**Attention-Based Convolutional Denoising Autoencoder for Two-Lead ECG Denoising and Arrhythmia Classification**

Prateek Singh, Student Member, IEEE, and Ambalika Sharma

**Abstract**—This article presents a fast and accurate electrocardiogram (ECG) denoising and classification method for low-quality ECG signals. To achieve this, a novel attention-based convolutional denoising autoencoder (ACDAE) model is proposed that utilizes a skip-layer and attention module for reliable reconstruction of ECG signals from extreme noise conditions. Skip-layer connections are used to reduce information loss while reconstructing the original signal, and a lightweight, efficient channel attention (ECA) module is used to update relevant features retrieved via cross-channel interaction efficiently. The model is trained and tested using four widely available databases. For evaluation, the signals are mixed with simulated additive white Gaussian noise (AWGN) ranging from $-20$ to $20$ dB and the Massachusetts Institute of Technology-Beth Israel Hospital (MIT-BIH) noise stress test database (NSTDB) noise ranging from $-6$ to $24$ dB. The model outperformed the most cited published works, achieving an average signal-to-noise ratio (SNR) improvement of $19.07 \pm 1.67$ and a percentage-root-mean-square difference (PRD) of $11.0\%$ at $0$-dB SNR. The model classification performance on 60,000 beats is $98.76\% \pm 0.44\%$ precision, $98.48\% \pm 0.58\%$ recall, and $98.88\% \pm 0.42\%$ accuracy, respectively, using a stratified fivefold cross-validation strategy.

**Index Terms**—Arrhythmias, atrial fibrillation (AF), convolutional neural network (CNN), denoising autoencoder (DAE), electrocardiogram (ECG).

I. INTRODUCTION

AN ELECTROCARDIOGRAM (ECG) is the most extensively utilized vital sign monitoring technology in modern healthcare devices. ECG is a noninvasive procedure that records the heart’s electrical activity and aids in diagnosing cardiac disease. Recent advancements in remote monitoring technology have enabled ECGs to be performed in the privacy and convenience of one’s own home or office rather than in a hospital setting. Remote monitoring technologies have improved healthcare for those with periodic cardiac arrhythmias, because they can continually monitor heart activity. With these advancements, massive amounts of ECG data are being gathered simultaneously, which needs processing and interpretation. Medical experts in the domain usually interpret and diagnose the ECG; however, it is time-consuming and expensive, as it requires the expertise of doctors. Furthermore, in low- and middle-income nations, there is a severe scarcity of cardiac experts, mainly in the rural areas [1]. Therefore, a simple, easy-to-use, reliable ECG analysis tool is required to deliver better insights into these remote monitoring devices. As a result, various computer-aided technologies have been developed by researchers in the recent past to assist doctors. Despite being the most often used diagnostic tool, computer-read ECGs have been found to have considerable errors [2]. This means that conventional algorithms can no longer be relied upon as a primary diagnostic tool. Therefore, more reliable computer-aided algorithms are required to support doctors and even nurses in primary diagnosis.

Arrhythmia detection is primarily concerned with classifying each heartbeat according to its morphology, which can be classified into several distinct classifications. The traditional method of detecting arrhythmias includes several steps, such as preprocessing, feature extraction, dimensionality reduction, and classification. Feature extraction is the heart of most machine learning tasks, and domain experts in the field do it. The extracted features determine the classification accuracy. Therefore, much attention is devoted to developing feature extraction techniques by the researchers. Some of the most used methods employed for the ECG classification task are wavelet transform [3], principal component analysis (PCA) [4], independent component analysis (ICA) [5], and higher-order statistics (HOS) [6]. After extracting features, they are given to the classifiers to interpret the results. The results will vary depending on the features used to interpret the ECGs. In addition, raw ECG signals are susceptible to noise, which can alter the amplitude and time intervals of the ECG. These are the factors that help diagnose arrhythmias, and if these changes are incorrectly diagnosed, misdiagnosis can occur. Therefore, filtering ECG signals is always necessary to avoid false positives (FPs) and erroneous diagnoses. As a result, adequate preprocessing is required before analyzing the ECG signal. Denoising ECG signals can be accomplished using various methods, the majority of which rely on traditional techniques based on parameters that are particularly susceptible to noise, such as fixed filters like finite impulse response (FIR) filters and infinite impulse response (IIR) filters [7]. Fixed filters eliminate all signals that lie in the cutoff frequencies, obliterating the ECG signal information in those frequencies. Therefore, to answer fixed filters drawbacks, adaptive filtering techniques [8] are proposed to denoise ECG signals. However, they have a problem that adaptive algorithms.

1557-9662 © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.
often require noise reference signals as input, which are
difficult to collect using the ECG signal acquisition system.
The researchers also exploited the nonstationary nature of
ECG signals, allowing them to use information in the
time-frequency domain. Thus, time-frequency-based techniques,
such as wavelet [9] and empirical mode decomposition
(EMD) [10], have gained popularity and shown promising
results in ECG denoising. These time-frequency filtering
methods have two flaws: first, they only focus on low- or high-
level characteristics, and second, for low frequencies, they give
greater frequency resolution but poor time resolution; for high
frequencies, they provide better frequency resolution but poor
time resolution [11].

