心律不齐

qrs检测是处理ECG信号的基础

Robust Arrhythmia Classification Based on QRS Detection and a Compact 1D-CNN for Wearable ECG Devices

可穿戴的

Nabil Sabor, *Member, IEEE*, Garas Gendy, *Student Member, IEEE*, Hazem Mohammed, *Student Member, IEEE,* Guoxing Wang, *Senior Member, IEEE*, and Yong Lian, *Fellow, IEEE*

***Abstract*— Embedded arrhythmia classification is the first step towards heart diseases prevention in wearable applications. In this paper, a robust arrhythmia classification algorithm, NEO-CCNN, for wearables that can be implemented on a simple microcontroller is proposed. The NEO-CCNN algorithm not only detects QRS complex but also accurately locates R-peak with the help of the proposed adaptive time-dependent thresholding technique, improving the accuracy and sensitivity in arrhythmia classification. An optimized compact 1D-CNN network (CCNN) with 9,701 parameters is used for classification. A QRS complex augmentation method is introduced in the training process to cater for R-peak location error (RLE). A nested k_1k_2-fold cross-validation method is utilized to evaluate the robustness of the proposed algorithm. Simulation results show that the proposed algorithm has the ability to detect more than 99.79% of R peaks with an RLE of 7.94 ms for the MIT-BIH database. Implemented on the STM32F407 microcontroller, NEO-CNN attains a classification accuracy of 97.83% and sensitivity of 96.46% using only 8s window size.**

***Index Terms*—QRS Complexes, Nonlinear Energy Operator, Cardiac Arrhythmia, Compact CNN, and Wearable Devices.**

单片机 峰值性能 增强 适应 嵌套的 鲁棒性 仿真

无论 应该是一个倒装句

I. Introduction

心血管

Cardiovascular diseases (CVDs) are the leading cause of death of 17.9 million people each year as reported by WHO in 2016 [1]. Approximately 85% of deaths are due to complications of cardiac arrhythmia such as heart attack and stroke. It is not sufficient just to detect the existence of arrhythmia, the type and frequency also matter [2]. The use of a wearable ECG (Electrocardiography) sensor and smartphone to monitor heart conditions is an effective way for the early detection of abnormal arrhythmia [3]. However, the raw ECG data transmission from the sensor to the mobile phone consumes significant power, limiting the battery life of the sensor [4]. This issue can be addressed if the sensor is built with an embedded arrhythmia classifier.

心律失常 足够的 中风

It is challenging for the development of an embedded arrhythmia classifier with reasonable accuracy due to the variations of ECG morphology caused by age, gender, physical conditions and activities, and the environment [5, 6]. The issue is further complicated by the sporadic nature of abnormal heart rhythms. Recent developments have shown good classification accuracy of 98% evaluated on the MIT-BIH Arrhythmia database (MITD-AR) [7] at the cost of using complicated neural networks [8], which are not suitable for embedded applications. Moreover, the evaluation is based on selected records from the MITD-AR as in [9-11]. In other words, the accuracy could deteriorate if evaluated on all records of MITD-AR [12].

异常心律的发散性使得问题更加复杂 发散性？？

The complexity of the classification algorithms depends on the preprocessing techniques and neural network models, i.e. input data size and shape, number of layers, kernel size and number, number of classes, etc. The blind segmentation-based algorithms depend less on the preprocessing with the exception that the consecutive R peaks within each segment must come from the same beat's type [11, 13]. This assumption is impractical in many real applications. Therefore, QRS detection is still an important step for a practical arrhythmia classifier and other applications like epilepsy detection [14, 15].

预处理 也就是说 盲分割？ 除了每个片段中连续的R峰值必须来自相同的节拍类型（怪怪的不理解） 癫痫

Most of the existing QRS detection algorithms adopt the thresholding technique [3, 16-20] in the decision-making step due to its simplicity compared to the deep networks. However, the selection of a proper threshold for a better detection rate regardless of the QRS morphology is a very challenging task. Moreover, few studies were considering the R-peak location error (RLE), i.e. the distance between the annotated and the detected R-peak locations, in their performance metric. In fact, the effect of RLE is not studied on the performance of the classification algorithm based on our knowledge.

通过 在决策步骤使用阈值技术，因为它相比其他方法更加简单 标注，注释

Despite the importance of detection and classification of arrhythmia, still there are no available wearable devices for this issue on market. Consequently, physicians still need to admit the patient to the hospital under the observation of more advanced machinery to detect and classify arrhythmia. If the wearable device was able to detect and classify arrhythmia, this would provide huge relief to arrhythmia patients [2].

In this paper, we present a robust arrhythmia classification algorithm, NEO-CCNN, for wearables with the implementation on a simple microcontroller. It consists of a nonlinear energy operator (NEO) based thresholding for R-peak detection, and a compact 1D conventional neural network (CCNN) for classification. The impact of RLE on the performance of classification is studied, which leads to the proposed QRS complex augmentation method for training. A nested k_1k_2-fold

未经处理的

This work is supported in part by the National Nature Science Foundation of China under Grant No. 61874171. Guoxing Wang is the corresponding author.

Nabil Sabor is with the Electrical Engineering Department, Faculty of Engineering, Assiut University, Assiut, 71516 Egypt (e-mail: nabil_sabor@aun.edu.eg,).

