modified loss function is as follows:

$$\begin{aligned} \min_D V(D) = & \mathbb{E}_{\mathbf{x} \sim p_{data}(\mathbf{x})} [(D(\mathbf{x}, \tilde{\mathbf{x}}) - 1)^2] \\ & + \mathbb{E}_{\tilde{\mathbf{x}} \sim p_{data}(\mathbf{x})} [D(G(\tilde{\mathbf{x}}), \tilde{\mathbf{x}})^2] \end{aligned} \quad (8)$$

IV. EXPERIMENTAL RESULTS AND DISCUSSION

A. Experimental Setup

1) Performance Metrics: Root mean square error (RMSE) and signal-to-noise ratio (SNR) are used as performance metrics, and the formulas are as follows:

$$RMSE = \sqrt{\frac{1}{N} \sum_{n=1}^N (\hat{\mathbf{x}}_n - \mathbf{x}_n)^2} \quad (9)$$

$$SNR = 10 \log_{10} \frac{\sum_{n=1}^N \mathbf{x}_n^2}{\sum_{n=1}^N (\hat{\mathbf{x}}_n - \mathbf{x}_n)^2} \quad (10)$$

where \mathbf{x} represents the raw signal, $\hat{\mathbf{x}}$ represents denoised signal.

These metrics are the key evaluation parameters of denoising methods. The higher the SNR and the smaller the RMSE, the better the denoising effect.

2) Database and Experimental Samples: Raw data are obtained from MIT-BIH Arrhythmia Database [33]. Each record lasts 30 minutes and it is sampled at a frequency of 360 Hz. Noise data are obtained from MIT-BIH Noise Stress Test Database [34]. BW, EM, and MA noises records are used. The following two types of data sets are used in this article.

(I) Modeling Experimental data set

In order to compare the proposed method with Improved DAE [16] and Adversarial Method [18], raw data are obtained from the same 10 records numbered 103, 105, 111, 116, 122, 205, 213, 219, 223, and 230 in the MIT-BIH Arrhythmia Database [33], these records data is from modified limb lead II (MLII).

(II) Generalization data set (See section IV.F for more details)

In order to explore the generalization ability of this method for different test data, we conduct experiments on the following two new data sets.

(a) New records data set

Another 10 new records are selected in experiments to explore the effects of the model on different data sets. The 10 new records are 100, 101, 106, 112, 117, 121, 123, 209, 220, and 228. These records data is also from MLII.

(b) New lead data set

Lead V1 is not used in previous experiments, so it is selected to explore the effects of the model on different lead data. Since record 103 does not have lead V1, we chose record 101 instead. The other nine records are the same. So, 10 records are 101, 105, 111, 116, 122, 205, 213, 219, 223, and 230.

For the above data sets, 80% were used for training and 20% for testing respectively.

The experimental samples are constructed according to the following steps:

(1) Referring to reference [18], the noisy signals used in our experiments are as follows:

$$\tilde{\mathbf{x}}_n = \mathbf{x}_n + \mathbf{n}_n \quad (11)$$

where, $\tilde{\mathbf{x}}_n$ represents the noisy signals, \mathbf{x}_n represents the raw signal in the MIT-BIH Arrhythmia Database, \mathbf{n}_n represents the noise. The SNR of noisy signals $\tilde{\mathbf{x}}$ is calculated as follows:

$$SNR = 10 \log_{10} \frac{\sum_{n=1}^N \mathbf{x}_n^2}{\sum_{n=1}^N \mathbf{n}_n^2} \quad (12)$$

When SNR is of different values, such as 0, 1, 2, 3, 4, and 5 dB, different noisy signals can be calculated.

(2) In [15], [16], and [18], they need to use one heartbeat as a training sample through locating the position of the R peak. But accurate R peak detection is difficult in the ECG signal with strong noise. Therefore, we propose a more practical way for slicing ECG data. A sample is segmented by certain number of sampling points as follows.

Assume that the sampling frequency of the database is F_s , the number of the sample points is M , and the heartbeat period is T . Generally, the heartbeat rate is about 50 to 120 beats per minute, so the heartbeat period is about 0.5 to 1.2 seconds. In order to ensure that each sample (M) covers at least one heartbeat (T) and the dimensions of the denoising model is not too large, we select the suitable length of sample according to the following formula.

$$M \geq F_s * T_{max} \quad (13)$$

Here, F_s is 360 Hz of the MIT-BIH database, T_{max} is 1.2 seconds, M should be bigger than 432. In order to facilitate the establishment of G network, we set M is 512.

(3) In order to balance the signal strength in each sample, accelerate the convergence speed and improve the stability of the model, the minimum-maximum normalization is as follows:

$$Normalized(\mathbf{x}_n) = \frac{\mathbf{x}_n - x_{min}}{x_{max} - x_{min}} \quad (14)$$

where x_{max} and x_{min} represent the maximum and minimum values in each sample, respectively.

3) The Computing Platform: We conduct all the training tests and validation of the experiments on an NVIDIA Tesla V100 server. And the codes are implemented in python 3.6 and PyTorch 1.1.

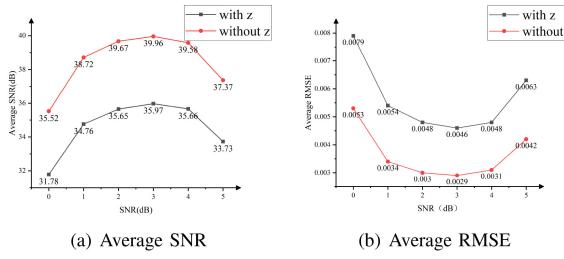


Fig. 6. Comparison of denoising results of models with z and without z .

The network parameters of both G and D are optimized by using the RMSprop optimizer. And the optimizer has a learning rate of 0.0001. The batch size is 256.

B. Model Comparison With and Without the Input Variables z

The input variables z has been retained in reference [19] and reference [21]. In order to further study the impact of the input variables z in ECG denoising task, we conduct a model comparison experiments. The generator model with z is shown in Fig. 3(a). We add the input variables z input between the encoder and the decoder. The input variables z is directly connected with the thought vector c , and the concatenated vector is decoded to obtain the denoised signal. As shown in Fig. 3(b), the generator model without z directly compresses the noisy signals into the thought vector c at the encoding stage, and then decodes it to the denoised signals at the decoding stage.