Mobile-based ECG devices and patches are introduced in
the modern era that requires less complex and high yield
algorithms. One such method for filtering ECG is denois-
ing autoencoders (DAEs), which has shown better filtering
capabilities than other powerful nonlinear mapping capabilities. Xiong et al. [12] used a
stacked contractive DAE to denoise the ECG signal. They
also demonstrated a DNN-based DAE for noise reduction
in the ECG, with significant improvements in the signal-to-
noise ratio (SNR) and root-mean-square error (RMSE) on
noise induced from the Massachusetts Institute of Technology-
Beth Israel Hospital (MIT-BIH) noise stress test database
(NSTDB), with SNR ranging from 0 to 5 dB for testing
their results. Xiong et al. [13] combined DAE and wavelet
transform with soft thresholding to achieve better denoising in
the range of 0–5 dB on NSTDB. Chiang et al. [14] proposed
a 13-layer fully convolutional network (FCN)-based DAE for
denoising ECG signals. They enhanced the SNR by 15.49 dB
using the NSTDB, and they investigated noise corruption
levels ranging from −1 to 7 dB. Romero et al. [15] presented
a multitensor linear and nonlinear (MKLANL) module for
baseline wander (BW) removal that is inspired by the inception
module and achieved highly promising results when compared
with state-of-the-art signal processing approaches. The study’s
only drawback is that it focused solely on reducing BW noise
and ignored other noises. Qiu et al. [16] proposed a two-stage
denoising approach. In the first stage, a U-net model is used to
remove the noise from the ECG. Then, a DR-net is designed
in the second stage for the detailed restoration of the ECG
signal. The experiment demonstrates the applicability of this
strategy in the presence of significant noise. Despite this, the
study could not remove noises from low SNR signals. Despite
the researcher’s best efforts, denoising ECG signal remains
challenging, particularly in noisy or low SNR environments.
This work aims to bridge the gap by denoising low SNR ECG
signals.

A. Contributions and Organization

To address the abovementioned issues of not being able
to denoise low SNR ECG signals, this study proposes a
deep convolutional DAE (CDAE) network with symmetric
skip-layer connections for two-channel ECG denoising. The
model is inspired by the work of [17] on image restoration;
however, the hyperparameters are redesigned for 1-D ECG
time series data. Furthermore, the efficient channel atten-
tion (ECA) structure is introduced to efficiently update the
features retrieved via cross-channel interaction, allowing the
network to pay more attention to the features of relevant
information. As a result, the complexity of CDAE is optimized
owing to the skip connections. The network is trained and
tested using four widely used and readily available ECG
databases. The databases are corrupted by the four most
prevalent noises in ECG measurement: 1) BW; 2) muscle
artifact (MA); 3) electrode motion (EM); and 4) channel noise
(additive white Gaussian noise (AWGN)) [18], to test the
effectiveness of the proposed model. The results are compared
with state-of-the-art techniques. The main contributions of
the study are as follows.

1) A novel neural network model named attention-based
CDAE (ACDAE) is proposed with symmetric skip-layer
connections for two-channel ECG denoising.
2) The ECA mechanism is integrated for the denosing
and classification tasks, which will help restore ECG
morphology.
3) After denoising, two models are used to evaluate
the atrial fibrillation (AF) classification performance
using compressed features obtained from the encoder
phase of the model. The proposed model’s classification
performance is assessed by introducing AWGN to the
ECG signal at various SNR values ranging from
−20 to 20 dB.
4) To the best of the author’s knowledge, this is the first
comprehensive attempt to investigate the different case
scenarios that influence classification performance from
noisy ECG signals.

The rest of this article is organized as follows.
Section II describes the dataset used and preprocessing steps.
In Section III, the primary methods, including DAE and
ECA, are briefly reviewed. The denoising problem to be
solved is formulated and describes the proposed ACDAE in
detail. Section IV presents three case studies to evaluate the
performance of the proposed method. Section V compares the
results with state-of-the-art methods. Finally, conclusions are
provided in Section VI.

II. DATASET USED AND PREPROCESSING

In this study, four prominent freely available databases from
the Physionet [19] are used to train and test ACDAE denoising
and AF classification models. In a nutshell, these databases are
as follows.

The MIT-BIH AF database (AFDB) includes 25 10 h of
two-channel ECG recordings sampled at 250 Hz. Two records,
“04936” and “05091,” are not used out of 25 records due
to incorrect annotations. These recordings come with rhythm
annotation files that have been painstakingly generated. If one
or more beats in a beat sequence show signs of AF, the beat
sequence is classed as AF (usually paroxysmal), while all other
beats are classified as normal in the database.

The MIT-BIH normal sinus rhythm database (NSRDB)
includes 18 long-term ECG recordings. No arrhythmia beats
are available except sporadic ectopy. The recordings are digi-
tized at 128 samples/second/channel. A reference annotation
file specifying the location and kind of each beat is included
with each digitized recording.

The MIT-BIH NSTDB [20] contains 12 1/2-h ECG record-
nings and three 1/2-h recordings of noise typical in ECG
recordings. The database includes noise records, such as
baseline “bw,” electrode motion “em,” and muscles artifact “ma,” that can be added to ECG records to create noise stress test records.

The MIT-BIH arrhythmia (BIHA) database has 48 two-lead ECG records of 30-min duration. The sampling frequency of BIHA is 360 Hz, which is resampled at 250 Hz for our experimentation. The network NSTDB noise is added for denoising purposes for training and testing.

