IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 15, 2020, accepted May 21, 2020, date of publication June 1, 2020, date of current version June 16, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2998788

DeepArrNet: An Efficient Deep CNN Architecture for Automatic Arrhythmia Detection and Classification From Denoised ECG Beats

TANVIR MAHMUD[1], (Student Member, IEEE), SHAIKH ANOWARUL FATTAH[1], (Senior Member, IEEE), AND MOHAMMAD SAQUIB[2], (Senior Member, IEEE)

[1]Department of Electrical and Electronic Engineering, Bangladesh University of Engineering and Technology, Dhaka 1000, Bangladesh
[2]Department of Electrical Engineering, The University of Texas at Dallas, Richardson, TX 75080, USA

Corresponding author: Shaikh Anowarul Fattah (fattah@eee.buet.ac.bd)

ABSTRACT In this paper, an efficient deep convolutional neural network (CNN) architecture is proposed based on depthwise temporal convolution along with a robust end-to-end scheme to automatically detect and classify arrhythmia from denoised electrocardiogram (ECG) signal, which is termed as 'DeepArrNet'. Firstly, considering the variational pattern of wavelet denoised ECG data, a realistic augmentation scheme is designed that offers a reduction in class imbalance as well as increased data variations. A structural unit, namely PTP (Pontwise-Temporal-Pointwise Convolution) unit, is designed with its variants where depthwise temporal convolutions with varying kernel sizes are incorporated along with prior and post pointwise convolution. Afterward, a deep neural network architecture is constructed based on the proposed structural unit where series of such structural units are stacked together while increasing the kernel sizes for depthwise temporal convolutions in successive units along with the residual linkage between units through feature addition. Moreover, multiple depthwise temporal convolutions are introduced with varying kernel sizes in each structural unit to make the process more efficient while strided convolutions are utilized in the residual linkage between subsequent units to compensate the increased computational complexity. This architecture provides the opportunity to explore the temporal features in between convolutional layers more optimally from different perspectives utilizing diversified temporal kernels. Extensive experimentations are carried out on two publicly available datasets to validate the proposed scheme that results in outstanding performances in all traditional evaluation metrics outperforming other state-of-the-art approaches.

INDEX TERMS Arrhythmia, biosignal analysis, CNN, deep learning, disease detection, ECG.

I. INTRODUCTION

Cardiovascular diseases (CVDs) have become one of the most common causes of death throughout the world in recent times. Early recognition of cardiac abnormality is vital for proper treatment before occurring any major irreversible damages. Among various CVDs, the arrhythmia is one of the most common problems that describes irregularity and abnormality in heart beats [1]. There are various types of arrhythmia, such as ventricular fibrillation, premature atrial contraction and supra-ventricular arrhythmia [2], [3]. Electrocardiogram (ECG) signal, a recording of the heart's electrical potential to show the electrical activity of the heart, is most widely used by physicians to check the proper functionality of the heart. Arrhythmia detection based on manual inspection of ECG signals by experts is the commonly used approach which is often complicated, time-consuming, human error-prone and difficult due to lack of experts.

For arrhythmia detection and classification, a number of methods have already been presented in the literature ranging from the traditional feature-based machine learning process to the end-to-end deep learning process in recent times. In feature-based arrhythmia detection techniques [4]–[13], various feature extraction approaches are employed, such as wavelet transform [6]–[10], principal component analysis [12], independent component analysis [13] and Hermite

The associate editor coordinating the review of this manuscript and approving it for publication was Derek Abbott.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

function [11]. For performing classification with the extracted features, support vector machine (SVM) [4], K-nearest neighbour [5], feed-forward neural network [8], [11], [13] and random forest [6] have been used. These approaches mostly depend on handcrafted feature extraction process that most often leads to loss of information required for the classification due to improperly chosen features or inadequate features. Automating the process of feature extraction and classification was the primary motivation behind the popularity of end-to-end deep learning-based frameworks.

A number of deep learning-based approaches have also been adopted recently for arrhythmia classification [14]–[26]. In [14]–[19], 1D convolutional neural network (CNN) and in [23]–[25], recurrent neural network (RNN) and LSTM network are employed while in [20]–[22], 2D CNN is used by converting 1D ECG beats into 2D images. Most of the deep learning-based approaches are facing some common issues: (1) raw ECG data collected from patients are being directly fed to the deep neural network making the classification process complicated due to presence of various low and high-frequency noises, (2) for dealing with 1D ECG signal, data augmentation is not necessarily used and even if it is used, natural variational pattern of ECG isn't properly captured or preserved, and (3) most of the approaches use very deep CNNs with large number of parameters that not only increase the computational complexity but also lead to overfitting the model to training data. Hence, a deep CNN based arrhythmia classification scheme which can overcome the above problems and can provide very satisfactory classification performance with low computational burden is still in great demand.

In this paper, a deep CNN architecture is proposed for automating the arrhythmia classification process, termed as 'DeepArrNet', where a PTP (Pointwise-Temporal-Pointwise) convolution structural unit along with its variants are introduced exploiting the advantage of depthwise separable convolution for 1D time-series data with multi-stage implementation. In the proposed architecture, multistage PTP blocks along with dimension reduction stages are introduced utilizing combination of features that are sequentially learned from different PTP blocks by employing residual connections between PTP blocks. Furthermore, the proposed architecture overcomes the need for dimension reduction block by using several strided spatial convolution in parallel within each PTP block. This architecture requires a significantly low number of model parameters while providing consistent feature characteristics. In the pre-processing stage, an adaptive wavelet threshold scheme is used for denoising. Moreover, a realistic augmentation scheme is developed considering the natural variation in ECG pattern that reduces the problem of class imbalance and increases the diversity in the dataset while preserving the original features of the beats. The performance of the proposed method is tested on two publicly available ECG arrhythmia datasets along with extensive comparative analysis.

II. PREPROCESSING

The raw data collected from patients need to be pre-processed first to make it compatible with the deep neural network. This pre-processing stage consists of five different operations. Each of them is described in detail below.

A. WAVELET BASED DENOISING

The input raw ECG signal, $x[n]$ can be expressed as:

$$x[n] = \hat{x}[n] + v[n] \tag{1}$$

where $\hat{x}[n]$ is the original clean ECG signal that has a certain pattern, and $v[n]$ is the additive noise present in the raw data–generally random in nature–may significantly vary during train and test phase. If $x[n]$ is used for training, there is a chance of getting poor performance during the test even with the highly trained model due to the random nature of $v[n]$. Instead of using noisy raw data, if noise reduction is possible to achieve clean $\hat{x}[n]$, much better training and testing performance is achievable.

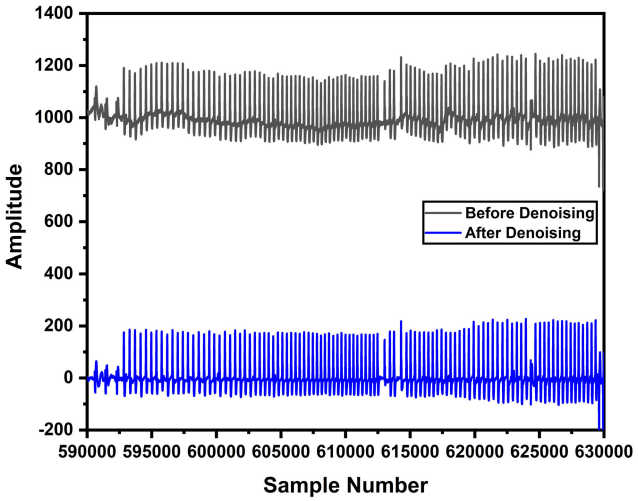


FIGURE 1. A portion of ECG data collected from a patient is shown before and after denoising. Raw data seems to contain high frequency noises with large base line shift and denoising process minimizes these effects of noises.