Garas Gendy, Hazem Mahommed, Guoxing Wang and Yong Lian are with the Department of Micro-Nano Electronics, Shanghai Jiao Tong University, Shanghai 200240, China, and also with the MoE Key Laboratory of Artificial Intelligence, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: garasgaras@yahoo.com, h.moh.88@aun.edu.eg, guoxing@sjtu.edu.cn, eleliany@sjtu.edu.cn).

TABLE I
TYPES OF QRS BEATS IN THE MIT-BIH ARRHYTHMIA AND QT DATABASES

AAMI	Beat Type	MITD-AR	QTDB	AAMI	Beat Type	MITD-AR	QTDB
NB	**N**: Normal beat	75,052	80,710	SB	**A:** Atrial premature beat	2,546	758
	L: Left bundle branch block beat	8,075	0		**a:** Aberrated atrial premature beat	150	2
	R: Right bundle branch block beat	7,259	679		**J:** Nodal (junctional) premature beat	83	3
	e: Atrial escape beat	16	9		**S**: Supraventricular premature or ectopic	2	731
	j: Nodal (junctional) escape beat	229	1	QB	**Q:** Unclassifiable beat	33	3
VB	**V:** Premature ventricular contraction	7,130	1,698		**/ :** Paced beat	7,028	1,847
	E: Ventricular escape beat	106	0		**f:** Fusion of paced and normal beat	982	306
FB	**F:** Fusion of ventricular and normal	803	251				
	Total beats					**109,494**	**86,995**

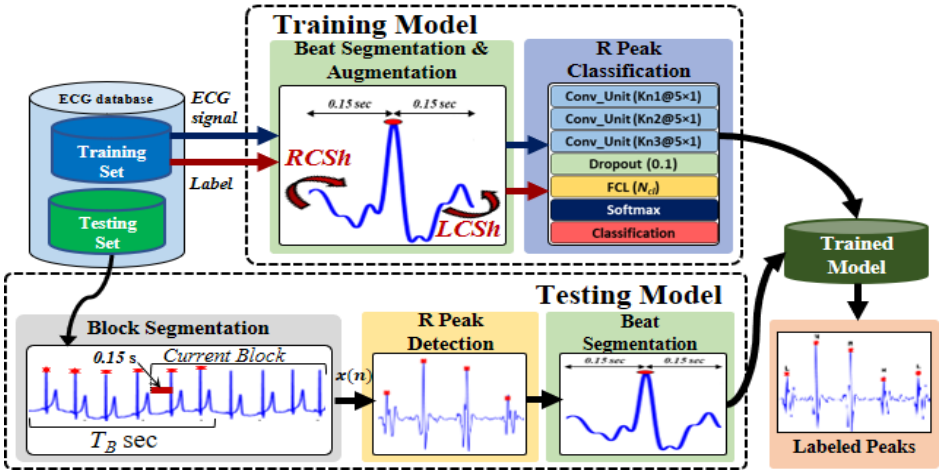


Fig. 1. Framework of the proposed arrhythmia classification (NEO-CCNN) algorithm. The right (RCSh) and left (LCSh) circular shift are used for augmenting data before training.

cross-validation method is adopted to measure the robustness of the NEO-CCNN algorithm using MITD-AR and QT databases [21]. Our implementation of the NEO-CCNN algorithm takes 46.61 s for detecting and classifying R beats of a 30 min record using the STM32F407 microcontroller. This makes our algorithm suitable for wearable ECG devices for the early detection and classification of abnormal arrhythmia.

The paper is organized as follows. Section II describes the databases used in this study. The details of the proposed approach are presented in Section III. The computational complexity as well as the hardware implementation is given in Section IV. Section V presents the evaluation results. Conclusion remarks are drawn in Section VI.

II. ECG DATABASES

Two databases, the MIT-BIH Arrhythmia database (MITD-AR) and QT database, are used in this study.

A. MIT-BIH Arrhythmia (MITD-AR): is one of the most commonly used databases for QRS detection and arrhythmia classification which contains 48 records of 47 patients (25 M/ 22 F, with the age of 23–89 years) with a total of 24 hours. Each record is 30-min long and sampled at 360 Hz. In record 207, 2-min of the ventricular flutter segments are excluded because they are not enough for training and testing the model. The rest of record 207 and all other records, containing 109,494 beats annotated by two cardiologists, are considered.

B. QT database (QTDB): consists of 105 ECG records and only 82 records are annotated. The total duration of the annotated records is 20.5 hours with 86,995 annotated beats. Each record has a duration of 15-min and is sampled at 250 Hz.

For consistency consideration, the samples in the MITD-AR are resampled to 250Hz (f_s = 250 Hz). According to the annotation of cardiologists, the distribution of beats for both datasets is listed in Table I, where the beat types are classified according to the AAMI [22], i.e. Normal non-ectopic beats (NB), Supraventricular ectopic beats (SB), Ventricular ectopic beats (VB), Fusion beats (FB), and Unknown beats (QB).