A. Preprocessing of ECG Signals

Since each database has a distinct sampling frequency, thus they all are resampled at 250 Hz to ensure uniformity across all the databases. The preprocessing is divided into the following steps. First, the two-channel ECG data from the AFDB are segregated into AF and normal beats using an annotation file. Then, the signals are normalized between 0 and 1. Next R peaks are detected using the Christov segmenter algorithm from BioSPPy toolbox [21] for segmentation. We need at least two R peaks in a segmented window for AF classification. Therefore, a window of 1.2 s is taken, as the normal range for RR interval is 0.6–1.2 s. Proper zero padding is performed to ensure that each window has the same size, as normal ECG signals RR intervals are much larger than AF signals. Similarly, NSRDB is also preprocessed; both database’s data are shuffled, and stratified fivefold cross validation is employed. From the 21 records in the AFDBs, we have 345 241 samples of AF and 619 345 samples of non-AF beats. We selected equal numbers of AF and N samples from each record, totaling 600 000 samples, and we took all the records of the NSRDB dataset.

B. Data Preparation for Experiments

Some studies suggest testing models on noises ranging from 12 dB to −6 dB, as the majority of wearable devices may produce noise in this range [22]. However, to examine the effectiveness of the proposed model, we have added the AWGN noise with levels ranging from −20 to 20 dB, with a 5-dB step size. A total of two experiments are carried out in this research, as follows.

1) First, experiment is used to evaluate the ECG denoising performance and classification performance of AF. For this experiment, AFDB and NSDB preprocessed data are mixed with the AWGN. So, overall nine experiments are done with the SNR levels of −20, −15, −10, −5, 0, 5, 10, 15, and 20 dB.

2) Second, experiment is carried out to evaluate the proposed model on more realistic noise experienced by ECG. The BIHA and NST databases are used for this experiment to train and test the model performance. The NSTDB comes with noise records, such as “bw,” “em,” and “ma”; therefore, these are added with varying SNR levels of −6, 0, 6, 12, and 24 dB to BIHA ECG records. The experiment is also performed using AWGN with −20- to 20-dB SNRs. A total of 47 records are used, as two records belong to the same subject; 30 records are used for training, five for validation, and 12 for testing. The experiment results are discussed in Sections IV and V.

Fig. 1. Flowchart of the proposed method.

III. METHODOLOGY

The proposed method is shown in Fig. 1. It has two primary functions: 1) an ACDAE that aids in the denoising of ECG signals and 2) two classification modules. For simplicity, classification module 1 is referred to as M1 and module 2 as M2 throughout this article. M1 takes the encoder’s learned feature maps. It passes them to the ECA module, where cross-channel weighted attention is given to the important feature maps before passing them to the fully connected (FC) layer for AF classification. M2 also uses the encoder’s learned feature maps, but now, instead of the ECA module, it is fed to the global average pooling (GAP) and FC layer. The goal of this analysis is to see how ECA attention impacts classification. Detailed information about each module is described in Sections III-A–III-C.

A. Convolutional DAE

Autoencoder (AE) is an excellent unsupervised learning approach for extracting data feature representation. They are commonly utilized in biomedical applications for compression, feature extraction, and denoising. Hou et al. [23] used long-short term memory (LSTM)-based AEs for ECG arrhythmia detection. Apart from ECG signals, AE is also used in machine applications, such as Mao et al. [24], Zhao et al. [25], and Yu et al. [26] used different AE networks for fault detection and to predict the health state of the machine. AE is divided into two paths: encoding, which compresses signals by learning features, and decoding, which expands the compressed signal back to its original shape. The repeated convolutional layer is followed by a nonlinear activation function and a max-pooling operation for downsampling in the encoder path. On the other hand, the decoder path starts with transposed convolutional layers followed by an upsampling layer and activation function and will give the denoised signal by reconstructing the input. The mathematical formulation is as follows.

The encoding path \( h = c(x; \theta) \) translates a given input, \( x \in [0, 1]^n \), to a hidden layer, \( h \in [0, 1]^m \), with parameters \( \theta \) and \( n, m \in \mathbb{N} \). The decoding path \( \hat{x} = f(h; \theta') \) converts \( h \) into a reconstruction in the input space \( \hat{x} \in [0, 1]^n \). The AE’s training goal is to identify parameters \( \theta, \theta' \) that minimize the reconstruction error \( L(x, \hat{x}) \), i.e., the difference between \( x \) and...
\( \hat{x} \) for all \( x^i, i \in (1, \ldots, \tau) \), samples in the training set

\[
\theta, \theta' = \arg \min_{\theta, \theta'} \left[ \frac{1}{\tau} \sum_{i=1}^{\tau} L(x_i, \hat{x}_i) \right].
\]  

(1)

To calculate the reconstruction error, traditional-squared error \( L(x, \hat{x}) = \|x - \hat{x}\|^2 \) can be used or the cross entropy error function as

\[
L(x, \hat{x}) = -\sum_{n=1}^{r} [x_n \log \hat{x}_n + (1 - x_n) \log(1 - \hat{x}_n)].
\]  

(2)

The hidden layer algorithm can learn input features along the primary axes of variation coordinates by minimizing reconstruction errors. They follow the same principle used in the PCA. Data are projected onto the primary component that captures the most important data. Some relevant information may be lost during the compression of the original feature map. It is designed to have a small reconstruction error for test data.

As the AE model task is to reconstruct the input, they should be sensitive enough to recreate the original signal but insensitive enough to the training data; i.e., the model should not learn the input while training, which will cause overfitting. Therefore, [27] proposed a study where they corrupted the input by introducing slight noise, so that the model now does not simply develop a mapping that memorizes the training data, because the input and target output are different. Instead, it is not simply to develop a mapping that memorizes the training data, but instead to learn a vector field for mapping the input data toward a lower-dimensional manifold.