ECG signals are corrupted by different types of noises, such as baseline wandering, power line interference, electromyogram (EMG) noise, electrode motion artifacts, and channel noise. Different noise reduction techniques are used in literature for removing noises from recorded ECG data [27]–[31]. Among them, wavelet transformation, a time-scale representation method, decomposes signals into basis functions of time and scale that makes it useful for data denoising. In this work, a wavelet transform based approach with soft thresholding scheme has been employed for the removal of the effects of noise. In Fig 1, the effect of denoising on a segment of raw data is shown.

B. R PEAK DETECTION

Following the denoising, the R peak of each ECG beat is located from the continuous beat stream. Various methods are proposed for R peak detection in the literature [32]–[38]. As reported in [37], it provides very fast and precise detection of R-peak. Hence, this algorithm has been employed in this study. In this method, the denoised ECG signal is squared, firstly, to enhance large values and boost high-frequency components. Next, blocks of interest are generated using two event-related moving average to extract the QRS features and to extract the QRSs beat, respectively. Afterward, an even related threshold is applied to the generated blocks to separate the blocks that contain the R-peak from the blocks that include noise. Finally, the maximum absolute value of each separated block is identified that provides the R-peak index.

C. BEAT EXTRACTION

To process the ECG beats using deep neural networks, all the beat length should be uniform. In [21], the median of the R-R intervals is considered as the nominal period to segment each beat by maintaining equal length in both sides of the R peak. After segmenting each beat, zero padding is done to make the length uniform. In [17], [22], [23], equal length of portion is cropped centering the R-peak as an individual beat. In this work, each beat is segmented centering the R peak by cropping at the midpoint of the adjacent R-R intervals. Following that, further cropping or padding with the edge values is carried out centering the R-peak to make the length of each beat uniform as in some cases, the extracted beat length becomes larger or smaller than the predefined length. This process can be described as follows.

1) After R-peak detection, each beat is extracted centering the detected R peak and cropping at two adjacent edges depending on the position of adjacent R peaks. The cropping edges are decided to be the midpoint of adjacent R-R intervals. Hence, if the sample number of the extracted beat is denoted by $\mathbf{n}$ and a, b, c representing the sample containing three adjacent R-peaks while b is representing the R-peak of the beat to be segmented, the range of the segmented beat can be written as,

$$n_{\min} = (a + b)/2 \tag{2}$$

$$n_{\max} = (b + c)/2 \tag{3}$$

2) As some of the segmented beats will be smaller/larger than the predefined length of beats, it is needed to be equalized for further processing on deep neural networks. Hence, by centering the R peak of the extracted beat, further cropping is done at the edges, if the beat is larger at the edges. Otherwise, padding with the edge value is carried out if the beat is shorter.

D. BEAT AUGMENTATION

Due to the scarcity of data for rare diseases compared to normal cases, data imbalance is a common problem in almost every biomedical application. In ECG, the imbalance is more prominent due to the lack of arrhythmia beats as many of the beats are normal even for a patient with cardiovascular disease. With such imbalance in the dataset, the trained model is prone to overfit with the large normal class, as it considers diseases with a smaller number of data as outliers. For 1D ECG signal, the problem still persists as the various augmentation techniques have not that much explored. In [22], 1D ECG data are converted to 2D images and some augmentations are done with different cropping techniques. Due to the 1D properties of ECG data, such techniques don't provide that many variations, especially in the case of smaller classes. In [23], the SMOTE augmentation technique is used which mainly operates through interpolation within various classes. As the position and shapes of the data still vary a significant amount within a class, such interpolations often lead to severe distortions in the generated synthetic data that result in a false representation of the original data class. In our previous work [18], we have provided different augmentation techniques for ECG signals. In this work, we have increased the augmentation techniques by modifying and combining these operations in an algorithmic way to incorporate more realistic variations in the dataset. The following steps are carried out sequentially while performing augmentations.

- **Step 1 (Amplitude Scaling):** The data, firstly, undergoes through amplification, attenuation or no operation. For amplification/attenuation, the scaling factor is chosen randomly from a range that is empirically determined. As in practical cases, there exist some variations in the relative value of amplitudes in ECG data, these introduced variations through amplitude scaling offer good augmented beats.
- **Step 2 (Time Scaling):** Next, the amplitude scaled beat undergoes through dilation, contraction or no operation on the time axis. For dilation, the beat is first oversampled followed by cropping operation while for contraction, the beat is under-sampled followed by padding operation. The time scaling factor is randomly chosen from an empirically selected range. As in practical cases, such dilations or contractions in beats are widely visible within a certain limit and the original morphology of the beat isn't changed, it offers a nice technique to introduce variations in the dataset.
- **Step 3 (Shifting):** Finally, the beat undergoes through left shifting, right shifting or no operation. After shifting, some samples of the particular beat are cropped at one edge while some samples are padded at the other edge. The number of samples shifted is chosen randomly from a predefined range. This operation leads to some variations in the information content that leads to making the classification action slightly challenging while demanding more priority to generalize the global features.

By iterating through various choices of these steps, 26 different combinations of operations are performed on the original beat that provides a more realistic representation of the synthesized beats. In Fig. 2, all such operations on a particular beat are shown.

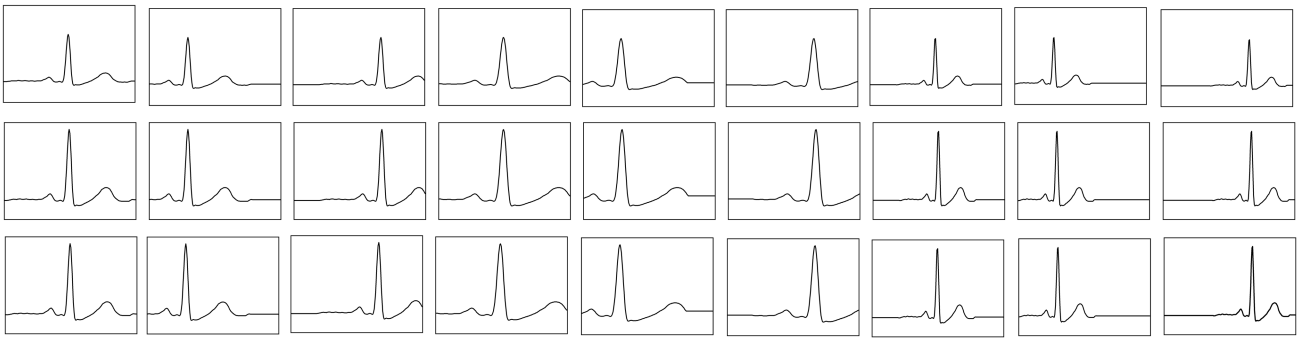


FIGURE 2. Augmentation on a right bundle branch block beat by applying 26 different combinations of operation on the original beat generated by the proposed augmentation scheme.

E. DATA NORMALIZATION

At the final stage of pre-processing, all the original and augmented beats in the training and testing dataset are normalized as normalization fastens the convergence of the neural network by increasing the stability of the model through confining the learning within a fixed range [39]. Min-max normalization is applied to map the whole dataset in the range of [0, 1], which is defined by:

$$X_{\text{normalized}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{4}$$

where $X_{\min}$ and $X_{\max}$ are the smallest and largest values of the samples present in the beat respectively.

III. PROPOSED DEEP NEURAL NETWORK

Once the pre-processed ECG beats are extracted, the next objective is to develop efficient deep convolutional neural network architecture for arrhythmia classification. In this case, instead of using traditional convolution, first, a structural unit is designed based on depthwise separable convolution in the 1D domain and then, a new deep CNN architecture is proposed utilizing the unit. At first, the background and opportunity of depthwise separable convolution for 1D data are discussed. Then, the topology and formation of the proposed deep neural network is presented in detail. In the end, the choice of optimization and learning methodologies including choice of various hyper-parameters are discussed.