III. METHODOLOGY

Fig. 1 shows the framework of the proposed NEO-CCNN algorithm. It can be roughly divided into two main stages: R peak detection and beat classification. In the R peak detection stage, a time-dependent threshold-based algorithm is proposed by incorporating the NEO with the first derivative for preprocessing. The detected R peaks are segmented into 0.3 s segments centered around R peaks, which is sufficient to cover a complete QRS complex, as the input to the proposed CCNN network for classifying into different beat types. The training is performed using the annotated QRS peaks, while the detected QRS peaks are used in the evaluation to measure the robustness of the proposed algorithm.

For real-time processing, each ECG record is divided into blocks of T_B second as input $x(n)$. The value of T_B is set to 15s which will be explained later. To avoid the discontinuity of non-overlapped segmentation, an overlapped segmentation method is proposed to ensure that the last two detected QRS peaks in the previous block will be included in the current block as illustrated by the Block Segmentation in Fig. 1.

A. Stage 1: R Peak Detection

The proposed QRS detection algorithm consists of three steps

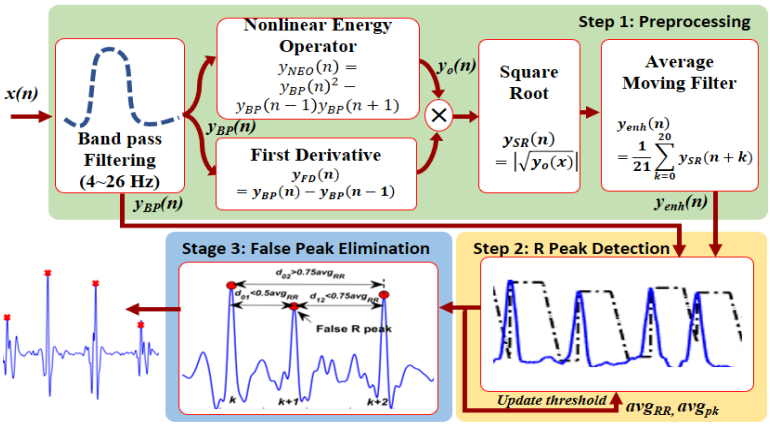


Fig. 2. Proposed QRS detection algorithm. The algorithm consists of three steps: preprocessing, decision-making, and False peak elimination step. In step 3, there are three criteria, but the picture of Criteria 2 is put as an example.

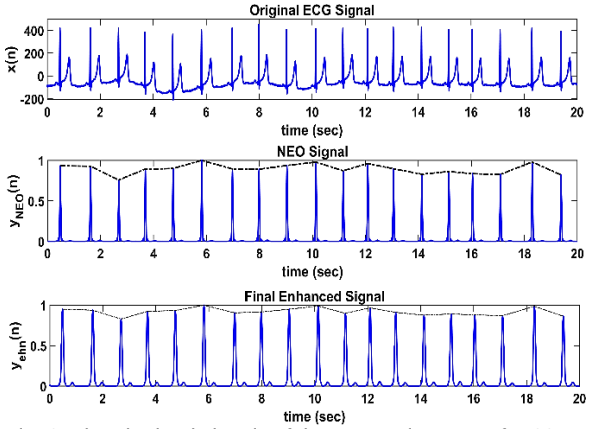


Fig. 3. The obtained signals of the processing stage for 20-sec ECG signal of record 113 with pointed T waves in MITD-AR dataset.

as shown in Fig. 2, namely the preprocessing, the decision-making, and the fine-tuning step.

1) Preprocessing Step

The preprocessing step is used to remove unwanted noises and to differentiate the QRS complexes from other peaks such as pointed P and T waves. Since most QRS energy lies in the frequency range of 5–22 Hz, a bandpass FIR filter $h(n)$ of 4 ~ 26 Hz and order of M=51 is designed to retain QRS. The NEO technique [23] and the first derivative are respectively applied to the filtered signal $y_{BP}(n)$ to differentiate R peaks from other P and T peaks, as in (1) and (2). The enhanced signal $y_{enh}(n)$ is obtained by smoothing the square root of $|y_O(n)|$, where $y_O(n) = y_{NEO}(n) \times y_{FD}(n)$ as illustrated in (3) and (4),

$$y_{NEO}(n) = y_{BP}(n)^2 - y_{BP}(n-1)y_{BP}(n+1), \quad (1)$$

$$y_{FD}(n) = y_{BP}(n) - y_{BP}(n-1). \quad (2)$$

$$y_{sqrt}(n) = \sqrt{|y_O(n)|}, \quad (3)$$

$$y_{enh}(n) = \frac{1}{21}\sum_{k=0}^{20} y_{sqrt}(n+k) = y_{enh}(n-1) + \frac{1}{21}(y_{sqrt}(n+20) - y_{sqrt}(n-1)) \quad (4)$$

As QRS duration varies from 70 to 120 ms [24], the length of the moving average filter is selected as 84 ms, i.e. 21 samples at f_s= 250 Hz. There are several benefits for utilizing the enhanced signal compared to using NEO alone as illustrated by record 113 in Fig. 3. It can be seen that the NEO technique enlarges the low-level R peaks and attenuates the pointed T-waves but shows a distinct variation among R peaks with a standard deviation (SD) of 0.062. After the enhancement, the SD drops by 25.8%, e.g. from 0.062 to 0.046, which equalizes the R peaks for better thresholding.