Here, each training example \( x \) is the ECG signal. It is corrupted by a stochastic mapping \( \hat{x} = q(\tilde{x}|x) \); i.e., AWGN is added to the input data, which partially destroys it according to the destruction rate (based on SNR). The DAE then uses the encoding and decoding functions to determine the reconstruction of the corrupted input, as \( \hat{x} = f(c(\tilde{x})) \). Then, the parameters are updated in the direction of \((\delta L(x, \hat{x})]/\delta \theta)\). As a result, the DAE tries to reconstruct \( x \) instead of \( \hat{x} \). After obtaining the reconstruction signal from DAEs, the SNR value must be determined to assess signal quality.

The signal corruption process can be formulated as follows:

\[
\hat{x} = \eta(x)
\]  

(3)

where \( \eta : \mathbb{R}^n \rightarrow \mathbb{R}^n \) is an arbitrary stochastic corrupting process that corrupts the input. The learning aim of the denoising job, thus, becomes

\[
f = \arg \min_y E_y \|f(\hat{x}) - x\|^2.
\]  

(4)

This formulation goal is to identify a function \( f \) that best approximates \( \eta^{-1} \). The signal denoising and restoration problems in a cohesive framework are tackled by selecting appropriate \( \eta \) in different contexts. Section III-B will discuss the ECA network (ECANet) and its working.

**B. ECA Network**

According to cognitive research, people use an attention mechanism to preferentially concentrate on a subset of all information while ignoring other observable information.

Convolutional neural networks (CNNs) performance has recently been demonstrated to be improved by a channel attention method. The squeeze-and-excitation network (SENet) [28] is one of these approaches, which captures channel attention for each convolutional block and achieves a noticeable performance boost for various CNN models. Wang et al. [29] realized that recording dependencies across all channels are wasteful and redundant. Therefore, they avoided the dimensionality reduction step to reduce trainable parameters and presented an ECANet that establishes channel weights by conducting a rapid 1-D convolution of size \( k \) to capture cross-channel interactions quickly. The ECA module working is as shown in Fig. 2 and discussed as follows.

By modeling the connection between the fused feature channels, the ECA structure is utilized to learn the weights of the features under several local channels. First, for the \( n \)th channel, a GAP is used to reduce the feature map size into single values that are equal to the number of channels in the convolutional layer as follows:

\[
g_n(F) = \frac{1}{L} \sum_{i=1}^{L} f_i
\]  

(5)

where \( L \) is the length of the feature sequence \( F \), following the GAP process. Features of \( g_n(F) \) have certain periodicity and correlation, which will be learned by fast 1-D CNN with a kernel size of \( K \). Fast 1-D CNN is run on \( g_n(F) \) to learn the feature weights under various channels, as shown in the following:

\[
\omega = \sigma(\text{Conv}^{1d}_k (g_n(F)))
\]  

(6)

where \( \text{Conv}^{1d} \) denotes a 1-D convolution, and to compute the kernel size \( K \) with the channel dimension \( C \) is given as follows:

\[
k = \left| \log 2(C) \right| + \frac{1}{\text{odd}}
\]  

(7)

where \( C \) is nonlinearly correlated with \( K \), and the sigmoid function is denoted by \( \sigma \). \( \sigma \) is used to compute the activation value of convolutional output to get new weights \( \omega \), which will reflect the local relationship and essential degree of the feature channel. New weights are now multiplied with each feature of the set \( F \), implying that relevant features will be given higher weights to be improved, while less important features will be given lower weights, resulting in their suppression. Thus, an improved feature map with more relevant information is obtained. As a result, the ECA module is added after every trans_conv block in the decoder of
our proposed network, so that no extraneous data from the preceding block’s feature maps are used in the reconstruction of the denoised signal. The architecture of the proposed model is described in Section III-C.

C. Proposed Model Architecture and Learning

In this study, we proposed a novel ACDAE, as shown in Fig. 3.

The network receives an AWGN-noised noisy ECG signal \( \tilde{x}_i \) and outputs a denoised ECG signal \( \hat{x}_i \). Initially, the feature maps are extracted using convolutional layers (Conv-layers) and downsampled using the max-pooling layer of two. The noise gets suppressed during this encoding task while retaining the underlying structure. Next, the transposed convolutional layers, Trans_Conv and upsampling layers, decode the compressed ECG abstraction. The dashed lines show the Conv-layers and the Trans_Conv layers symmetrically connected via skip connections. The role of the skip connections is twofold. First, they help in backpropagating the gradients to the bottom layers and also help in recovering the ECG signal details lost during the encoding and decoding process. As indicated by the red box, the ECA modules are connected after Trans_Conv_1 to Trans_Conv_3; they efficiently update the features retrieved via cross-channel interaction, allowing the network to focus on the features of relevant information between the channels. The output \( \hat{x}_i \) is then reconstructed through Trans_Conv and ECA module. The target of the training is to minimize the mean square value given by (11) between \( \hat{x}_i \) and the input \( x_i \). The smaller the loss function value, the more likely the output \( \hat{x}_i \) will reconstruct the input \( x_i \). We also experimented by placing the ECA module before Conv-2 and Conv-3 layers of the encoder, but it hardly improved the performance. Therefore, ECA modules are only placed in the decoder path to save computational time and resources. We also found that the dropout layers are not effective in the proposed model setup, as ECA helps get the attention of relevant channels, thus solving the problem of overfitting.