We select the EM noise and add it into the raw data by setting different SNRs (0 dB, 1 dB, 2 dB, 3 dB, 4 dB, and 5 dB) to get the noisy data. Then, we randomly select 80% of the data for training and the remaining data for testing. There are 61020 samples in training set and 15120 samples in test set.

Fig. 6 intuitively compares the average SNR and the average RMSE of the two models. It is clear that the model without z has higher SNR and smaller RMSE for each record under different SNR conditions. By removing the input variables z , the average SNR is increased by about 4 dB, and the average RMSE is decreased by about 0.002. Thus, it implies that removing input variables z can reach better performance in terms of SNR and RMSE.

As Fig. 7(e) and (f) show, both models can generate denoised signals close to the raw signals, indicating that both models can effectively remove noise. By comparing the waveforms in Fig. 7(g) and (h), it is confirmed that the model without the input variables z retains more details of the waveforms, and makes the denoised signals have higher medical value. Therefore, the input variables z of CGAN should be removed in the ECG denoising task.

C. Dimensions Tuning

In order to explore the influence of dimensions of G on denoising performance, we build five models of various dimensions with the same number of layers. As shown in Fig. 8, the dimensions of the five models are all the same after the first

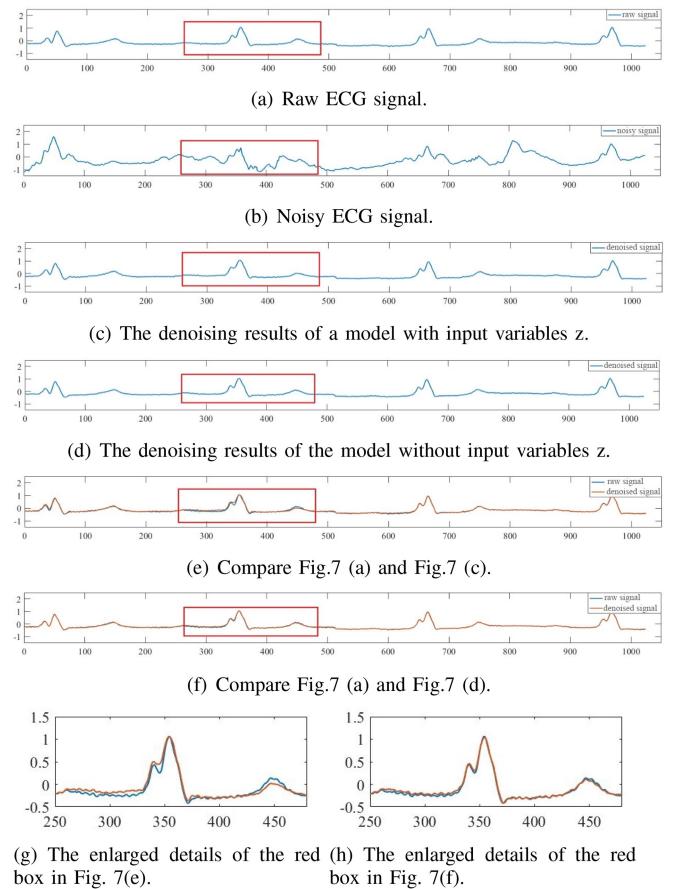


Fig. 7. Comparison of denoising waveforms using different models.

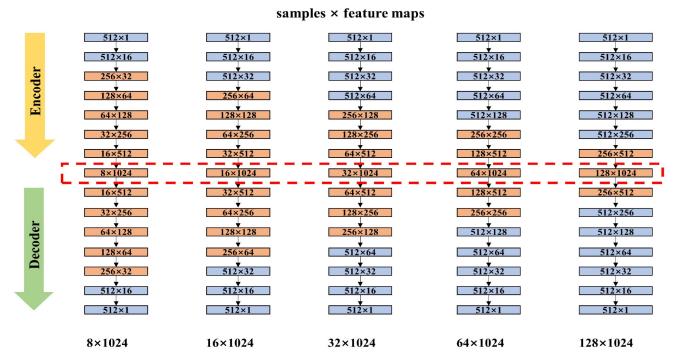


Fig. 8. Generator models of different coding sizes (dimensions).

convolutional layer, and the feature maps double from the second layer to 1024 layer by layer. In terms of the change in samples, samples of the five models are halved layer by layer from the second, third, fourth, fifth, and sixth layers, respectively. As is shown in the red box in Fig. 8, after 7 convolutional layers, the five models get the dimensions of 8×1024 , 16×1024 , 32×1024 , 64×1024 , and 128×1024 in the form of samples × feature maps, respectively. We use the dimensions of this layer to represent different models.

Fig. 9 shows the denoising results of five models with different dimensions. It can be seen from Fig. 9 that when the dimensions are set to 32×1024 , SNR reaches the maximum and RMSE

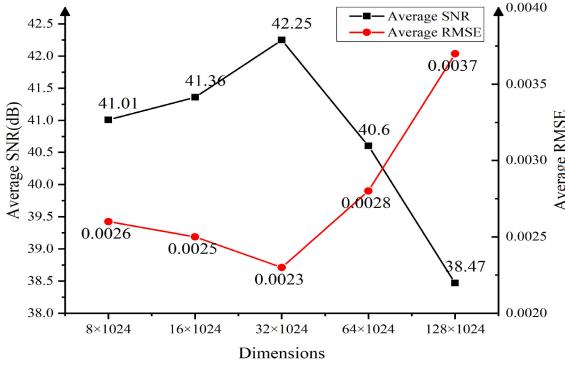


Fig. 9. Denoising results of five models with different dimensions.

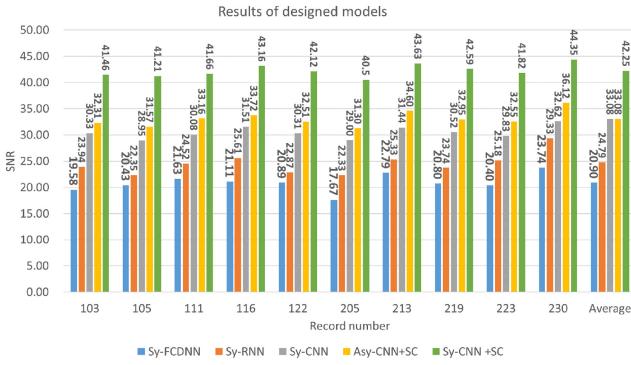


Fig. 10. Results of different model structures.

reaches the minimum. To sum up, the resulting dimensions for each layer are 512×1 , 512×16 , 512×32 , 512×64 , 256×128 , 128×256 , 64×512 , and 32×1024 . The decoder of G is a mirroring of the encoder, and they have the same width and number of filters for the corresponding layer.