A. DEPTHWISE SEPARABLE CONVOLUTIONS FOR 1D SIGNAL

For conventional 2D convolution operation, both spatial convolution of each channel and the inter-channel convolution are performed jointly at the same time. This increases the number of arithmetic operations exponentially with the larger size of kernels. This overhead in computational cost also results in a larger network that becomes prone to overfitting the training data.

In depthwise separable 2D convolution, the spatial convolution and inter-channel convolution operations are performed separately. Generally, at first, a spatial convolution is done on each channel separately and then, pointwise convolution is carried out considering the inter-channel information together. This spatial convolution followed by pointwise convolution is jointly termed as separable convolution. This type of convolution offers a similar transformation with a small number of model parameters compared to traditional convolution. It was first proposed in [40] and is widely used for 2D image analysis [41]–[44]. Recently, depthwise separable convolution has also been incorporated for processing speech signals [45], [46].

The ECG signal is a 1D signal and in this case, the depthwise separable 2D convolution needs to be modified. Instead of block-wise spatial convolution on each 2D channel, simple 1D temporal convolution is required to capture temporal information individually. To classify among different classes of arrhythmia with a small size of available data for most of the abnormal classes, the network is highly prone to overfit with the large normal class. Hence, that opens the door of optimization between network overall capacity, i.e. the number of parameters and the network diversity to capture the minuscule difference in features of smaller classes. By utilizing the depthwise separable convolution operation in the 1D domain for ECG arrhythmia classification, both of these objectives can be achieved.

For better understanding, a comparison between traditional 1D convolution and depthwise separable 1D convolution in terms of computational complexity and required number of network parameters is presented here:

Let's consider an input data of length l_i with C_i number of channels. Standard 1D convolution with kernel size k can be used to transform from input data (l_i, C_i) to output data (l_i, C_o), with C_o number of channels, which requires a computational cost of χ_s with ζ_s number of network parameters where:

$$\chi_s = l_i \times C_i \times C_o \times k \tag{5}$$

$$\zeta_s = k \times C_i \times C_o \tag{6}$$

In case of depthwise separable 1D convolution, the temporal 1D convolution requires a computational cost of $l_i \times C_i \times k$ with $k \times C_i$ number of network parameters and the pointwise

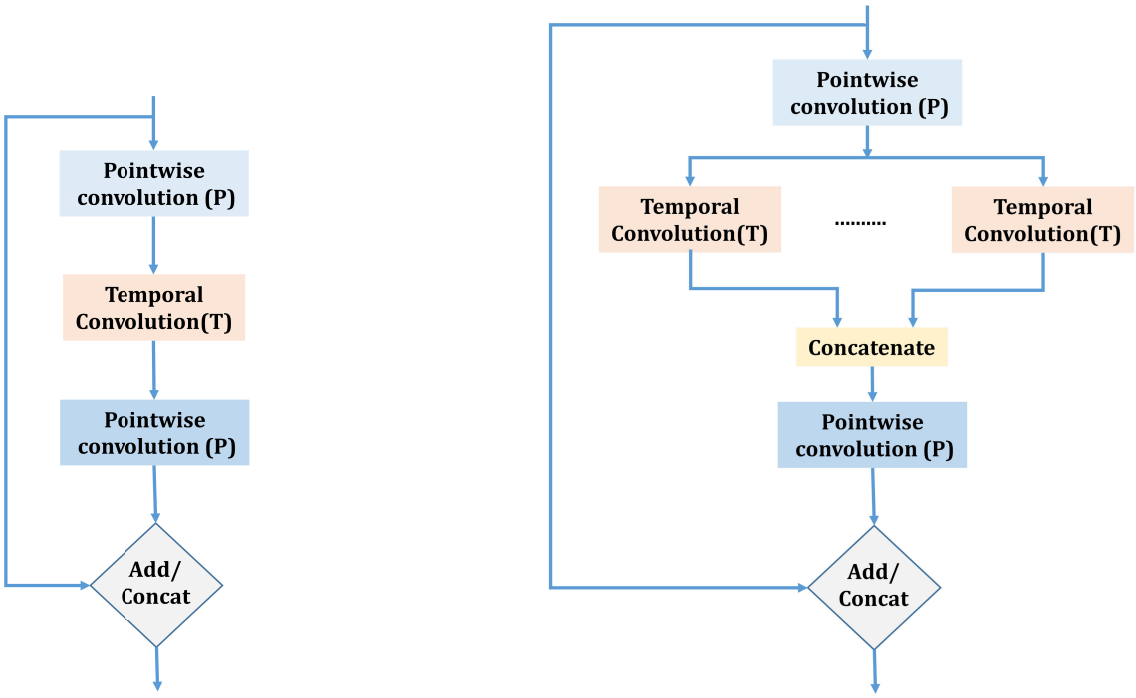


FIGURE 3. Proposed structural units utilizing (a) single temporal convolution, (b) multiple temporal convolution with various kernels in parallel with prior and post pointwise convolution.

convolution requires a computational cost of $C_i \times C_o \times l_i$ with $C_i \times C_o$ number of network parameters which results in total cost of χ_d with total network parameter of ζ_d where:

$$\chi_d = l_i \times C_i \times k + C_i \times C_o \times l_i \tag{7}$$

$$\zeta_d = k \times C_i + C_i \times C_o \tag{8}$$

Here, the depth of the output feature map is changed by a factor of C_0/C_i.

As a result, the reduction in computational cost is

$$\frac{\chi_d}{\chi_s} = \frac{1}{C_o} + \frac{1}{k} \tag{9}$$

and the reduction in the number of network parameter is

$$\frac{\zeta d}{\zeta s} = \frac{1}{C_o} + \frac{1}{k} \tag{10}$$

Therefore, depthwise separable convolution offers performance comparable to traditional convolution with a large reduction in computational cost with a smaller number of network parameters in case of 1D signals for the increased number of channels with the larger kernel.

B. PROPOSED STRUCTURAL UNIT

Based on depthwise separable 1D convolution, a structural unit is designed where before and after the depthwise temporal convolution, inter-channel pointwise convolutions are performed. A simplified schematic of the proposed structural unit is shown in Fig. 3a. The motivation and purpose of different convolution operations in this structural unit are described below.

- At first, a pointwise convolution is performed to combine the inter-channel input data information and project the information on a larger space with an increased number of channels. The depth increase factor is chosen empirically for proper optimization.
- Next, to capture temporal information from each channel, a separate temporal convolution is carried out in the deeper feature map following the prior pointwise convolution. The kernel dimension for this temporal convolution can be varied.
- After that, another pointwise convolution is performed to merge the temporal information of different channels and to project the extracted feature information in a smaller space. The dimension reduction factor of the final space is empirically selected to provide new features from each structural unit after performing the temporal convolution from a larger window.
- Finally, these extracted new features will be combined with the input features through concatenation or addition before entering into the deeper structural unit. Such operations offer the opportunity to go deep with such units while reducing the vanishing/exploding gradient problems by establishing linkage between output and input feature map.