This study is carried out on the Keras library with Google Tensorflow 2.6, and Python 3.7. NVIDIA GeForce GTX 1070 graphics card with Cuda libraries, 16 GB of RAM, and an Intel Core i7-8700 3.20-GHz CPU are installed on the machine and segmented noisy ECG samples from channels 1 and 2 of 300 (1.2 s), each concatenated and given input to the network. On the other hand, the proposed approach can also be used with any reasonable segmented window size (e.g., 5–30 s). The normalized mean square error given in (11) is minimized during the network’s training. Keras-tuner [30] random search library is used to find the optimal hyperparameters of the neural network. The Adam algorithm, which has a learning rate of 0.0001, is chosen for optimization. A batch size of 32 is used, and epochs are set to 50 with early stopping callbacks, which is used to avoid overfitting. The network’s details are provided in Table I.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. Evaluation Metrics

Denosing performance of the ACDAE model for ECG signals is measured using three metrics: mean-squared error (MSE), SNR, and percentage-root-mean-square difference (PRD), given as follows:

\[
\text{SNR}_{\text{in}} = 10 \log_{10} \frac{\sum_{i=1}^{n} (x_i)^2}{\sum_{i=1}^{n} (x_i - \hat{x}_i)^2} \tag{8}
\]

\[
\text{SNR}_{\text{out}} = 10 \log_{10} \frac{\sum_{i=1}^{n} (x_i)^2}{\sum_{i=1}^{n} (x_i - \tilde{x}_i)^2} \tag{9}
\]

\[
\text{SNR}_{\text{imp}} = \text{SNR}_{\text{out}} - \text{SNR}_{\text{in}} \tag{10}
\]

\[
\text{RMSE}_{\text{out}} = \sqrt{\frac{1}{N} \sum_{i=1}^{n} (x_i - \hat{x}_i)^2} \tag{11}
\]

\[
\text{PRD} = \left( \frac{\sum_{i=1}^{n} (x_i - \hat{x}_i)^2}{\sum_{i=1}^{n} (\hat{x}_i)^2} \right) \times 100 \tag{12}
\]
where \( x_i \) is a sample of the original signal, \( \hat{x}_i \) is the sample of the denoised signal, and \( \tilde{x}_i \) is the sample of noise-induced ECG. MSE is used during pretraining, since it serves as a loss function for weight update. On the other side, the SNR is utilized to compare different denoising algorithms.

To evaluate the performance, performance measures, such as precision, recall, \( F_1 \)-score, and accuracy, are calculated. Formulas for calculating performance measures are given as follows:

\[
\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \tag{13}
\]

\[
\text{Precision} = \frac{TP}{TP + FP} \tag{14}
\]

\[
\text{Recall} = \frac{TP}{TP + FN} \tag{15}
\]

where TP is true positive, TN is true negative, FP is false positive, and FN is false negative.

### B. Denoising Performance of the Proposed Model

Two experiments are performed to evaluate the denoising performance of the proposed model. In the first experiment, the segmented window of AFDB and NSRDB is used as described in Section II. These databases are added with AWGN noise of different SNRs to evaluate the denoising and AF classification performance. In the second experiment, the BIHA database is noised with AWGN noise and NST database noise to evaluate the denoising performance of the model. The details are described in Sections IV-B1 and IV-B2.

1) **Denoising Performance Using AWGN:** In this experiment, 1.2-s segmented two-channel ECG signals from the AFDB and NSRDB are used for the experiment. Each window contains \( 250 \times 1.2 = 300 \) samples, where 250 Hz is the signal’s sampling frequency, two channels are concatenated, and they are fed to the model for training. ECG signals are corrupted using AWGN by varying standard deviations from \(-20\) to \(20\) dB and a \(5\)-dB step size. The AWGN stipulates the same amount of energy across all spectral bands. Denoising based on AWGN demonstrates the model’s resilience to random perturbations. As a system’s SNR decreases, its denoising potential decreases. We can observe that the signal almost lost all morphological information with an SNR of \(-10\) dB and below, as shown in Figs. 4 and 5. A random window from the test set is chosen to visualize the results of the denoised signal from \(0\) to \(-15\) dB. Here, the top figure is the original signal, the blue color shows the channel 1 signal, and the red color shows the channel 2 signal. Subsequent plots show the noised signal and their denoised output from the model ranging from \(0\) to \(-15\) dB. As depicted, the model achieves a good denoising result till \(-15\) dB, as it can reconstruct the morphology of the signal from the significant noise-induced.

However, for \(-20\) dB, as shown in Fig. 5, the model is not able to reconstruct the morphology of the signal from channel 1 as indicated by the blue color. In contrast, the morphology of the signal from channel 2 is reconstructed accurately. In such cases, training epochs is usually increased to achieve the desired result. However, because model training employs an early stopping callback with a patience value of 5, it is reasonable to infer that the model has reached its limit.

As a result, signal reconstruction at and beyond \(-15\)-dB AWGN SNR necessitates additional tuning. As previously stated, the worst case SNR that can disrupt ECG on a wearable device is \(-6\) dB; however, our model performs far better than

![Fig. 4. Denoising performance of ACDAE with different AWGNs added to ECG signal.](attachment:image)
TABLE II
COMPARISON OF ACDAE PERFORMANCE WITH AWGN ON DIFFERENT DATABASES

| Database      | -10 dB | -5 dB | 0 dB |
|---------------|---------|-------|------|
|               | SNR<sub>imp</sub> | RMSE  | PRD(%) | SNR<sub>imp</sub> | RMSE  | PRD(%) | SNR<sub>imp</sub> | RMSE  | PRD(%) |
| AFDB database | 31.87   | 0.0587 | 21.1  | 27.4   | 0.0315 | 11     | 19.31   | 0.0134 | 6.90   |
| BIHA Database | 22.06   | 0.0612 | 31.72 | 18.57  | 0.0432 | 24.1   | 14.63   | 0.0231 | 9.8    |

![Fig. 5. Denoising performance of ACDAE at -20-dB AWGN.](image)

that. As a result, we chose a range of -15- to 20-dB SNR for our proposed model to produce successful performance. Our assertion is also supported by a quantitative comparison in terms of SNR improvement and PRD, as shown in Table II.