D. Structure Designing

To determine the appropriate generator structure, we chose different structures to try to build the network. We defined symmetrical coding means the dimensions of an encoder are the same as that of the decoder. We design the following five model structures. (1) symmetrical coding with fully connected deep neural networks (Sy-FCDNN) (2) symmetrical coding with recurrent neural networks (Sy-RNN) (3) symmetrical coding with convolutional neural network (Sy-CNN) (4) asymmetrical coding with convolutional neural network and skip connection (Asy-CNN+SC) (5) symmetrical coding with convolutional neural network and skip connection (Sy-CNN+SC).

As shown in Fig. 10, by comparing the results of Sy-FCDNN, Sy-RNN and Sy-CNN, we found the Sy-CNN structure has a better effect by adopting the convolutional auto-encoder. By comparing the results of Sy-CNN and Sy-CNN+SC, we found that better result can be obtained by introducing skip connection between each coding layer and corresponding decoding layer, and skip connection facilitates gradient propagation. By comparing Asy-CNN+SC and Sy-CNN+SC, the result of the symmetrical encoder is better. Sy-CNN+SC is the best of the

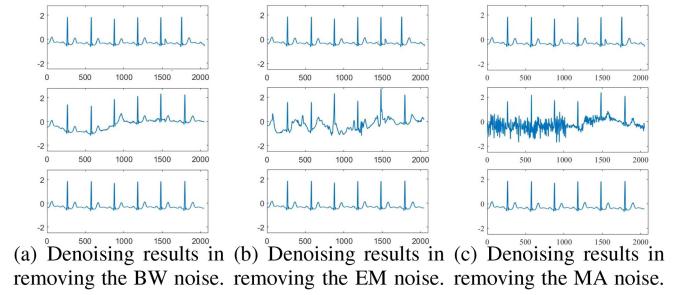


Fig. 11. Denoising results in removing the single noise.

five methods. Finally, we adopted Sy-CNN+SC as the generator in our CAE-GAN.

E. Noise Reduction for Different Noise Types

We add different noises (i.e., EM, BW, and MA noises) to the raw signals to get noisy signals by setting different SNRs (i.e., 0 dB, 1 dB, 2 dB, 3 dB, 4 dB, and 5 dB). In addition, considering that mixed noises are more common in the actual situation, we also denoise the mixed noises (i.e., EM+BW, MA+BW, MA+EM, and MA+EM+BW noises).

Fig. 11 shows the results of noise reduction of the BW, EM, and MA noises with our method. In subfigures, from top to bottom, they are raw signals, noisy signals, and denoised signals. It can be clearly seen that the signals denoised by the proposed method are very close to the raw signals, indicating that this method can effectively remove three kinds of single noise: BW, EM, and MA.

Fig. 12 shows the noise reduction results of the mixed noises with our method. Subfigures a, b, c, and d respectively represent the denoising results of the EM+BW, MA+BW, MA+EM, and MA+EM+BW. For each group of subgraphs, from top to bottom, they are raw signals, noisy signals, and denoised signals respectively. As can be seen from these figures, the denoised ECG signals are very similar to the raw ECG signals. Obviously, the proposed method can also remove the mixed noises effectively.

Table I shows the denoising results of the proposed method for single noise and mixed noises on the test set. From the test set, for a single type of noise, the denoised SNR can be improved above 39.09 dB, and for mixed noises, the denoised SNR can reach more than 39.49 dB, indicating that this method has outstanding denoising effects for both single noise and mixed noise.

F. Evaluation of Generalization Ability

The practicality of the denoising method should be evaluated by the generalization ability of the model, that is the prediction ability of the model to learn unknown data. In order to explore the generalization ability of this method for different test data, we conduct experiments on the following two new data sets (i.e., new records and new lead) under different input SNRs (i.e., 0 dB, 1.25 dB, and 5 dB). The average denoising results are shown in Table II and Table III.

From Table II, after denoising, the average SNR of the new data set with single noise (i.e., EM, MA, and BW) still reaches

TABLE I
DENOISING RESULTS OF PROPOSED METHOD

Input SNR	Denoised Metrics	Noise Type							
		BW	EM	MA	EM+BW	MA+BW	MA+EW	MA+EW+BW	Average
0dB	SNR(dB)	39.59	40.09	39.09	40.08	40.43	40.38	39.49	39.88
	RMSE	0.0031	0.0029	0.0033	0.0029	0.0028	0.0029	0.0031	0.0030
1dB	SNR(dB)	41.29	42.31	41.21	42.42	42.57	42.67	41.92	42.06
	RMSE	0.0025	0.0022	0.0025	0.0022	0.0021	0.0022	0.0023	0.0023
2dB	SNR(dB)	41.87	43.04	41.78	43.22	43.23	43.38	42.66	42.74
	RMSE	0.0023	0.0020	0.0023	0.0020	0.0020	0.0020	0.0021	0.0021
3dB	SNR(dB)	42.09	43.26	41.96	43.41	43.36	43.65	42.86	42.94
	RMSE	0.0023	0.0020	0.0023	0.0019	0.0020	0.0019	0.0021	0.0021
4dB	SNR(dB)	41.96	43.04	41.74	43.20	43.13	43.40	42.65	42.73
	RMSE	0.0023	0.0021	0.0024	0.0020	0.0020	0.0020	0.0021	0.0021
5dB	SNR(dB)	41.10	41.75	40.59	41.80	41.93	42.03	41.28	41.50
	RMSE	0.0026	0.0024	0.0027	0.0024	0.0023	0.0023	0.0025	0.0025