An alternate unit structure is shown in Fig. 3b where instead of doing single temporal convolution with larger

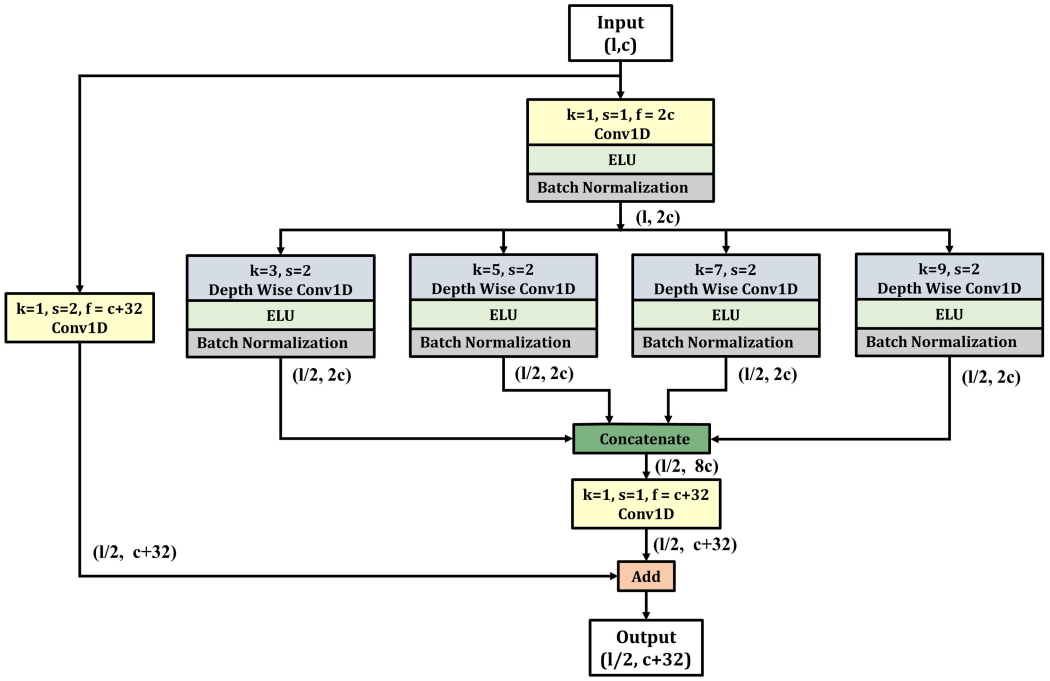


FIGURE 4. DeepArrNet Unit Block utilizing multiple temporal convolutions with various kernels in parallel. Here, 'l' stands for tensor length, 'c' for number of channels while 'k' stands for kernel size, 'f' for number of filters and 's' for the strides in the convolution.

kernels in deeper sequential structural units, multiple temporal convolutions with varying kernel dimensions are performed in parallel utilizing the broadened feature map from the first pointwise convolution. This will pave the way to combine various temporal correlation collected from smaller to broader time windows (by varying kernel dimension) at the same time with a small increase in computational cost as this temporal convolution will be performed separately on each channel. In this case, all the outputs from temporal convolutions are concatenated depthwise before the final pointwise convolution. Similar to Fig. 3a, the output of final pointwise convolution can be added/concatenated with the input feature map.

In the development of the entire architecture, this structural unit can be used (where temporal kernel dimension may be varied) repeatedly to incorporate features from a broader spectrum. Utilizing the variations in proposed structural units, different deep convolutional neural network architectures can be designed for arrhythmia detection and classification. In this study, the most effective and efficient form with multiple parallel temporal kernels (Fig. 3b) is used for the construction of 'DeepArrNet' architecture which is discussed with implementation details as below.

C. PROPOSED DeepArrNet ARCHITECTURE

In the proposed DeepArrNet architecture, to make more efficient use of temporal and pointwise convolution, based on the proposed structural unit shown in Fig. 3b, a 'DeepArrNet Unit Block' is designed which is presented in Fig. 4. In this block, instead of using a single temporal convolution operation, multiple temporal convolution operations are performed in parallel using various kernel sizes at the same time. In order to limit the computational complexity in parallel temporal convolution operations, a strided temporal convolution is performed which also reduces the length of the output feature map. As a result, this will reduce the computational complexity in the subsequent stages of the proposed DeepArrNet architecture as well, while extracting more generalized features combining various temporal windows. The detail description of the operations performed in 'DeepArrNet Unit Block' is summarized below.

- At first, the input data undergo through pointwise convolution with depth increase factor of 2 and nonlinear activation function followed by normalization.
- Following that, the data are passed through four parallel paths to perform separate temporal convolution with strides of 2 on each of them considering varying temporal kernel dimensions (e.g. kernel sizes of 3, 5, 7 and 9 are chosen here). This strided multi-kernel temporal convolution operations not only reduce the computational complexity but also provide adequate temporal information extracted from varying observation perspective.
- Next, after undergoing through nonlinear activation and normalization, the output feature maps from all four paths are concatenated vertically which causes an increase in the number of channels.

- Thereafter, pointwise convolution is performed to project the resultant concatenated features on a smaller space. Here, the number of channels in the output feature map is increased by 32 from the input feature map.
- Finally, the output of this pointwise convolution is added with the input feature map after being passed through a strided pointwise convolution operation to generate the final output feature map.

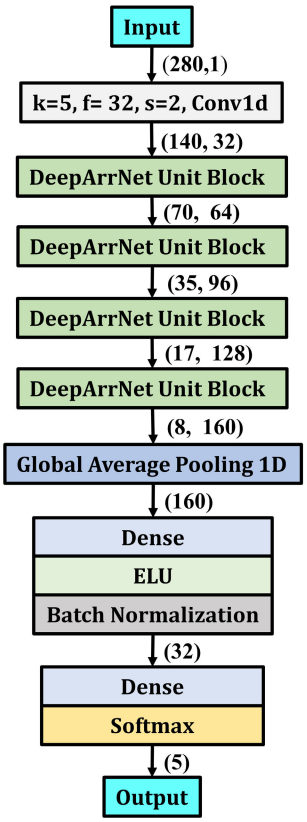


FIGURE 5. Architecture of the Proposed DeepArrNet. Here, 'k' stands for kernel size, 'f' for number of filters and 's' for strides in the convolution.

The complete architecture of the proposed DeepArrNet is presented in Fig. 5 where the input data are first passed through a standard convolution block. Next, the output is fed to consecutive four 'DeepArrNet Unit Block's, In each block, the length of the transformed feature map is reduced while increasing the depth. Hence, after passing through these blocks, final features are obtained with a reduced length of 8 with a depth of 160 channels. After that, the global average 1D pooling operation followed by traditional classification operations are performed.

On the other hand, different architectures can be formed utilizing the basic structural unit, shown in Fig. 3a, where single temporal convolution is used instead of multiple parallel temporal convolutions. To increase the diversity of the feature extraction process, various temporal kernels can be used in subsequent structural units. Moreover, computational burden will be smaller in this form of networks so that more number of such structural units can be employed in the deep network. Nonetheless, the proposed scheme with multiple parallel temporal convolutions provides optimum performance for more diversified feature extraction with effective use of numerous temporal kernels.

Therefore, the proposed DeepArrNet architecture is capable of performing the depthwise temporal and pointwise convolutions efficiently to merge features from different observation windows. Consequently, this results in a very light deep neural network, which can also capture the complex functionality of the data to distinguish among minuscule variations in features of different classes effectively.

D. CHOICE OF OPTIMIZATION AND LEARNING TECHNIQUES

Various optimization and learning techniques have been carried out in the proposed architecture to increase the convergence of the network. Some of them are discussed below.

1) ACTIVATION FUNCTIONS

Among various activation functions used to introduce non linearity in between layers of convolutions, exponential linear unit (ELU) activation is found to perform better for ECG beat analysis. Therefore, ELU activation is extensively used in our proposed networks that can be represented by:.

$$\text{ELU}: f(\alpha, x) = \begin{cases} \alpha(e^x - 1), & \text{if } x \leq 0 \\ x, & \text{if } x > 0 \end{cases} \tag{11}$$

In this work, $\alpha = 0.2$ is chosen to converge faster. For performing the final classification, softmax activation is used to make the prediction probability of one of the classes to be close to 1 while making others to be close to zero. The standard (unit) softmax function $\sigma: \mathbb{R}^k \rightarrow \mathbb{R}^k$ is defined by the formula:

$$\sigma(z)_i = \frac{e^{zi}}{\sum_{j=1}^{K} e^{zi}}$$
$$\text{for } i = 1, \ldots, K \text{ and } z = (z_1, \ldots, z_k) \in \mathbb{R}^k \tag{12}$$

2) KERNEL INITIALIZATION

A number of kernel initializers are proposed in the literature, such as random initialization, Xavier initialization, and He-Normal initialization [47]–[49]. In particular, Xavier [47] and He-normal [48] initializations are extensively used in modern deep neural networks. As the He-normal initializer performs better with ReLU activation and ELU activation has been employed which is a variation of ReLU, He-normal initialization has been applied in our proposed CNN architectures.