2) Denoising Performance on NST Database: This experiment is performed using the BIHA and NST databases, as explained in Section II. For this experiment, the signal window of 10 s is taken, and the signal is resampled at 250 Hz before giving it to the model. A sample size of 10 x 250 = 2500 samples is fed to the model as input. The prepared data are then added with “em,” “bw,” “ma” noises in the same proportion for training and testing. The record numbers used for test set are 103, 107, 116, 117, 119, 124, 205, 207, 212, 220, 228, and 233. The model performance for a random window with -6- and 0-dB SNR can be seen in Fig. 6. Here, the first window represents noised signal with added NSTDB noise, and the second window represents the reconstructed signal in red and the original signal in green. Individual SNR<sub>imp</sub> as defined in (10) for the test records is shown in Fig. 7. Moreover, the BIHA database is also tested with AWGN noise as presented in Section IV-B1, and results obtained are shown quantitatively in Table II. Individual SNR<sub>imp</sub> for the test records is shown in Fig. 8. The plots are between records (on the X-axis) and SNR<sub>imp</sub> (on the Y-axis).

C. Classification Performance of the Proposed Model

Modern wearable healthcare devices, such as smartwatches and patches, are cloud-based battery-operated technologies that only collect ECG signals from the skin noninvasively and send them to a server for processing. To send large data, we will need more battery and memory, whereas compressed data use less power and memory. Therefore, we used the output of the encoder, which is a compressed representation of the original signal. The signals are compressed eight times, as can be seen from Table I. This study uses two classification modules for AF from the normal signal. The experiments are performed on two-channel ECG segment windows. A total of 600000 samples are used, with 300000 AF beats and 300000 non-AF beats. SNR values ranging from -20 to 20 dB with a step size of 5 dB are used to train and test each module, similar to the denoising task. Classification performance is evaluated using (13)–(15). M1 uses ECA attention modules for the classification task, as they can help get relevant information from the compressed feature maps, as they have the GAP layer that will average each feature map and reduce its size. These will be passed through the 1-D CNN layer with adaptive kernel size for local channel attention, suppressing the weak channel features and giving attention only to important features, as explained in Section III-B. M2 uses the FC layer followed by the GAP layer, as shown in Fig. 1. This experiment aims to see the model’s classification performance without the ECA module. Therefore, M2 employs the GAP and FC layers instead of the ECA module. The model evaluation used stratified fivefold cross validation. This means the dataset is divided into five categories with equal proportions of categorical observations. For each of the five folds, the model was trained on the four folds and validated on the fifth. This way we trained our model for five times and tested it on every data point. Each training phase had its best weights restored after 50 epochs with early stopping callbacks. The final performance is now calculated as an average of the five run.

M1 achieved the precision, recall, and accuracy of 98.76% ± 0.44%, 98.48% ± 0.58%, and 98.88% ± 0.42%, respectively, followed by M2 with 96.85% ± 0.93%, 96.58%
TABLE III
PERFORMANCE OF TWO MODULES FOR AF CLASSIFICATION

| Models       | Precision | Recall | Accuracy | Precision | Recall | Accuracy |
|--------------|-----------|--------|----------|-----------|--------|----------|
| Module 1(M1) | 98.78     | 98.50  | 99.04    | 98.74     | 98.46  | 98.72    |
| Module 2(M2) | 96.53     | 96.19  | 96.92    | 97.16     | 96.76  | 97.40    |

± 1.00%, and 97.16% ± 0.97%, respectively. Table III shows the classification performance at 10 dB of added noise. For the sake of comparison, we used a signal boost of 10 dB, because it adds very little noise. The experiment is performed for nine SNR levels for both M1 and M2, as explained in Section II, and the results are shown in Fig. 9, where the results for the AF class are at the top and the normal class at the bottom. In Fig. 9, we can see that the performance of classification models up to −10 dB is nearly identical, and for −15 dB, overall accuracy for M1 is reduced to 95.84% from 99.25%, and similarly, precision and recall also decreased to 92.51% and 97.62%, respectively; a similar trend can be seen for M2. The performance of M1 and M2 drops drastically, at −20 dB, as shown in Figs. 8 and 7. The performance boost comes at the expense of M1’s computational complexity. M1’s trainable parameters are increased to 677 920, whereas M2 has 545 325. With this experiment, we can conclude that the ECA module helps in the classification and denoising tasks, and it also shows that our denoising performance is robust till −15-dB SNR.