2) Decision-Making Step

The accuracy of QRS detection depends largely on the threshold strategy. Fixed threshold normally leads to poor performance due to the variation of ECG morphology caused by physical conditions and activities. Moreover, it is difficult to differentiate the pointed P and T waves from small R peaks. We address the issue by proposing a time-dependent threshold method, as illustrated in **Algorithm 1,** to dynamically set the threshold value $Th(n)$ based on the RR interval and amplitude of detected R peaks. With the proposed threshold, the R peaks are determined by first identifying the potential R peak locations from the enhanced signal, $y_{enh}(n)$, then searching for the maxima in the vicinity of the potential R peaks in $y_{BP}(n)$.

The potential R peaks are located by scanning a block of $y_{enh}(n)$ of T_B second to capture the first peak that is greater than Th_0; where Th_0 is the initial threshold value determined by the mean and standard deviation of $y_{enh}(n)$. This potential R peak is marked as "i" while $Th(n)$ is updated to $y_{enh}(i)$, i.e. $Th(n) = y_{enh}(i)$. To avoid the threshold being dominated by either a too large or too small peak of $y_{enh}(n)$, a block of $y_{enh}(n)$ is divided into four equal subsections with a size of $0.25T_B$ ($y_{enh}^s(n), s = 1, \ldots, 4$) and the average of the maximum values of these subsections ($avg_{enh-max}$) is considered for setting the value of $Th(n)$ as follows:

$$Th(n) = h_{max_i} = \begin{cases} y_{enh}(i), & if\ y_{enh}(i) \le avg_{enh-max} \\ avg_{enh-max}, & Otherwise \end{cases}$$

, where $avg_{enh-max} = \frac{1}{4}\sum_{s=1}^{4} max(y_{enh}^s(n))$ (5)

Fixing $Th(n)$ at h_{max_i} can wrongly detect the pointed T peaks and avoid the detection of the low-level potential R peaks. To avoid this problem, the value of $Th(n)$ will only keep its value h_{max_i} for M_0 samples, afterward its value linearly decreases to a reference value th_{ref_i} within M_1 samples, as shown in Fig.4. The values of M_0 and M_1 depend on the RR intervals of the detected peaks as will be discussed later in this section. The value of th_{ref_i} is calculated depending on the average amplitude (avg_{enh-Pk}) of detected potential R peaks, and the current value of $y_{enh}(n)$ as follows:

$$th_{ref_i} = \eta\left(avg_{enh-Pk} + \alpha(y_{enh}(n) - avg_{enh-Pk})\right), \quad (6)$$

where α and η are the weight factors within the range of [0 1], and they are determined empirically as 0.5 and 0.12, respectively, based on the MITD-AR dataset. The tracing of the threshold value, shown in Fig. 4, that is repeated after capturing each potential peak at index "i" can be described by the following equation:

$$Th(n) = \begin{cases} h_{max_i}, & i \le n \le i + M_0 \\ h_{max_i} - \frac{h_{max_i} - th_{ref_i}}{M_1}(n - i - M_0), & i + M_0 < n \le i + M_0 + M_1 \\ th_{ref_i}, & n > i + M_0 + M_1 \end{cases} \quad (7)$$

If another peak is found within the duration of M_0 samples, the highest peak will be recorded as the potential R peak and all others will be discarded. The detected R peaks from $y_{enh}(n)$ are used as indicators to find the actual R peaks in $y_{BP}(n)$.

Algorithm 1: *Proposed R Peaks Detection Algorithm*

Inputs: $y_{BP}(n)$, $y_{enh}(n)$
Outputs: *R peaks of ECG* (P_R)
Q_{nn}=200 samples, η=0.12; α=0.5;
M_0=0.33× Q_{nn}, M_1=0.17× Q_{nn}
Th_i=0.5×(mean($y_{enh}(n)$) + std($y_{enh}(n)$))
for *each data block (x(n))*
  i=1;
  Divide $y_{enh}(n)$ *into four subsections and calculate* $avg_{enh-max}$ *using eq. 5*
  While i <=length($y_{enh}(n)$)
    if $y_{enh}(i)$> $y_{enh}(i+1)$ & $y_{enh}(i)$> $y_{enh}(i-1)$ & $y_{enh}(i)$> Th_i
      if i – *loc_lastPeak* < M_0
        select the largest peak and delete the other one
      else, *add i to locations of the candidate peaks*
      end if
      find location of the true R peak (loc_Rpeak$_i$) of ECG signal within ±15 samples (i.e. y_{BP}(i − 15: i + 15))
      Add location of the detected peak of ECG (loc_Rpeak$_i$) into P_R *vector*
      Calculate $avg_{enh-Pk} = mean(y_{enh}(P_R))$
      Update threshold as (eq.(5)) $h_{max_i} = min\ (y_{enh}(i),\ avg_{enh-max})$
      Calculate $th_{ref_i} = \eta\left(avg_{ehg-Pk} + \alpha(y_{enh}(i) - avg_{enh-Pk})\right), i > 1$
      count=1;
    else
      if $i - P_R(end) < M_0$
        Th_i= $hmax_i$;
      elseif $i - P_R(end) >= M_0$ & $i - P_R(end) < M_0 + M_1$
        $Th_i = hmax_i - \frac{hmax_i - th_{ref_i}}{M_1} \times count$
        count=count+1
      else, $Th_i = th_{ref_i}$
      end if
    end if
    i=i+1;
  end while
  Calculate avg_{RR} *of detected peaks of* $y_{BP}(P_R)$
  $RR_q = min(avg_{RR}, Qnn)$, M_0=0.33×RRq, M_1=0.17×RRq
end for