3) LEARNING METHODOLOGY

Though a number of cost functions are proposed for numerous operations, the cross-entropy cost function is widely used for classification that can be shown as:

$$J = -\frac{1}{N}\left[y_n \log(\hat{y}_n) + (1 - y_n)\log(1 - \hat{y}_n) \right] \tag{13}$$

TABLE 1. AAMI classes for different types of beat from MIT BIH database.

The AAMI heartbeat class description	N all other hearbeats except S,V, F and Q class	SVEB Supraventricular ectopic beat	VEB Ventricular ectopic beat	F Fusion beat	Q Unknown beat
MIT-BIH heartbeat types	Normal beat(N) Right bundle branch block beat(R) Left bundle branch block beat(L) Atrial escape beat(e) Nodal escape beat(j)	Atrial premature beat(A) Nodal premature beat(J) Aberrated atrial premature beat(a) Supraventricular premature beat(S)	Ventricular escape beat(E) Premature ventricular contraction(V)	Fusion of normal and ventricular beat(F)	Paced beat(P) Fusion of paced and normal beat(f) Unclassified beat(U)

where J is the total cost, N is the number of training data, y_n is the expected output and $\hat{y}_n$ is the actual output generated by the network.

To optimize the cost function, Adam optimizer with a batch size of 64 is used which is a first-order gradient-descent based optimizer. For Adam optimizer, the initial learning rate is chosen to be 0.001 and the exponential decay rates for the first and second moments are chosen to be 0.9 and 0.999 respectively.

4) REGULARIZATION

To increase generalization of the trained network, various regularization techniques, such as L1 and L2 regularization [50], dropout [51] and batch normalization [51] have been proposed in the literature. In this work, batch normalization is applied immediately after the non-linear activation function. Though batch normalization is usually carried out before activation, through experimentation, it has been found to perform better if placed after nonlinear activation.

IV. RESULTS AND DISCUSSION

In this section, for the purpose of demonstrating the performance of the proposed method experimental results are presented along with performance comparison and detail analysis on the effects of various parameters on the performance. Two publicly available datasets are used to carry out the experimentation. Description of the datasets and method of training/testing are first discussed. Next, the results and analysis are presented focusing on major findings.

A. DESCRIPTION OF THE DATASET

In this work, a very widely used publicly available MIT BIH arrhythmia dataset is used for analyzing the ECG beats [52]. The dataset contains 48 half-hours of two-channel (MLII and V1) ECG recordings collected from 47 patients, which are digitized at 360 samples per second per channel with 11-bit resolution. There are approximately 110,000 ECG beats in this dataset and the beats are classified into five broad categories by the Association for the Advancement of Medical Instrumentation (AAMI) [3] as presented in Table 1. For notational simplicity, the five categories are denoted with five alphabets, such as N (normal and some other classes), S (supraventricular), V (ventricular), F (Fusion) and Q (Unknown). However, for details about the categories Table 1 needs to be considered. The beat annotation provided by the AAMI is adopted in this paper. It is to be noted that out of two channels, similar to most of the research works, only MLII channel is considered as it provides better information regarding the condition of the heart [53]. Considering the high degree of imbalance among the five categories N, S, V, F, and Q in the dataset represented visually in Fig. 6, different types of augmentations are employed in the training phase. Computational tools used for the purpose of implementing the proposed deep CNN architectures are presented with detail specifications in Table 2.

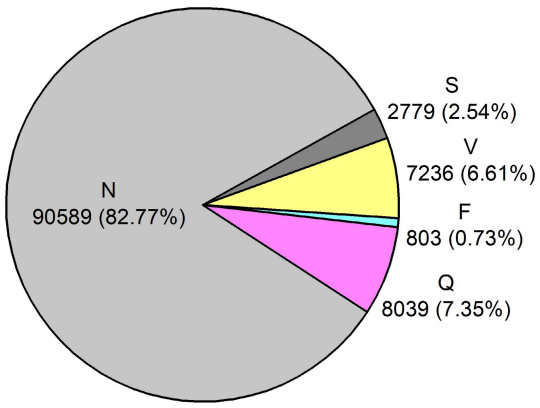


FIGURE 6. Visual representation of the large imbalance present in the number of beats from different classes.

TABLE 2. Computational tools used for the experimentation.

Resources	Specifications
CPU	Intel®Xeon®D-1653N CPU @2.80GHz with 12M Cache and 8 cores
RAM	12.72 GB
DISK	2 TB (partially used)
GPU	1 x NVIDIA Titan RTX having with 4608 CUDA cores running @1770 MHz, 24 GB GDDR6 memory
Languages and Packages	Python with Tensorflow

PTB database [54], another publicly available dataset, is also used for this study. It contains 290 records of which 148 are diagnosed as MI, 52 are healthy and rest ones are with 7 different diseases. Each record contains ECG signals sampled to the sampling frequency of 1000Hz.

B. EXPERIMENTATION

All the available data are randomly divided into 10 equal folds for 10-fold cross-validation scheme. Next, 9 of the folds are considered for training, while the remaining one is considered for testing at a time. All the proposed augmentation methods are carried out on the training folds while keeping the test fold unaltered to make the proper evaluation of the test scenario. Similarly, this process is repeated for 10 times by changing the test and train fold and the network is trained and tested by executing the same process. Finally, all the measured metrics on each test fold are averaged to get the final cross-validation score.

For performance evaluation of multi-class arrhythmia classification methods, commonly used four performance criteria are used, namely accuracy, sensitivity, positive predictive value (PPV) and F1 score [55].

C. PERFORMANCE EVALUATION

At first, the performance of the proposed architecture is analyzed from different perspectives. Later, the performances of some existing approaches are compared with that of the proposed method.

1) ANALYSIS OF THE PROPOSED ARCHITECTURE

The proposed architecture is trained on the dataset after completing the pre-processing stage. In Tables 3, the confusion matrix obtained by evaluating the proposed architecture is provided. This matrix represents the overall performance of the proposed architecture during testing. It is observed that diagonal values of this matrix, representing the number of correctly predicted beats, are much higher compared to others. Though the number of incorrect predictions seems to be large for the normal class, with a large number of tested beats, these belong to a very small percentage of total normal beats. Moreover, the proposed network consists of 238,629 number of total parameters and maximum accuracy is reached in 99 epochs. Hence, this network is very lightweight that converges to the optimum performance in considerably smaller number of iterations.

TABLE 3. Confusion matrix for proposed DeepArrNet.

Actual	Predicted				
	N	*S*	*V*	*F*	*Q*
N	89955	118	272	109	135
S	11	2749	8	4	7
V	19	25	7170	7	15
F	6	2	3	789	3
Q	16	7	12	5	7999

* N: Normal, S: Supraventricular, V: Ventricular, F: Fusion, Q: Unknown Beat.

In Fig. 7, the performance of the proposed network is presented in terms of sensitivity, positive predictive value and F1 score for each class (N, S, V, F, Q). Generally, the presence of a large number of training beats in a particular class can create a bias towards predicting that class in most of the cases which may result in a higher value of sensitivity in that class.

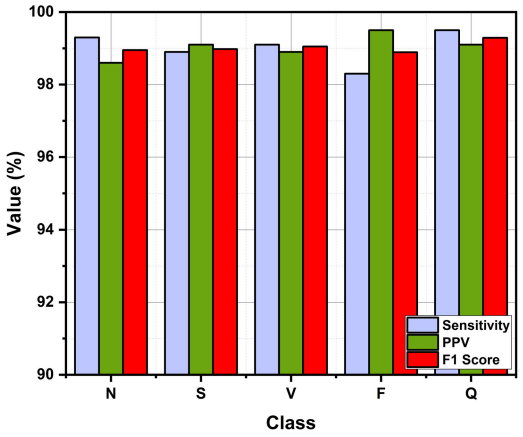


FIGURE 7. Representing per class performance of the proposed network in sensitivity, positive predictive Value, and F1 score in each class.