TABLE II
AVERAGE DENOISING RESULTS OF NEW RECORDS DATA SET

Noise Type	Denoised Metrics	MIT-BIH Record Number									
		100	101	106	112	117	121	123	209	220	228
EM	SNR(dB)	40.37	37.52	40.90	43.52	44.96	39.39	42.40	41.48	40.75	43.34
	RMSE	0.0020	0.0029	0.0036	0.0023	0.0023	0.0022	0.0026	0.0024	0.0025	0.0030
MA	SNR(dB)	40.00	38.25	40.65	43.10	44.30	39.71	43.29	42.97	44.77	40.61
	RMSE	0.0021	0.0021	0.0027	0.0028	0.0033	0.0026	0.0023	0.0027	0.0023	0.0033
BW	SNR(dB)	40.92	38.72	41.27	43.89	45.35	40.64	44.65	44.37	45.98	41.95
	RMSE	0.0019	0.0019	0.0028	0.0026	0.0029	0.0024	0.0020	0.0023	0.0019	0.0030
MA+EM	SNR(dB)	39.56	37.79	39.84	42.04	42.72	38.82	42.71	42.15	43.80	39.66
	RMSE	0.0022	0.0022	0.0031	0.0032	0.0041	0.0029	0.0025	0.0030	0.0025	0.0037
EM+BW	SNR(dB)	39.94	37.92	40.57	43.12	44.53	39.04	44.62	43.57	45.90	40.58
	RMSE	0.0021	0.0021	0.0027	0.0028	0.0032	0.0028	0.0021	0.0025	0.0019	0.0033
MA+BW	SNR(dB)	40.73	38.66	41.29	43.04	44.98	40.45	43.03	43.59	44.64	41.35
	RMSE	0.0020	0.0020	0.0026	0.0029	0.0031	0.0024	0.0024	0.0026	0.0023	0.0030
MA+EM+BW	SNR(dB)	37.37	35.70	37.96	40.12	41.58	36.67	40.94	40.19	42.07	37.48
	RMSE	0.0028	0.0027	0.0039	0.0039	0.0044	0.0037	0.0030	0.0037	0.0030	0.0047

TABLE III
AVERAGE DENOISING RESULTS OF NEW LEAD DATA SET

Noise Type	Denoised Metrics	MIT-BIH Record Number									
		101	105	111	116	122	205	213	219	223	230
EM	SNR(dB)	37.54	38.53	41.26	40.54	34.13	39.17	41.64	39.01	40.55	38.44
	RMSE	0.0086	0.0089	0.0066	0.0077	0.0073	0.0074	0.0068	0.0056	0.0078	0.0062
MA	SNR(dB)	41.50	44.59	47.08	46.35	38.22	43.94	47.47	43.07	46.94	42.21
	RMSE	0.0054	0.0044	0.0034	0.0040	0.0047	0.0043	0.0034	0.0035	0.0037	0.0040
BW	SNR(dB)	43.70	44.22	45.92	45.48	41.00	44.13	46.45	44.06	46.31	43.53
	RMSE	0.0042	0.0046	0.0038	0.0043	0.0034	0.0041	0.0038	0.0031	0.0039	0.0034
MA+EM	SNR(dB)	41.81	44.42	46.57	46.59	39.76	44.28	46.63	43.16	46.37	42.45
	RMSE	0.0053	0.0045	0.0036	0.0039	0.0040	0.0042	0.0039	0.0034	0.0040	0.0041
EM+BW	SNR(dB)	41.95	44.37	46.48	46.46	39.34	44.73	47.73	43.28	46.55	43.28
	RMSE	0.0051	0.0045	0.0036	0.0038	0.0040	0.0038	0.0034	0.0033	0.0038	0.0039
MA+BW	SNR(dB)	44.09	45.77	47.88	47.64	38.90	45.36	48.67	44.11	48.02	43.89
	RMSE	0.0039	0.0038	0.0031	0.0034	0.0043	0.0036	0.0030	0.0031	0.0033	0.0032
MA+EM+BW	SNR(dB)	42.02	44.45	46.67	46.73	40.29	44.61	47.07	43.68	46.72	44.51
	RMSE	0.0051	0.0049	0.0037	0.0038	0.0037	0.0042	0.0037	0.0033	0.0038	0.0040

41 dB, the average SNR of two mixed noise cases (i.e., MA+EM, EM+BW, and MA+BW) reaches 40 dB, and the average SNR of three mixed noise case (i.e., MA+EM+BW) reaches 39 dB. It is proved that this method is also applicable to other records in the MIT-BIH Arrhythmia dataset.

As shown in Table III, for new lead test data, the denoised average SNRs are respectively over 39 dB, 44 dB, and 44 dB under a single noise (i.e., EM, MA, and BW), two mixed noise cases (i.e., MA+EM, EM+BW, and MA+BW), and three mixed noise case (i.e., MA+EM+BW).

G. Computational Load and Energy Consumption

In this section, we estimate the computational load and energy consumption. The trained generator is used in the noise reduction process, so we mainly focus on the generator. The total number of parameters in the generator model is 50.31 M, and the saved generator model takes up 191 MB, and the computational quantity of each sample denoising is 4.08243 GFLOP (giga floating point operations). The proposed model runs on an NVIDIA Tesla V100 server whose max power consumption is 250 W and max neural network computing ability is about