This is done by searching for the maximum value of $y_{BP}(n)$ within ±15 samples (~120 ms is the maximum duration of the normal QRS complex [24] at f_s=250 Hz) around the detected potential peak *of* $y_{enh}(n)$. To handle the changes in QRS morphologies and avoid detecting the pointed P and T-waves as R peaks, the value of M_0 and M_1 are set for each data block depending on the average of RR intervals (avg_{RR}) of the detected R peaks in the previous data blocks. As the normal QT interval is 0.45 s that represents ~ 33% of normal RR interval [24], the value of M_0 is set by $0.33 \times avg_{RR}$ to avoid detecting T wave. Also, the value of M_1 is set by $0.17 \times avg_{RR}$ to keep the sum of M_0 and M_1 within $0.5 \times avg_{RR}$ to ensure that the T wave is ended. The upper limit of M_0 and M_1 is set by the normal RR interval (Q_{nn}) as illustrated by (8), *i.e.* Q_{nn}=*200* samples at f_s=250 Hz and heart rate of 75 bpm. The initial value of avg_{RR} is set by Q_{nn} for the first data block, then the values of M_0 and M_1 are updated after each block as follows:

$$M_0 = 0.33 \times min(avg_{RR}, Q_{nn}); \quad M_1 = 0.17 \times min(avg_{RR}, Q_{nn}); \tag{8}$$

Fig. 4 shows the detected candidate R peaks of record 104 in the MITD-AR using Algorithm 1 and also shows the values of the time-dependent threshold against time. It is noticed that the threshold algorithm adapts its value depending on the peaks and RR intervals of $y_{enh}(n)$ to detect the candidate R peaks.

3) Elimination of False R Peaks Step (Fine-tuning Step)

剔除 消除

After detecting the R peaks of $x(n)$, a post-processing algorithm is applied to remove the false detected peaks. The existing post-processing algorithms such as the search-back algorithm [16] depending on the last eight RR intervals to detect

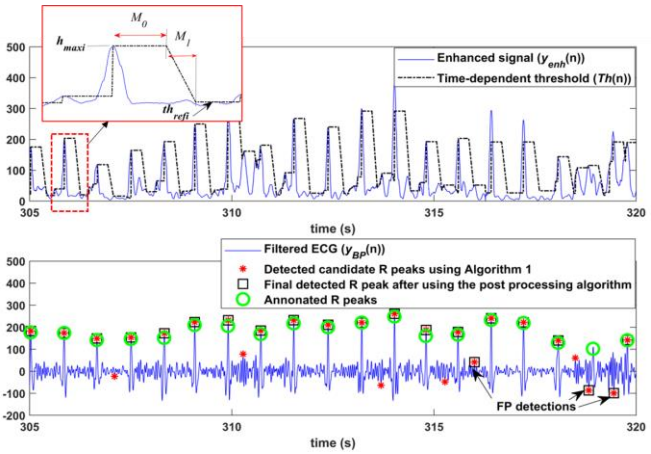


Fig. 4. R peaks detection of 15-sec ECG signal of record 104 in the MITD-AR database using Algorithm 1 and post-processing algorithm.

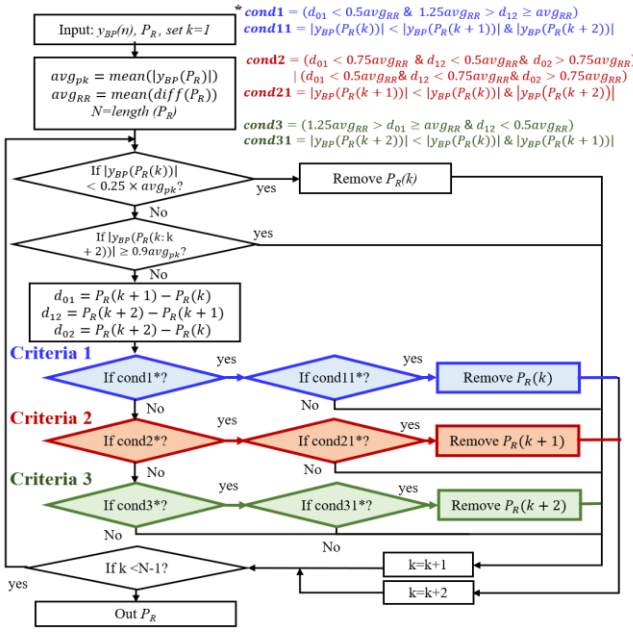
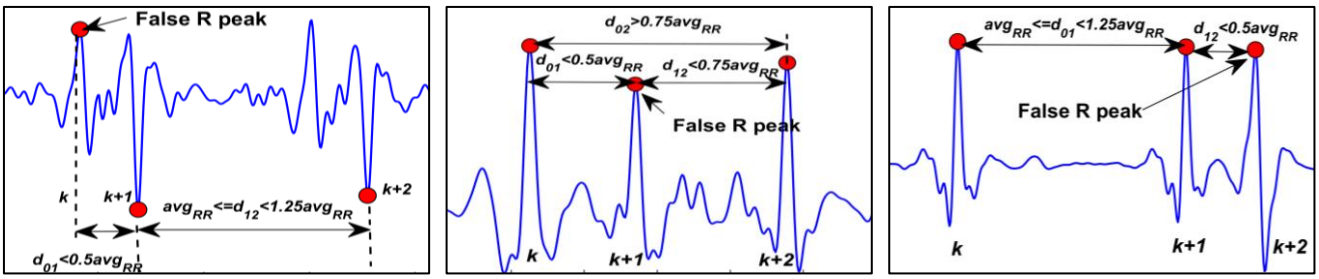