V. COMPARISON OF RESULTS

A. Comparison of Denoising Performance

The experimental results showed that the proposed ACDAE model performed reasonably well within the range of −15 to 20 dB. This demonstrates the model’s ability to filter ECG signals in the presence of excessive noise. The performance is evaluated using (10)–(12), with RMSE being used to measure the variance between the ACDAE model output and the original signal. PRD calculates the total distortion in the denoised signal, and SNRimp calculates the improvement in SNR between the denoised and input signal. For comparing the performance of the proposed model, a comprehensive review paper [18] is referred. Our model is solely compared with the best performing model reported in [18] and shown in Table IV. The proposed model outperformed all the published studies. This is the first time a deep learning (DL)-based is designed to filter the excessive noisy ECG signals with the SNR range of −20- to 20-dB SNR range. The worst SNR tested in the previous study was at −1-dB SNR, and Chiang et al. [14] used a 13-layer model to achieve this performance on NSTDB. However, this study did not use a −1-dB SNR, but our reported SNRimp for 0-dB SNR is 16.28 dB, and for −6 dB, it is 22.80 dB, which is better than the published study. The model used in this study is very lightweight, with four convolutional and four transposed convolutional layers, and we can denoise ECG signals with excessive noise while also recovering the morphology of the signal using a single model, as opposed to the recent work by Qiu et al. [16], in which they developed a two-stage denoising technique, one for filtering ECG signals and the other for recovering the morphology of the signal, making the model bulky and computationally expensive.

B. Comparison of Classification Performance

The performance of the proposed model achieved better results against many of the top-cited works. Our study looked at two classification approaches. The M1 uses an attention mechanism (ECA layer) just before the FC layer for classification, and M2, which uses the most commonly used GAP layer before the FC layer, with the results indicating
that the ECA layer helps to reduce FPs. M1 receives a 1.92% boost in precision, as well as a 1.93% improvement in recall and 1.74% improvement in accuracy. This work also employed Gradient-weighted Class Activation Mapping (Grad-CAM) [34], an explainable artificial intelligence technique, to compare the two classification modules. This will help us to look at how the attention module is working, and we discovered that the ECA module significantly increased attention. This is achieved by selecting a random window that was previously misclassified without the attention module (M2) and was corrected after the ECA module was incorporated (M1). In Fig. 10, the top heat map shows M1’s Grad-CAM plot, while the bottom heat map shows M2’s Grad-CAM plot for the selected window. Red indicates the maximum level of focus, while blue indicates the lowest level of concentration by the proposed network. As previously stated, the AF window will have two or more QRS complexes, whereas the N window will have two or less QRS complexes. From M1’s Grad-CAM plot, we can see that the third QRS peak receives greater attention, as well as an increase in attention at the first peak and a minor improvement at the second peak. With its capacity to change key features’ weight, we can be confident that the suggested ECA module will assist neural networks in correctly classifying while paying attention to all complexities and removing noise from the signal.

We also compared our findings to the most referenced recent papers using AFDB, which are listed in Table V. Asgari et al. [35] reported 97.1% precision and 97% recall using stationary wavelet transform (SWT), feature extraction approach, and support vector machine (SVM) to classify AF. Xia et al. [36] transformed the ECG signal into a 2-D image using SWT and employed a three-layer 2-D convolutional layer to classify AF with an accuracy of 98.63%. Petmezas et al. [37] employed a combination of CNN and LSTM with focal loss to achieve 99.29% precision, whereas [38] another work that also uses a combination of CNN and LSTM layer got 97.80% of precision. With all of the research listed, our proposed model M1 achieves better results, demonstrating its resilience.

VI. CONCLUSION

The proposed ACDAE effectively denoises the low SNR ECG signal. It outperformed the AF classification results compared with the most cited recent works. The study is comprehensively evaluated using four publicly available databases. The study uses two classification modules to test the effectiveness of an ECA module that efficiently update the features retrieved via cross-channel interaction, allowing the network to pay more attention to the features of relevant information between the channels. The model with the ECA module is found better compared with the state-of-the-art results. This study uses NST database noises and AWGN; however, real-time noise, particularly motion artifacts, may differ. As a result, the recommended strategy may not yield the same outcomes in such situations. In the future, we will refine our algorithm by running it through a slew of real-world training scenarios. This can be done by acquiring a real-time ECG database utilizing various movement and sleeping positions to validate the proposed algorithm.
REFERENCES

[1] World Health Organization, “Global status report on noncommunicable diseases 2014,” World Health Org., Tech. Rep. WHO/NMH/NVI/15.1, 2014.

[2] J. Wu et al., “Editor’s choice–impact of initial hospital diagnosis on mortality for acute myocardial infarction: A national cohort study,” Eur. Heart J., Acute Cardiovascular Care, vol. 7, no. 2, pp. 139–148, Mar. 2018.

[3] A. Abdelazez, S. Rajan, and A. D. C. Chan, “Detection of atrial fibrillation in compressively sensed electrocardiogram measurements,” IEEE Trans. Instrum. Meas., vol. 70, pp. 1–9, 2021.

[4] P. Bera, R. Gupta, and J. Saha, “Preserving abnormal beat morphology in long-term ECG recording: An efficient hybrid compression approach,” IEEE Trans. Instrum. Meas., vol. 69, no. 5, pp. 2084–2092, May 2020.

[5] R. J. Martis, U. R. Acharya, and L. C. Min, “ECG beat classification using PCA, LDA, ICA and discrete wavelet transform,” Biomed. Signal Process. Control, vol. 8, no. 5, pp. 437–448, 2013.

[6] S. Oowski and T. H. Linh, “ECG beat recognition using fuzzy hybrid neural network,” IEEE Trans. Biomed. Eng., vol. 48, no. 11, pp. 1265–1271, Nov. 2001.

[7] S. Mirza, K. Bhole, and P. Singh, “Fetal ECG extraction and QRS detection using independent component analysis,” in Proc. 16th IEEE Int. Symp. Signal Process. Appl. (CISS), Feb. 2020, pp. 57–61.