TABLE IV
DENOISING RESULTS FOR REMOVING THE BW NOISE

Methods	Input SNR	Denoised Metrics	MIT-BIH Record Number										
			103	105	111	116	122	205	213	219	223	230	
Improved DAE [16]	0dB	SNR(dB)	23.78	25.40	23.31	23.51	20.07	20.07	21.30	23.02	24.25	22.72	22.74
		RMSE	0.0260	0.0280	0.0340	0.0270	0.0500	0.0500	0.0320	0.0240	0.0270	0.0370	0.0340
Adversarial Method [18]	0dB	SNR(dB)	40.26	39.49	34.13	32.81	32.09	39.70	31.64	31.23	34.59	32.36	34.83
		RMSE	0.0032	0.0035	0.0066	0.0068	0.0075	0.0034	0.0086	0.0086	0.0059	0.0081	0.0062
Proposed Method	0dB	SNR(dB)	38.62	39.07	39.44	39.99	39.31	37.74	40.86	39.70	39.64	41.52	39.59
		RMSE	0.0025	0.0031	0.0040	0.0030	0.0027	0.0027	0.0032	0.0030	0.0028	0.0039	0.0031
Improved DAE [16]	1.25dB	SNR(dB)	23.82	25.42	23.32	23.59	20.08	20.08	21.36	23.31	24.41	22.74	22.81
		RMSE	0.0260	0.0280	0.0340	0.0270	0.0500	0.0500	0.0320	0.0230	0.0270	0.0370	0.0330
Adversarial Method [18]	1.25dB	SNR(dB)	40.72	39.87	34.53	33.51	32.42	40.34	32.05	32.09	35.12	32.44	35.31
		RMSE	0.0031	0.0034	0.0063	0.0063	0.0072	0.0031	0.0082	0.0078	0.0056	0.0080	0.0059
Proposed Method	1.25dB	SNR(dB)	40.83	41.56	42.42	42.68	41.70	40.06	43.63	41.95	41.92	44.47	42.12
		RMSE	0.0019	0.0023	0.0028	0.0021	0.0021	0.0020	0.0023	0.0023	0.0022	0.0026	0.0022
Improved DAE [16]	5dB	SNR(dB)	23.89	25.45	23.35	23.76	20.08	20.08	21.46	24.08	24.64	22.79	22.96
		RMSE	0.0250	0.0270	0.0340	0.0260	0.0500	0.0500	0.0310	0.0210	0.0260	0.0370	0.0330
Adversarial Method [18]	5dB	SNR(dB)	41.60	40.56	35.27	34.99	32.89	41.73	32.89	34.05	36.33	32.58	36.29
		RMSE	0.0027	0.0031	0.0058	0.0053	0.0068	0.0027	0.0074	0.0062	0.0048	0.0079	0.0053
Proposed Method	5dB	SNR(dB)	40.09	40.29	41.18	41.79	40.72	39.52	42.63	40.71	40.87	43.15	41.10
		RMSE	0.0021	0.0027	0.0033	0.0024	0.0023	0.0022	0.0026	0.0027	0.0025	0.0031	0.0026

TABLE V
DENOISING RESULTS FOR REMOVING THE EM NOISE

Methods	Input SNR	Denoised Metrics	MIT-BIH Record Number										
			103	105	111	116	122	205	213	219	223	230	
Improved DAE [16]	0dB	SNR(dB)	22.75	23.70	23.39	21.34	17.70	23.47	19.33	18.38	23.17	22.40	21.56
		RMSE	0.0290	0.0330	0.0340	0.0350	0.0500	0.0330	0.0400	0.0410	0.0310	0.0390	0.0365
Adversarial Method [18]	0dB	SNR(dB)	38.09	34.27	33.07	30.02	28.74	38.44	30.27	28.24	31.75	30.87	32.38
		RMSE	0.0050	0.0080	0.0093	0.0115	0.0134	0.0048	0.0125	0.0150	0.0101	0.0119	0.0102
Proposed Method	0dB	SNR(dB)	39.49	38.89	39.65	40.97	39.76	38.45	41.28	40.40	39.72	42.34	40.09
		RMSE	0.0022	0.0032	0.0040	0.0027	0.0026	0.0031	0.0027	0.0029	0.0034	0.0029	
Improved DAE [16]	1.25dB	SNR(dB)	22.97	23.94	23.57	21.82	18.76	23.57	19.79	19.07	23.55	22.54	21.96
		RMSE	0.0290	0.0330	0.0330	0.0330	0.0420	0.0330	0.0370	0.0380	0.0300	0.0380	0.0346
Adversarial Method [18]	1.25dB	SNR(dB)	38.56	34.79	33.45	30.77	29.28	38.96	30.68	29.21	32.19	31.11	32.90
		RMSE	0.0049	0.0075	0.0089	0.0105	0.0126	0.0046	0.0119	0.0134	0.0096	0.0116	0.0096
Proposed Method	1.25dB	SNR(dB)	42.34	42.26	42.75	44.20	43.00	41.46	44.93	43.69	42.74	45.48	43.28
		RMSE	0.0016	0.0021	0.0027	0.0018	0.0018	0.0017	0.0020	0.0018	0.0020	0.0023	0.0020
Improved DAE [16]	5dB	SNR(dB)	23.45	24.66	23.65	23.08	20.81	23.66	20.69	21.01	24.00	22.81	22.78
		RMSE	0.0270	0.0300	0.0330	0.0300	0.0350	0.0300	0.0340	0.0300	0.0280	0.0370	0.0314
Adversarial Method [18]	5dB	SNR(dB)	39.39	35.67	34.10	21.72	30.01	39.89	31.37	31.23	32.96	31.53	32.79
		RMSE	0.0044	0.0068	0.0082	0.0095	0.0116	0.0041	0.0110	0.0106	0.0088	0.0111	0.0086
Proposed Method	5dB	SNR(dB)	41.05	40.72	41.18	42.65	41.89	40.29	42.73	42.03	41.43	43.56	41.75
		RMSE	0.0018	0.0033	0.0056	0.0022	0.0020	0.0021	0.0026	0.0022	0.0024	0.0029	0.0027

112 TFLOPS. Based on estimating, the energy consumption for each sample processing is about 9.1 mW. Through our testing, each sample denoising needs about 18 milliseconds. The above energy consumption cannot represent the power consumption of the method to be transplanted to portable low-power devices in the future.

H. Comparison With Other Methods

We compare the performance metrics of Improved DAE [16], Adversarial Method [18], and the proposed method (CAE-CGAN) by experiments. It should be noted that the experimental samples in the three methods are different. Their length of the samples are 101, 310 and 512, respectively. Their number of the training samples are 30000, 54000 and 61020, respectively. Their number of the test samples are 2000, 5940 and 15120, respectively. In addition, the proposed method is every 512

sampling points as a sample, while the other two methods need to use a heartbeat as a sample.

Tables IV, V, and VI respectively show the SNR and RMSE after noise reduction for ten records containing BW, EM and MA noise. It can be seen that our method has the best performance in SNR and RMSE for different noises. For BW and MA, the average SNRs are over 39 dB, for EM, the average SNRs are over 40 dB.

Our proposed CAE-CGAN adds additional information in the structure as a condition to direct the data generation process, and the adversarial training way of CAE-CGAN does not require the selection of samples. Therefore, the denoised signal is closer to the raw signal. The above experiments also confirm that this method avoids their shortcomings of the previous methods and has a better noise reduction effect.