Fig. 5. Flowchart of the post-processing procedure for eliminating false R peaks. Different criteria of the procedure are illustrated by different colors.

the missed R peaks. Moreover, it considers only two consecutive peaks for deciding the false and the missed peaks. This makes the search-back algorithm not efficient for eliminating false R peaks. We propose an algorithm, as shown in Fig.5, which takes into consideration of three consecutive peaks for eliminating false peaks based on the average amplitude (avg_{Pk}) and the average RR interval (avg_{RR}) of all the detected R peaks of previous data blocks. The algorithm follows a set of criteria:

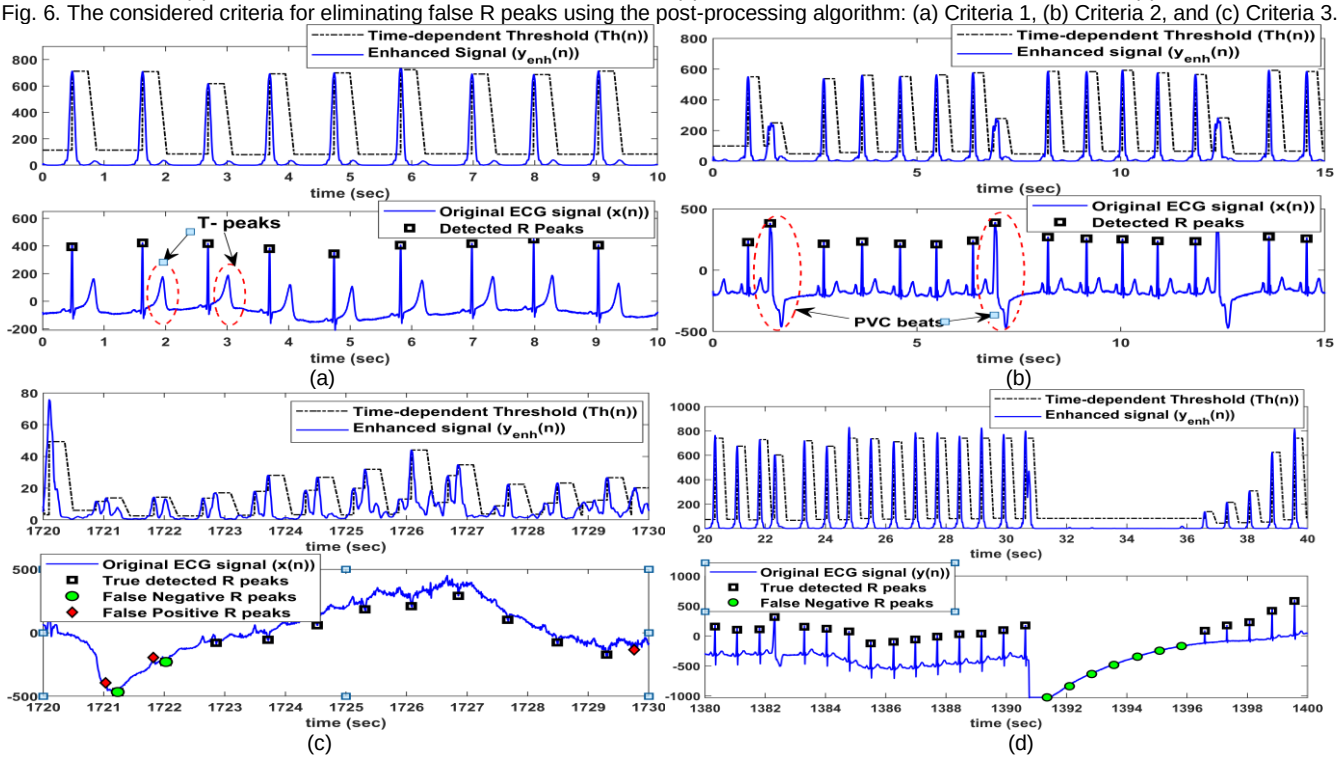
Criteria 1: the k^{th} R peak (P_k) is considered a false peak if:
a) $d_{01} < 0.5 \times avg_{RR}$ & $1.25 \times avg_{RR} > d_{12} \geq avg_{RR}$.
b) $|y_{BP}(P_R(k))| < |y_{BP}(P_R(k+1))|$ & $|y_{BP}(P_R(k+2))|$.

Criteria 2: the $(k+1)^{th}$ R peak is considered a false peak if:
a) $d_{01} < 0.75 \times avg_{RR}$ & $d_{12} < 0.5 \times avg_{RR}$; **OR** $d_{01} < 0.5 \times avg_{RR}$ & $d_{12} < 0.75 \times avg_{RR}$. Besides $d_{02} > 0.75 \times avg_{RR}$.
b) $|y_{BP}(P_R(k+1))| < |y_{BP}(P_R(k))|$ & $|y_{BP}(P_R(k+2))|$.