TABLE 4. Representing the performance of the proposed scheme in different cross validation sets of ten-fold cross validation scheme.

Cross Validation Set	Average Sensitivity(%)	Average PPV(%)	Average F1 Score(%)
1	99.15	99.13	99.14
2	99.32	99.08	99.20
3	99.24	98.94	99.09
4	99.14	99.11	99.12
5	99.36	99.05	99.21
6	99.29	98.92	99.10
7	99.31	98.99	99.15
8	99.19	99.04	99.13
9	99.24	99.12	99.18
10	99.27	99.03	99.15

However, this problem is almost overcome with the proposed scheme providing quite satisfactory sensitivity values in all classes. On the other hand, due to the tendency of biasing towards a larger sized class, the number of false-positive predictions from other smaller sized classes likely to be higher that results in lower value of positive predictive value in the larger sized class. It is noticeable that the DeepArrNet architecture provides noteworthy predictive value in all classes consistently. Moreover, the F1 score combines the results of positive predictive value with sensitivity to provide a robust way of comparison. It can be inferred that proposed scheme provides significant performance with high consistency in the F1 metric also for all five classes.

In Table 5, the performances of the proposed method using various cross-validation schemes are presented. With the decreasing number of training folds, the number of training beats on a single fold also decreases. That tends to lower the performance due to smaller training information on each set. As we have employed various data augmentation, this effect of training data variations is reduced. However, some reductions in the value of evaluation metrics are noticeable with decreasing fold. For the tenfold scheme, the performance of the proposed network in different cross-validation sets is

TABLE 5. Results on various cross validation scheme.

Cross Validation Scheme	Average Sen.(%)	Average PPV(%)	Accuracy (%)
Leave-one-out	99.29	99.13	99.45
10-fold	99.13	99.08	99.28
5-fold	98.94	98.81	99.09
2-fold	98.11	98.54	98.96

TABLE 6. Effect of proposed data augmentation and denoising methods on sensitivity.

Class	Without Proposed Augmentation	Without denoising	Proposed Method
N	98.5	98.8	99.3
S	96.4	98.5	98.9
V	98.2	98.7	99.1
F	95.9	97.8	98.3
Q	98.1	98.6	99.5

* N: Normal, S: Supraventricular, V: Ventricular, F: Fusion, Q: Unknown Beat.

presented in Table 4. The average values of each evaluation metric for all five classes are calculated in every cross-validation set. Though the train and test data on each set are changed, the evaluation metrics show a small variation over their mean value in different cross-validation sets. It represents the robustness of the proposed methods against data variations.

Due to the scarcity of data for classes with a very small number of members, the trained network is prone to struggle to extract proper features for classification. A proper augmentation method is thus necessary to increase the overall sensitivity of the network. From Table 6, significant improvement in the sensitivity of minority 'S' and 'F' beat classes are noticeable in the proposed method comparing with the one unaccompanied by the proposed realistic augmentation scheme. However, to make fair comparison, in the scheme of without proposed augmentation, the minority classes are oversampled by taking aliases of the existing beats in each training fold to reduce the class imbalance. Hence, this comparison represents the significance of the varieties of realistic augmentation techniques employed during training. Moreover, the denoising operation applied to raw data reduces the complexity of processing and offers more opportunities to generalize various classes. The effect of this denoising operation is clearly visible in Table 6 with a noticeable increase in the sensitivity.

In many cases, it is required to detect arrhythmia rather than detecting all detailed arrhythmia classes. In order to demonstrate the performance of the proposed method in such arrhythmia detection problems, all the classes of arrhythmia are considered to be the diseased class as a whole and the proposed methods are applied to identify them with the normal beats. In Table 7, the performance of the proposed network in arrhythmia detection is presented. As it becomes a binary classification problem, the proposed network is performing even better in this case. In Tab. 8, performance of the proposed

TABLE 7. Performance of the proposed scheme on arrhythmia detection employing total arrhythmia and normal class.

Metric	Value(%)
Positive Predictive Value	99.59
Sensitivity	99.71
Specificity	99.91
Accuracy	99.87

TABLE 8. Performance of the proposed scheme on PTB database [54].

Metric	Value(%)
Accuracy	99.21
Average Precision	99.03
Average Recall	99.12
Average F1 Score	99.08

architecture is presented on the secondary PTB database [54]. It is clear that the proposed architecture performs consistently on this database also.

2) COMPARISON OF THE PROPOSED APPROACH WITH OTHER EXISTING APPROACHES

A comparative analysis of various approaches with the proposed one is presented in Table 9 in terms of the evaluated metrics. Our 1D CNN based proposed architecture with the applied techniques of augmentations and data denoising provide outstanding results that outperform most other approaches. Moreover, the accuracy metric that mainly represents the total number of correct predictions as a whole, provides a significant improvement compared to others. Jun *et al.* [22] converted 1D ECG beats into 2D images and utilized a 2D CNN architecture which is relatively denser with a large number of parameters. Moreover, in their approach, raw data is directly used without denoising and with a lack of variations in applied augmentations. Übeyli [24] provided an RNN based approach with comparable results in all metrics. However, as RNN is difficult to train compared to CNN while suffering from vanishing and exploding gradient problems, proposed 1D CNN based methods provide better performance. Li and Zhou [6] and Matris *et al.* [9] used traditional handcrafted feature-based approaches using discrete wavelet transform with traditional classifiers commonly used for shallow networks, which offers unsatisfactory performance as expected. Ihsanto *et al.* [19] classified over a large number of arrhythmia classes. However, due to lack of augmentation methods with extremely smaller number of beats in many of the rare classes, average performance became lower for overfitting towards larger classes.

Despite significant performance is achieved using the proposed scheme, some other perspectives can be explored in future studies. Similar to the most other established studies in arrhythmia classification [6], [9], [15], [16], five arrhythmia classes are considered in this study and very satisfactory performance is achieved. However, the proposed scheme can

TABLE 9. Comparison of proposed scheme with existing methods on average values of evaluation metrics on MIT BIH dataset [52].

Work	Method	No. of Class	Accuracy (%)	Average Sensitivity(%)	Average PPV(%)	Average F1 Score(%)
Matris *et al.* [9]	DWT + SVM	5	93.8	91.5	87.9	89.06
Li *et al.* [6]	DWT + Random Forest	5	94.6	92.4	91.9	92.15
Derya *et al.* [24]	RNN	4	97.07	97.15	97.03	97.09
Acharya *et al.* [17]	1D CNN	5	93.5	93.35	93.47	93.41
Kiranyaz *et al.* [15]	1D CNN	5	96.4	79.2	68.8	74.1
Ihsanto *et al.* [19]	Ensemble 1D CNNs	16	99.02	76.32	86.23	80.97
Jun *et al.* [22]	2D CNN	8	97.5	97.05	97.85	97.44
Proposed DeepArrNet	**1D CNN**	**5**	**99.28**	**99.13**	**99.08**	**99.11**

be extended considering more number of arrhythmia classes as considered in [19], [22]. This may increase the class imbalance problem for much smaller number of beats in some of the rare classes that will put more emphasis on the augmentation methods. Hence, the proposed augmentation methods should be explored more to increase the diversity of operations for reducing class imbalance. Nonetheless, the DeepArrNet architecture can be easily employed with more arrhythmia classes as it can extract features from broad spectrum with varying temporal kernels that provides consistent performance in all five classes. Moreover, more number of challenging datasets can be studied using the proposed scheme. Additionally, the performance of DeepArrNet is dependent on accurate R-peak detection. Though the R-peak detection algorithm used in this study provides mostly accurate performance, performance of this scheme should be explored in more challenging cases for increasing robustness against false detection of R-peaks.