[8] P. Singh, K. Bhole, and A. Sharma, “Adaptive iteration techniques for impulsive noise removal from ECG,” in Proc. 14th IEEE India Council Int. Conf. (INDICON), Dec. 2017, pp. 1–4.

[9] J. P. Martínez, R. Almeida, S. Olmos, A. P. Rocha, and P. Laguna, “A wavelet-based ECG delineator: Evaluation on standard databases,” IEEE Trans. Biomed. Eng., vol. 51, no. 4, pp. 570–581, Apr. 2004.

[10] M. 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.

[11] Z. Golrizkhatami and A. Acan, “ECG classification using three-level deep convolutional encoder–decoder networks with symmetric skip connections,” in Deep Learning for Biomedical applications, vol. 42, no. 11, Nov. 2021, Art. no. 115002.

[12] S.-Y. Chien, “Noise reduction in ECG signals using fully convolutional denoising autoencoders,” IEEE Trans. Biomed. Eng., vol. 63, Jan. 2021, Art. no. 102194.

[13] B. Hou, J. Yang, P. Wang, and R. Yan, “LSTM-based auto-encoder model for ECG arrhythmias classification,” IEEE Trans. Instrum. Meas., vol. 69, no. 4, pp. 1232–1240, Apr. 2020.

[14] P. Xiong, H. Wang, M. Liu, F. Lin, Z. Hou, and X. Liu, “A stacked denoising auto-encoder for ECG signal denoising,” Physiol. Meas., vol. 37, no. 12, pp. 2214–2230, 2016, doi: 10.1088/0967-3334/37/12/2214.

[15] 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.

[16] H.-T. Chiang, Y.-Y. Hsieh, S.-W. Fu, K.-H. Hung, Y. Tsao, and S.-M. Chen, “Noise reduction in ECG signals using fully convolutional denoising autoencoders,” IEEE Access, vol. 7, pp. 60806–60813, 2019.

[17] F. P. Romero, D. C. Piñol, and C. R. Vázquez-Seisdedos, “Deepfilter: An ECG baseline wander removal filter using deep learning techniques,” Biomed. Signal Process. Control, vol. 70, Sep. 2021, Art. no. 102992.

[18] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, “ECA-Net: Efficient channel attention for deep convolutional neural networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 13–19, doi: 10.1109/CVPR42600.2020.00115.

[19] T. O’Malley et al. (2019). Kerastuner. [Online]. Available: https://github.com/keras-team/keras-tuner.

[20] S. Samadi and M. B. Shamsollahi, “ECG noise reduction using empirical mode decomposition based on combination of instantaneous half period and soft-thresholding,” in Proc. 2nd Middle East Conf. Biomed. Eng., Feb. 2016, pp. 244–249.

[21] H. D. Hesar and M. Mohebbi, “An adaptive particle weighting strategy for ECG denoising using marginalized particle extended Kalman filter: An evaluation in arrhythmia contexts,” IEEE J. Biomed. Health Informat., vol. 21, no. 6, pp. 1581–1592, Nov. 2017.

[22] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-CAM: Visual explanations from deep networks via gradient-based localization,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 618–626.

[23] S. Asgari, A. Mehnia, and M. Moussavi, “Automatic detection of atrial fibrillation using stationary wavelet transform and support vector machine,” Comput. Biol. Med., vol. 60, pp. 132–142, May 2015.

[24] Y. Xia, N. Wulan, K. Wang, and H. Zhang, “Detecting atrial fibrillation by deep convolutional neural networks,” Comput. Biol. Med., vol. 93, pp. 84–92, Feb. 2018.

[25] G. Petmezas et al., “Automated atrial fibrillation detection using a hybrid CNN-LSTM network on imbalanced ECG datasets,” Biomed. Signal Process. Control, vol. 63, Jan. 2021, Art. no. 102194.

[26] R. S. Andersen, A. Peimankar, and S. Puthusserypady, “A deep learning model for ECG arrhythmias classification,” IEEE Access, vol. 8, pp. 143802–143817, 2020.

[27] J. Yu, X. Liu, and L. Ye, “Convolutional long short-term memory autoencoder-based feature learning for fault detection in industrial processes,” IEEE Trans. Instrum. Meas., vol. 70, pp. 1–15, 2021.

[28] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, “Extracting and composing robust features with denoising autoencoders,” in Proc. Int. Conf. Mach. Learn., 2008, pp. 1096–1103.

[29] J. Lu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 7132–7141.

[30] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, “ECA-Net: Efficient channel attention for deep convolutional neural networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 13–19, doi: 10.1109/CVPR42600.2020.00115.

Prateek Singh (Student Member, IEEE) received the B.Tech. degree in electronics and instrumentation engineering from Gautam Buddha Technical University, Lucknow, India, in 2011, and the M.Tech. degree in biomedical instrumentation engineering from the College of Engineering (COE), Pune, India, in 2015. He is currently pursuing the Ph.D. degree with the Department of Electrical Engineering, Indian Institute of Technology (IIT) Roorkee, Roorkee, India.

He was a Senior Research Fellow with Virtual Labs, IIT Roorkee, from 2016 to 2017. His research interests include signal processing, artificial intelligence (AI), and ethical AI.

Ambalika Sharma received the B.E. and M.E. degrees from the University of Roorkee, Roorkee, India, in 1983 and 1988, respectively, and the Ph.D. degree from Annamalai University, Chidambaram, India.

She is currently an Assistant Professor with the Department of Electrical Engineering, Indian Institute of Technology Roorkee, Roorkee. Her research interests include biomedical engineering, signal processing, and neural networks.