(a) (b) (c)

Fig. 6. The considered criteria for eliminating false R peaks using the post-processing algorithm: (a) Criteria 1, (b) Criteria 2, and (c) Criteria 3.



(a) (b) (c) (d)

Fig. 7. Different examples show the ability of the proposed detection algorithm for detecting R peaks: (a) Record 113 with pointed T-peaks, (b) Record 119 with PVC beats, (c) Record 108 with muscular noise, and (d) Record 116 with very low-level R peaks.

Criteria 3: the $(k+2)^{th}$ R peak is considered a false peak if:

a) $1.25 \times avg_{RR} > d_{01} \geq avg_{RR}$ & $d_{12} < 0.5 \times avg_{RR}$.
b) $|y_{BP}(P_R(k+2))| < |y_{BP}(P_R(k))|$ & $|y_{BP}(P_R(k+1))|$.

where, d_{01}, d_{02}, and d_{12} are the RR intervals between the k^{th} and $(k+1)^{th}$ peaks, the k^{th} and $(k+2)^{th}$ peaks, and the $(k+1)^{th}$ and $(k+2)^{th}$ peaks, respectively. The results are shown in Fig.6.

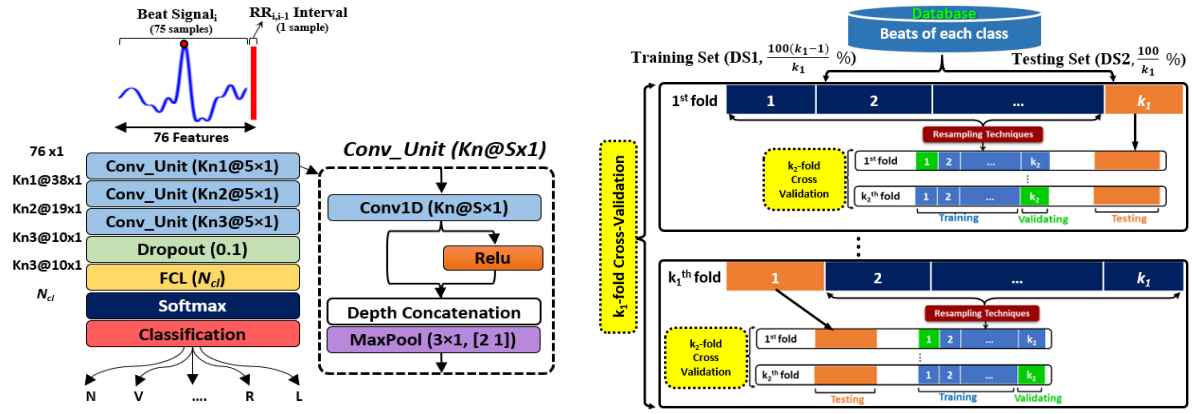
In order to simplify the processing, the detected peak is accepted as a true R peak if its amplitude is greater than $0.9\times avg_{Pk}$, and is eliminated if its amplitude is less than $0.25\times avg_{Pk}$. Otherwise, the post-processing algorithm is applied to check its status. Fig. 4 shows the effect of using the proposed post-processing algorithm for eliminating the false R peaks of record 104 in the MITD-AR. Refereeing to the annotated R peaks, it is observed that 8 peaks are detected by wrong due to the muscular noise in the ECG signal. With the help of the post-processing algorithm, 5 false peaks (i.e. marked by a red star symbol) are eliminated.

Figs. 7a and b illustrate the performance of the proposed detection algorithm on different ECG records with in-challenge cases of pointed T-waves and PVC complexes, respectively. It is noticed that the proposed NEO and differentiation techniques enlarge the R peaks compared to P and T-waves. Note that low-level R peaks in comparison to P- or T-waves in some records like 108, 116, and 203 increase the difficulty of detecting these R peaks, especially in the presence of muscular noise as shown in Figs. 7c, and d.