As shown in Fig. 13, in the seven noise cases, the denoising results of our method all exceed the Adversarial Method [18]

TABLE VI
DENOISING RESULTS FOR REMOVING THE MA NOISE

Methods	Input SNR	Denoised Metrics	MIT-BIH Record Number										
			103	105	111	116	122	205	213	219	223	230	
Improved DAE [16]	0dB	SNR(dB)	21.38	24.72	23.15	19.22	19.57	24.23	19.59	18.80	22.91	22.58	21.62
		RMSE	0.0340	0.0300	0.0350	0.0450	0.0400	0.0310	0.0380	0.0390	0.0320	0.0380	0.0360
Adversarial Method [18]	0dB	SNR(dB)	41.36	36.49	35.90	32.47	31.06	40.58	33.73	32.37	33.46	33.98	35.14
		RMSE	0.0042	0.0073	0.0079	0.0107	0.0126	0.0045	0.0100	0.0112	0.0100	0.0098	0.0088
Proposed Method	0dB	SNR(dB)	37.94	38.28	39.21	39.81	38.28	36.98	40.31	39.65	39.00	41.42	39.09
		RMSE	0.0027	0.0035	0.0042	0.0030	0.0031	0.0029	0.0035	0.0030	0.0031	0.0039	0.0033
Improved DAE [16]	1.25dB	SNR(dB)	22.41	24.86	23.27	20.22	20.02	24.49	19.78	19.63	23.41	22.60	22.07
		RMSE	0.0310	0.0290	0.0340	0.0400	0.0380	0.0300	0.0370	0.0340	0.0300	0.0380	0.0340
Adversarial Method [18]	1.25dB	SNR(dB)	42.10	37.46	36.16	33.67	31.88	41.19	34.26	33.40	34.36	34.22	35.87
		RMSE	0.0038	0.0065	0.0076	0.0093	0.0115	0.0042	0.0094	0.0100	0.0090	0.0096	0.0081
Proposed Method	1.25dB	SNR(dB)	40.59	41.43	42.74	42.50	41.33	39.67	43.79	42.68	41.75	44.88	42.14
		RMSE	0.0019	0.0023	0.0027	0.0022	0.0021	0.0021	0.0022	0.0021	0.0022	0.0024	0.0022
Improved DAE [16]	5dB	SNR(dB)	23.33	25.13	23.33	22.41	20.63	24.67	20.63	21.97	24.21	22.63	22.89
		RMSE	0.0270	0.0280	0.0340	0.0310	0.0360	0.0300	0.0340	0.0270	0.0280	0.0380	0.0310
Adversarial Method [18]	5dB	SNR(dB)	43.24	39.55	36.66	35.88	33.50	42.22	34.95	35.38	36.35	34.55	37.23
		RMSE	0.0033	0.0050	0.0072	0.0072	0.0096	0.0037	0.0087	0.0080	0.0072	0.0092	0.0069
Proposed Method	5dB	SNR(dB)	39.34	39.65	40.66	40.92	40.08	38.25	42.18	41.08	40.42	43.28	40.59
		RMSE	0.0023	0.0030	0.0035	0.0026	0.0025	0.0026	0.0028	0.0025	0.0025	0.0030	0.0027

TABLE VII
EXPERIMENTAL RESULTS OF CLASSIFICATION

Noise Type (SNR=0dB)	Signal Type	N	V	A	L	Acc	Improved	Average Improved
		Raw signal	100.00%	95.45%	100.00%	98.11%	98.50%	
EM	Noisy signal	69.77%	42.31%	42.11%	42.55%	51.00%		
	Denoised signal	100.00%	94.00%	88.37%	100.00%	96.00%	45.00%	
MA	Noisy signal	90.74%	65.57%	71.74%	89.74%	78.50%		
	Denoised signal	100.00%	92.50%	82.76%	96.08%	92.50%	14.00%	
BW	Noisy signal	87.76%	78.57%	73.68%	82.35%	81.11%		
	Denoised signal	100.00%	97.37%	91.49%	95.83%	96.50%	15.35%	
EM+MA	Noisy signal	85.37%	46.94%	28.26%	45.45%	51.50%		
	Denoised signal	100.00%	92.86%	93.62%	93.02%	95.50%	44.00%	32.24%
EM+BW	Noisy signal	67.39%	48.78%	40.00%	41.86%	48.50%		
	Denoised signal	100.00%	95.74%	90.70%	97.83%	96.00%	47.50%	
MA+BW	Noisy signal	91.49%	54.55%	81.82%	88.89%	80.50%		
	Denoised signal	100.00%	95.45%	97.37%	98.04%	97.50%	17.00%	
EM+BW+MA	Noisy signal	66.67%	60.00%	32.56%	59.52%	53.50%		
	Denoised signal	100.00%	93.62%	97.87%	94.44%	97.00%	43.50%	

significantly, and it is increased by 35.81% at most. The average SNR of seven kinds of noises is increased by 25.55%. It is obvious that the proposed method is better than the Adversarial Method [18].

I. Classification Experiments

Besides the metrics RMSE and SNR, we also give the new metric that is the classification accuracy of cardiac diseases.

The cardiac disease classification experiment is carried out as follows. Firstly, Pan-Tompkins (PT) algorithm [35] is used to locate the QRS position of the signal, and then 250 points are intercepted as a heartbeat period based on the QRS position. According to the annotation of the MIT-BIH database, four types of heartbeats are saved, which are normal beat (N), premature ventricular contraction (V), atrial premature beats (A), and left bundle branch block beat (L), respectively. In order to

balance the data distribution of different types of heartbeats, the same number of heartbeats is selected for each type. In this experiment, the number is 50. Secondly, 5-order wavelet decomposition is performed for each heartbeat data. Daubechies 6 (DB6) wavelet is used as the wavelet function, and the approximation coefficients of the wavelet transform are taken to represent the characteristics of each heartbeat data. Finally, the Support Vector Machine (SVM) was used to classify heartbeats in four categories (i.e., N, V, A, and L).