V. CONCLUSION

In this paper, a deep CNN architecture is proposed for arrhythmia detection and classification from ECG data. This architecture is based on a structural unit performing PTP (Pointwise-Temporal-Pointwise) convolution that utilizes depthwise separable convolution in the 1D domain. First, the raw ECG beat streams are denoised through a wavelet decomposition technique and then the denoised data are augmented through a variety of realistic augmentation approaches to minimize the data imbalance. Next, to detect and classify arrhythmia, a deep CNN architecture (namely DeepArrNet) is proposed that utilizes PTP blocks sequentially with different temporal windows and combines features by introducing the residual path for collecting contributions from different sequential PTP blocks. It is observed that the proposed architecture provides better generalization among various smaller numbered classes. Moreover, the proposed architecture utilizes various temporal windows in parallel while reducing the feature map using strided convolution. This offers a very lightweight architecture with great generalization capability that becomes the best fit for arrhythmia classification and provides state of the art result in all the evaluation metrics. It is expected that the proposed architecture can also be used in other applications similar to arrhythmia classification employing various other 1D bio-signals.

REFERENCES

[1] S. C. Wang, D. Meng, H. Yang, X. Wang, S. Jia, Y.-F. Wang, and P. Wang, "Pathological basis of cardiac arrhythmias: Vicious cycle of immune-metabolic dysregulation," *Cardiovascular Disorders Med.*, vol. 3, no. 1, pp. 1–7, 2018.

[2] H. Wellens, "Cardiac arrhythmias: The quest for a cureA historical perspective," *J. Amer. College Cardiol.*, vol. 44, no. 6, pp. 1155–1163, Sep. 2004.

[3] *Testing and Reporting Performance Results of Cardiac Rhythm and ST Segment Measurement Algorithms*, Association for the Advancement of Medical Instrumentation, Arlington, VA, USA, ANSI/AAMI, 1998.

[4] F. Melgani and Y. Bazi, "Classification of electrocardiogram signals with support vector machines and particle swarm optimization," *IEEE Trans. Inf. Technol. Biomed.*, vol. 12, no. 5, pp. 667–677, Sep. 2008.

[5] J. Park, K. Lee, and K. Kang, "Arrhythmia detection from heartbeat using k-nearest neighbor classifier," in *Proc. IEEE Int. Conf. Bioinf. Biomed.*, Dec. 2013, pp. 15–22.

[6] T. Li and M. Zhou, "ECG classification using wavelet packet entropy and random forests," *Entropy*, vol. 18, no. 8, p. 285, Aug. 2016, doi: 10.3390/e18080285.

[7] S.-N. Yu and Y.-H. Chen, "Electrocardiogram beat classification based on wavelet transformation and probabilistic neural network," *Pattern Recognit. Lett.*, vol. 28, no. 10, pp. 1142–1150, Jul. 2007.

[8] I. Guler and E. Ubeyli, "ECG beat classifier designed by combined neural network model," *Pattern Recognit.*, vol. 38, no. 2, pp. 199–208, Feb. 2005.

[9] 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, Sep. 2013.

[10] S. Banerjee and M. Mitra, "Application of cross wavelet transform for ECG pattern analysis and classification," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 2, pp. 326–333, Feb. 2014.

[11] T. Hoai Linh, S. Osowski, and M. Stodolski, "On-line heart beat recognition using Hermite polynomials and neuro-fuzzy network," *IEEE Trans. Instrum. Meas.*, vol. 52, no. 4, pp. 1224–1231, Aug. 2003.

[12] R. J. Martis, U. R. Acharya, C. M. Lim, and J. S. Suri, "Characterization of ECG beats from cardiac arrhythmia using discrete cosine transform in PCA framework," *Knowl.-Based Syst.*, vol. 45, pp. 76–82, Jun. 2013.

[13] S. Yu and K. Chou, "Integration of independent component analysis and neural networks for ECG beat classification," *Expert Syst. Appl.*, vol. 34, no. 4, pp. 2841–2846, May 2008.

[14] A. Y. Hannun, P. Rajpurkar, M. Haghpanahi, G. H. Tison, C. Bourn, M. P. Turakhia, and A. Y. Ng, "Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network," *Nature Med.*, vol. 25, no. 1, pp. 65–69, Jan. 2019.

[15] S. Kiranyaz, T. Ince, and M. Gabbouj, "Real-time patient-specific ECG classification by 1-D convolutional neural networks," *IEEE Trans. Biomed. Eng.*, vol. 63, no. 3, pp. 664–675, Mar. 2016.

[16] N. Strodthoff and C. Strodthoff, "Detecting and interpreting myocardial infarction using fully convolutional neural networks," 2018, *arXiv:1806.07385*. [Online]. Available: http://arxiv.org/abs/1806.07385

[17] U. R. Acharya, S. L. Oh, Y. Hagiwara, J. H. Tan, M. Adam, A. Gertych, and R. S. Tan, "A deep convolutional neural network model to classify heartbeats," *Comput. Biol. Med.*, vol. 89, pp. 389–396, Oct. 2017.

[18] T. Mahmud, A. R. Hossain, and S. A. Fattah, "ECGDeepNET: A deep learning approach for classifying ECG beats," in *Proc. 7th Int. Conf. Robot Intell. Technol. Appl. (RiTA)*, Nov. 2019, pp. 32–37.

[19] E. Ihsanto, K. Ramli, D. Sudiana, and T. S. Gunawan, "An efficient algorithm for cardiac arrhythmia classification using ensemble of depthwise separable convolutional neural networks," *Appl. Sci.*, vol. 10, no. 2, p. 483, Jan. 2020, doi: 10.3390/app10020483.

[20] M. Salem, S. Taheri, and J.-S. Yuan, "ECG arrhythmia classification using transfer learning from 2- dimensional deep CNN features," in *Proc. IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Oct. 2018, pp. 1–4.

[21] M. Kachuee, S. Fazeli, and M. Sarrafzadeh, "ECG heartbeat classification: A deep transferable representation," in *Proc. IEEE Int. Conf. Healthcare Informat. (ICHI)*, Jun. 2018, pp. 443–444.

[22] T. J. Jun, H. M. Nguyen, D. Kang, D. Kim, D. Kim, and Y.-H. Kim, "ECG arrhythmia classification using a 2-D convolutional neural network," 2018, *arXiv:1804.06812*. [Online]. Available: http://arxiv.org/abs/1804.06812

[23] S. Mousavi and F. Afghah, "Inter- and Intra- patient ECG heartbeat classification for arrhythmia detection: A sequence to sequence deep learning approach," in *Proc. ICASSP IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 1308–1312.

[24] E. D. Übeyli, "Combining recurrent neural networks with eigenvector methods for classification of ECG beats," *Digit. Signal Process.*, vol. 19, no. 2, pp. 320–329, Mar. 2009.

[25] Ö. Yildirim, "A novel wavelet sequence based on deep bidirectional LSTM network model for ECG signal classification," *Comput. Biol. Med.*, vol. 96, pp. 189–202, May 2018.

[26] E. J. D. S. Luz, W. R. Schwartz, G. Cámara-Chávez, and D. Menotti, "ECG-based heartbeat classification for arrhythmia detection: A survey," *Comput. Methods Programs Biomed.*, vol. 127, pp. 144–164, Apr. 2016, doi: 10.1016/j.cmpb.2015.12.008.

[27] 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.

[28] B. M. M. A. Tinati, "ECG baseline wander elimination using wavelet packets," *World Academy Sci., Eng. Technol.*, vol. 3, no. 2005, pp. 14–16, 2005.

[29] R. Sameni, M. B. Shamsollahi, C. Jutten, and G. D. Clifford, "A nonlinear Bayesian filtering framework for ECG denoising," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 12, pp. 2172–2185, Dec. 2007.