The classification results are shown in Table VII. The accuracy of the denoised signals is significantly improved compared with that of the noisy signals and is very close to that of raw signals. Since EM can mimic the appearance of ectopic beats [34], it has a great impact on classification accuracy. For the noisy signals with EM (i.e., EM, EM+MA, EM+BW, and EM+BW+MA), the improved accuracy is the most obvious, which reaches above 43%. This shows that the denoising effect for EM is greatly

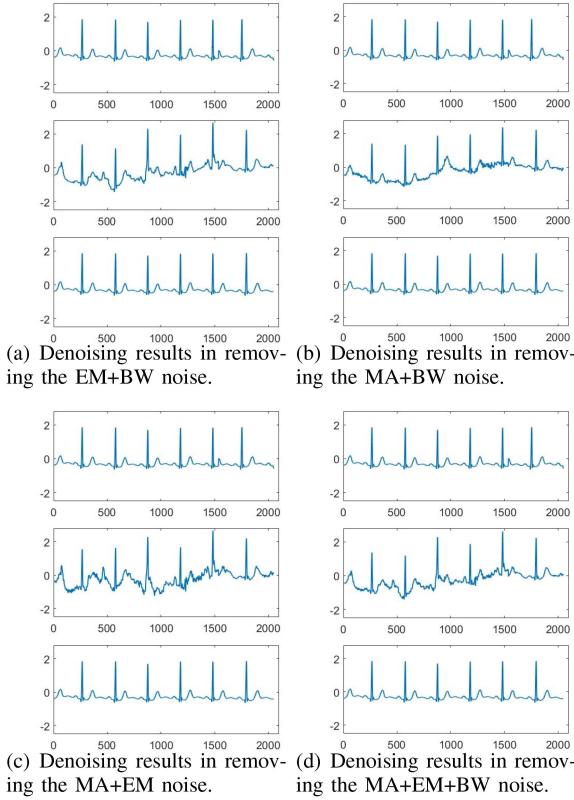


Fig. 12. Denoising results in removing the mixed noise.

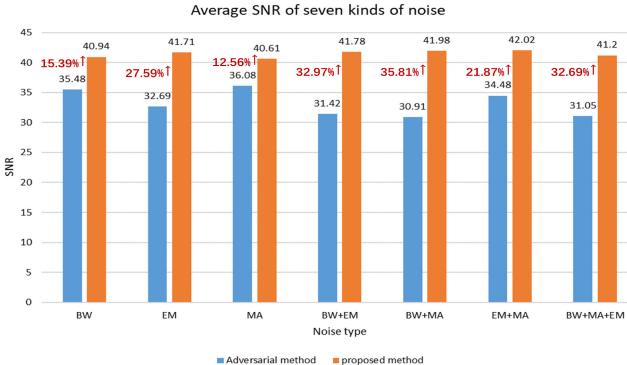
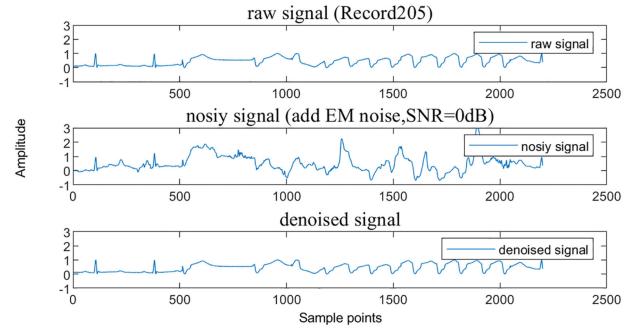


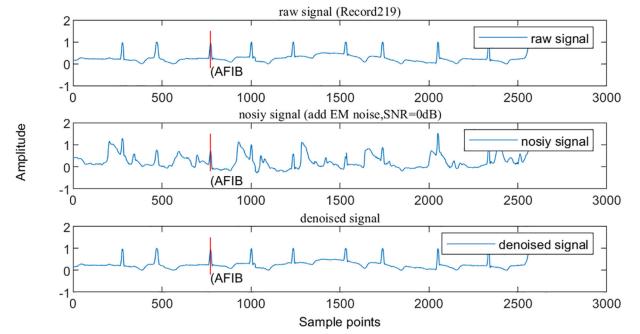
Fig. 13. Comparison with the Adversarial Method [18].

improved by the proposed method. For the seven noise cases under $\text{SNR} = 0 \text{ dB}$, the diseases classification accuracy of denoised signals can reach more than 92%, and the average accuracy increased above 32%. The results showed that the denoised signal retained medical features of raw data.

Taking record 205 as an example, the raw signal, the noisy signal, and the denoised signal are shown in Fig. 14(a). We can find that the denoised signal maintains the rhythm irregularity of the raw data. Taking the segment containing atrial fibrillation (AF) in record 219 as an example, the raw signal, the noisy signal, and the denoised signal are shown in Fig. 14(b). The red line represents the rhythm annotation of AF (“AFIB” in MIT-BIH database). It also shows the denoising signals can retain beneficial medical information.



(a) The denoising result of record 205



(b) The denoising result of record 219 (AF)

Fig. 14. The denoising result of record 205 and 219 by the proposed method.

J. Summary

CGAN is mainly used for generating tasks (such as image generation, audio generation, etc.). These tasks generate different output by inputting the input variables z , whereas image enhancement tasks do not require input variables z . We introduce CGAN into the ECG denoising task, so it is necessary to explore the role of the input variables z in the ECG denoising task. The experimental results confirm that removing the input variables z can not only improve the denoising performance but also retain more ECG waveform details.

After the structure design experiment, our generator chose a convolutional auto-encoder. We also perform dimensions tuning experiments on the generator to achieve the best denoising effect. Our method achieves better denoising effect on both single noise and mixed noise, indicating that the selected model (Sy-CNN+SC) has better adaptability to remove different types of noises.

V. CONCLUSION

We propose a denoising method of ECG signals based on CGAN in this paper. Our proposed CAE-CGAN adds additional information in the structure as a condition to direct the data generation process and this method avoids the shortcomings of the previous methods. For single noise (i.e., EM, BW, and MA), the average SNRs can reach more than 40.61 dB. For two mixed noise cases (i.e., MA+EM, EM+BW, and MA+BW), the average SNRs can reach more than 41.78 dB. For the three mixed noise (i.e., MA+EM+BW), the average SNR can reach 41.2 dB. The

average SNR of seven kinds of noises was increased by 25.05% compared with the state-of-the-art method. These results show that the proposed method has outstanding denoising effects. In addition, the method is also applicable to other records and lead in the MIT-BIH arrhythmia dataset, the denoising results also prove our method has good generalization ability. We also carried out the cardiac disease classification experiment. For the seven noise cases under SNR=0 dB, the diseases classification accuracy of denoised signals can reach more than 92%, and the average accuracy increased above 32%. All the above results indicate that this method is promising in ECG denoising.