[30] P. Karthikeyan, M. Murugappan, and S. Yaacob, "ECG signal denoising using wavelet thresholding techniques in human stress assessment," *Int. J. Electr. Eng. Informat.*, vol. 4, no. 2, pp. 306–319, Jun. 2012.

[31] M. Blanco-Velasco, B. Weng, and K. E. Barner, "ECG signal denoising and baseline wander correction based on the empirical mode decomposition," *Comput. Biol. Med.*, vol. 38, no. 1, pp. 1–13, Jan. 2008.

[32] Q. Qin, J. Li, Y. Yue, and C. Liu, "An adaptive and time-efficient ECG R-Peak detection algorithm," *J. Healthcare Eng.*, vol. 2017, Sep. 2017, Art. no. 5980541, doi: 10.1155/2017/5980541.

[33] S. A. Chouakri, F. Bereksi-Reguig, and A. Taleb-Ahmed, "QRS complex detection based on multi wavelet packet decomposition," *Appl. Math. Comput.*, vol. 217, no. 23, pp. 9508–9525, Aug. 2011.

[34] P. Jafari Moghadam Fard, M. H. Moradi, and M. R. Tajvidi, "A novel approach in r peak detection using hybrid complex wavelet (HCW)," *Int. J. Cardiol.*, vol. 124, no. 2, pp. 250–253, Feb. 2008.

[35] D. Sadhukhan and M. Mitra, "R-peak detection algorithm for ECG using double difference and RR interval processing," *Procedia Technol.*, vol. 4, pp. 837–877, Jan. 2012, doi: 10.1016/j.protcy.2012.05.143.

[36] M. S. Manikandan and K. P. Soman, "A novel method for detecting R-peaks in electrocardiogram (ECG) signal," *Biomed. Signal Process. Control*, vol. 7, no. 2, pp. 118–128, Mar. 2012.

[37] M. Elgendi, "Fast QRS detection with an optimized knowledge-based method: Evaluation on 11 standard ECG databases," *PLoS ONE*, vol. 8, no. 9, Sep. 2013, Art. no. e73557, doi: 10.1371/journal.pone.0073557.

[38] M. Elgendi, A. Mohamed, and R. Ward, "Efficient ECG compression and QRS detection for E-Health applications," *Sci. Rep.*, vol. 7, no. 1, pp. 1–16, Mar. 2017.

[39] T. Jayalakshmi and A. Santhakumaran, "Statistical normalization and back propagation for classification," *Int. J. Comput. Theory Eng.*, vol. 3, no. 1, pp. 1793–8201, 2011.

[40] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: http://arxiv.org/abs/1704.04861

[41] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1251–1258.

[42] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2019, pp. 801–818.

[43] G. Li, I. Yun, J. Kim, and J. Kim, "DABNet: Depth-wise asymmetric bottleneck for real-time semantic segmentation," 2019, *arXiv:1907.11357*. [Online]. Available: http://arxiv.org/abs/1907.11357

[44] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient CNN architecture design," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 116–131.

[45] L. Kaiser, A. N. Gomez, and F. Chollet, "Depthwise separable convolutions for neural machine translation," 2017, *arXiv:1706.03059*. [Online]. Available: http://arxiv.org/abs/1706.03059

[46] S. Kriman, S. Beliaev, B. Ginsburg, J. Huang, O. Kuchaiev, V. Lavrukhin, R. Leary, J. Li, and Y. Zhang, "QuartzNet: Deep automatic speech recognition with 1D time-channel separable convolutions," 2019, *arXiv:1910.10261*. [Online]. Available: http://arxiv.org/abs/1910.10261

[47] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, Mar. 2010, pp. 156–249.

[48] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.

[49] S. Krishna Kumar, "On weight initialization in deep neural networks," 2017, *arXiv:1704.08863*. [Online]. Available: http://arxiv.org/abs/1704.08863

[50] S. Shalev-Shwartz and T. Zhang, "Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization," in *Proc. Int. Conf. Mach. Learn.*, Jan. 2014, pp. 64–72.

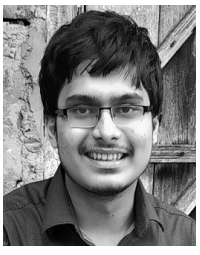
[51] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[52] 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, May 2001. [Online]. Available: http://physionet.org/physiobank/database/mitdb/

[53] L. Sörnmo and P. Laguna, "Electrocardiogram (ECG) signal processing," in *Wiley Encyclopedia of Biomedical Engineering*. Hoboken, NJ, USA: Wiley, 2006, doi: 10.1002/9780471740360.ebs1482.

[54] R. Bousseljot, D. Kreiseler, and A. Schnabel, "Use of the PTB's ECG signal database CARDIODAT via the Internet," (in german), *Biomedizinische Technik/Biomed. Eng.*, vol. 40, no. s1, pp. 317–318, 1995.

[55] M. Hossin and M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations," *Int. J. Data Mining Knowl. Manage. Process*, vol. 5, no. 2, pp. 1–11, 2015.



TANVIR MAHMUD (Student Member, IEEE) received the B.Sc. degree from EEE Department, Bangladesh University of Engineering and Technology, where he is currently pursuing the master's degree. He is currently serving as a Lecturer with EEE Department, Bangladesh University of Engineering and Technology. His research interests include VLSI circuit design, approximate computing, biomedical signal processing, image processing, and machine learning.

SHAIKH ANOWARUL FATTAH (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees from the Bangladesh University of Engineering and Technology (BUET), Bangladesh, and the Ph.D. degree in ECE from Concordia University, Canada. He held a visiting postdoctoral position and later a visiting Research Associate with Princeton University, Princeton, NJ, USA. He has been serving as a Professor with the Department of EEE, BUET. He has published over 200 international journal articles and conference papers with some best paper awards. His major research interests include biomedical engineering and signal processing. He is a Fellow of IEB. He is regularly delivering keynote/invited/visiting talks in many countries. He received several prestigious awards, such as the Concordia University's Distinguished Doctoral Dissertation Prize in ENS, the Dr. Rashid Gold Medal (M.Sc., BUET), the NSERC Postdoctoral Fellowship, the URSI Canadian Young Scientist Award 2007, and the BAS-TWAS Young Scientists Prize 2014. He is the General Chair of IEEE R10 HTC2017, ICAICT 2020, the TPC Chair of IEEE TENSYMP 2020, IEEE WIECON-ECE 2016, 2017, MediTec 2016, IEEE ICIVPR 2017, and ICAEE 2017. He is a Committee Member of IEEE PES (LRPC), IEEE SSSIT (SDHTC), IEEE HAC (2018–2020), and R10. He was the Chair of the IEEE Bangladesh Section (2015–2016). He was the Chair of the IEEE EMBS Bangladesh Chapter (2017–2019). He is the Founder Chair of the IEEE RAS and SSIT Bangladesh Chapters. He was an Editor of the *Journal of Electrical Engineering of IEB*. He is an Editor of the IEEE PES eNews, an Editorial Board Member of IEEE Access, and an Associate Editor of CSSP (Springer).

MOHAMMAD SAQUIB (Senior Member, IEEE) received the B.Sc. degree from the Bangladesh University of Engineering and Technology, Bangladesh, in 1991, and the M.S. and Ph.D. degrees from Rutgers University, New Brunswick, NJ, USA, in 1995 and 1998, respectively, all in electrical engineering. He worked as a member of the technical staff with the Massachusetts Institute of Technology Lincoln Laboratory, and as an Assistant Professor with Louisiana State University, Baton Rouge, LA, USA. He is a Professor with Electrical Engineering Department, The University of Texas at Dallas, Richardson, TX, USA. His current research interests include the various aspects of wireless data transmission, radio resource management, and signal processing techniques for low-cost radar applications.