Although the denoising effect of our method has improved obviously, we found that the denoising results on several records (103, 105 and 205) were still not ideal under single MA or BW noise case. The problem might be caused by a variety of factors, such as noise type, noise intensity, and sample state, etc. In addition, the data used in the current study was only derived from modified limb lead II (MLII) and lead V1, so 12-lead data needs to be explored in the future. They will be our further studies.

REFERENCES

- [1] K. Li, W. Pan, Y. Li, Q. Jiang, and G. Liu, "A method to detect sleep apnea based on deep neural network and hidden Markov model using single-lead ECG signal," *Neurocomputing*, vol. 294, pp. 94–101, Jun. 2018.
- [2] Y. Li, Y. Pang, J. Wang, and X. Li, "Patient-specific ECG classification by deeper CNN from generic to dedicated," *Neurocomputing*, vol. 314, pp. 336–346, Nov. 2018.
- [3] S. Hong, Y. Zhou, J. Shang, C. Xiao, and J. Sun, "Opportunities and challenges of deep learning methods for electrocardiogram data: A systematic review," *Comput. Biol. Med.*, vol. 122, Jul. 2020, Art. no. 103801.
- [4] D. G. Katsiris, G. C. Sontis, and A. J. Camm, "Prognostic significance of ambulatory ECG monitoring for ventricular arrhythmias," *Prog. Cardiovasc. Dis.*, vol. 56, no. 2, pp. 133–142, Sep. 2013.
- [5] G. Friesen, T. Jannett, M. Jadallah, S. Yates, S. Quint, and H. Nagle, "A comparison of the noise sensitivity of nine QRS detection algorithms," *IEEE Trans. Biomed. Eng.*, vol. 37, no. 1, pp. 85–98, Jan. 1990.
- [6] M. Z. U. Rahman, R. A. Shaik, and D. R. K. Reddy, "Efficient sign based normalized adaptive filtering techniques for cancellation of artifacts in ECG signals: Application to wireless biotelemetry," *Signal Process.*, vol. 91, no. 2, pp. 225–239, Feb. 2011.
- [7] 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.
- [8] M. A. Kabir and C. Shahnaz, "Denoising of ECG signals based on noise reduction algorithms in EMD and wavelet domains," *Biomed. Signal Process. Control*, vol. 7, no. 5, pp. 481–489, Sep. 2012.
- [9] S. Ari, M. K. Das, and A. Chacko, "ECG signal enhancement using s-transform," *Comput. Biol. Med.*, vol. 43, no. 6, pp. 649–660, Jul. 2013.
- [10] M. Gadaleta and A. Giorgio, "A method for ventricular late potentials detection using time-frequency representation and wavelet denoising," *ISRN Cardiol.*, vol. 2012, pp. 1–9, Aug. 2012.
- [11] L. Smital, M. Vítek, J. Kozumplík, and I. Provažník, "Adaptive wavelet wiener filtering of ECG signals," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 2, pp. 437–445, Feb. 2013.
- [12] Z. Wang, F. Wan, C. M. Wong, and L. Zhang, "Adaptive Fourier decomposition based ECG denoising," *Comput. Biol. Med.*, vol. 77, pp. 195–205, Oct. 2016.
- [13] S. Poungponsri and X.-H. Yu, "Electrocardiogram (ECG) signal modeling and noise reduction using wavelet neural networks," in *Proc. IEEE Int. Conf. Automat. Logistics*, 2009, pp. 394–398.
- [14] S. Poungponsri and X.-H. Yu, "An adaptive filtering approach for electrocardiogram (ECG) signal noise reduction using neural networks," *Neurocomputing*, vol. 117, pp. 206–213, Oct. 2013.
- [15] P. Xiong, H. Wang, M. Liu, F. Lin, Z. Hou, and X. Liu, "A stacked contractive denoising auto-encoder for ECG signal denoising," *Physiol. Meas.*, vol. 37, no. 12, pp. 2214–2230, Nov. 2016.
- [16] 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.
- [17] I. Goodfellow *et al.*, "Generative adversarial nets," *Adv. Neural Inf. Process. Syst.*, vol. 27, pp. 2672–2680, 2014.
- [18] J. Wang *et al.*, "Adversarial de-noising of electrocardiogram," *Neurocomputing*, vol. 349, pp. 212–224, Jul. 2019.
- [19] 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.
- [20] B. Xu, R. Liu, M. Shu, X. Shang, and Y. Wang, "An ECG denoising method based on the generative adversarial residual network," *Comput. Math. Methods Med.*, vol. 2021, pp. 1–23, Apr. 2021.
- [21] M. Mirza and S. Osindero, "Conditional generative adversarial nets," Nov. 2014, Accessed: Mar. 09, 2022. [Online]. Available: <http://arxiv.org/abs/1411.1784>
- [22] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. IEEE Eur. Conf. Comput. Vis.*, 2016, pp. 694–711.
- [23] C. Ledig *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4681–4690.
- [24] R. A. Yeh, C. Chen, T. Y. Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with deep generative models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5485–5493.
- [25] H. Zhang, V. Sindagi, and V. M. Patel, "Image de-raining using a conditional generative adversarial network," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 11, pp. 3943–3956, Nov. 2020.
- [26] R. Li, J. Pan, Z. Li, and J. Tang, "Single image dehazing via conditional generative adversarial network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8202–8211.
- [27] X. Kuang, X. Sui, Y. Liu, Q. Chen, and G. Gu, "Single infrared image enhancement using a deep convolutional neural network," *Neurocomputing*, vol. 332, pp. 119–128, Mar. 2019.
- [28] X.-H. Qin *et al.*, "Using a one-dimensional convolutional neural network with a conditional generative adversarial network to classify plant electrical signals," *Comput. Electron. Agriculture*, vol. 174, Jul. 2020, Art. no. 105464.
- [29] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2794–2802.
- [30] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 481–489.
- [31] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *Lecture Notes Comput. Sci.*, 2011, pp. 52–59.
- [32] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1125–1134.
- [33] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH arrhythmia database," *IEEE Eng. Med. Biol. Mag.*, vol. 20, no. 3, pp. 45–50, 2001.
- [34] G. B. Moody, W. K. Muldrow, and R. G. Mark, "Noise Stress Test for Arrhythmia Detectors," *Comput. Cardiol.*, vol. 11, pp. 381–384, 1984.
- [35] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," *IEEE Trans. Biomed. Eng.*, vol. BME-32, no. 3, pp. 230–236, Mar. 1985.