B. Stage 2: R Peak Classification

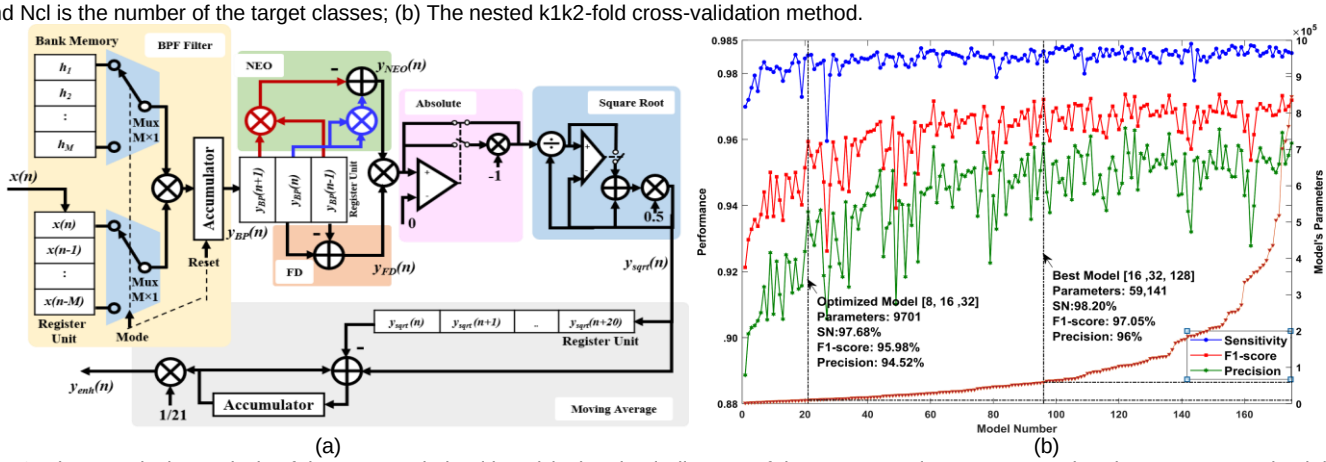
For beat classification, a segment of ECG with a duration of 0.3 s (i.e. 75 samples at f_s=250 Hz) centered at the detected R peak is passed directly to the network. The similarity between patterns of some beat types increases the difficulty of the beat classification. For example, the atrial premature beat (A) resembles a normal sinus beat, but its cardiac cycle is less than that of the normal [25]. To address this issue, the $RR_{i,i-1}$ interval of the current beat is passed as a feature with the data of the beat to the classification network. As a result, 76 samples are used as features for the heartbeats classification.

Fig. 8a shows the block diagram of the proposed 1D-CCNN network for beat classification. It consists of 3 convolution units with different numbers of kernels (*Kn1, Kn2, and Kn3*), 1 fully connected layer (FCL) with N_{cl} hidden units, softmax, and a classification layer. The convolution unit is used to extract the local feature map and consists of a 1D convolution layer with a kernel size of 5×1, rectified linear activation (ReLu), and



(a) (b)

Fig. 8 (a) The architecture of the proposed 1D-CCNN network; Kn1, Kn2 and Kn3 are the number of kernels in the three conv layers of the model, and Ncl is the number of the target classes; (b) The nested k1k2-fold cross-validation method.



(a) (b)

Fig. 9. The complexity analysis of the proposed algorithm. (a) The circuit diagram of the preprocessing stage; note that the square root circuit is designed based on the Babylonian algorithm, (b) Optimization of the proposed network model.

maxpool layer. The FCL layer is used to analyze the extracted feature map and convert the output size of the previous layers into the number of events to classify. The softmax layer calculates the probability of each target class over all possible target classes. Finally, the classification output layer estimates the cost function to detect the target output.

In the *k*-fold cross-validation, the best hyperparameter settings are selected based on the average *k*-fold performance of the same test set. This may cause significant bias and the model performance will not be unbiased anymore [26]. Recently, a nested k_1k_2-fold cross-validation is developed in [27] for evaluating the performance of the model by rotating data among training, evaluation, and testing as illustrated in Fig. 8b. So, the nested cross-validation is adopted to measure the generalization capabilities of the model. Firstly, the dataset is divided into k_1 partitions through k_1 iterations (i.e., outer loop), where (k_1-1) partitions are used as a training set (DS1) and one partition is used as a testing set (DS2) through a rotation manner in each iteration. The imbalance in the classes' data biases the model towards the classes with a high population. Therefore, resampling techniques [28] are used to balance data of classes in DS1 before training. In each outer loop, DS1 is resampled and goes through 10-fold cross-validation (*i.e.*, k_2=10, inner loop); where in each inner loop, 90% of DS1 is used for training the model, and 10% of DS1 is used for validating the model, while DS2 is used for testing the network. The average results of the $k_1 \times k_2$ folds are taken as the final values.

Practically, the system should detect the QRS complexes first before classifying them. In most previous works, authors focused only on the classification task, so they evaluated their classification algorithms using the original annotated QRS data. Ignoring the RLE error of the detection algorithm could impact the performance of the classifier. So, we train our model using the annotated QRS data (DS1) and evaluate it using the QRS complexes of DS2 that are detected by our algorithm, which is close to practical use. The trained model expects segments centered around the R peak, but the detected R peak location could be a distance from the actual R location (annotated one) due to the RLE. Therefore, a data augmentation method based on a circular shift is applied to DS1 to train the model to classify the QRS properly even if it was shifted, which reduces the effect of RLE error. In the augmentation process, each QRS beat of DS1 is augmented to 10 segments in addition to the original one. The 10 segments are obtained by shifting the original QRS beat 5 times by 1 sample to the right and 5 times by 1 sample to the left as illustrated by red arrows in Fig. 1. The cross-entropy is used as a loss function and the adaptive moment estimation (ADAM) training algorithm was adopted for backpropagation with a maximum number of epochs is 100. The main hyperparameters used for the ADAM algorithm were: learning rate is 0.001 which is dropped out by a rate of 75% after every 10 epochs and a mini-batch size of 512.

TABLE II
COMPARISON OF THE RESOURCE OF QRS DETECTION

Task		R	A	M	C	M
Preprocessing step	Filtering	1	1	1		2
	Eq(1)	1	1	2		
	Eq(2)		1			
	Eq(3)		1	4	2	
	Eq(4)	1	2	1		
Decision-making step	Avg_{pk}	1	2	1		
	$Th(n)$		2	2	1	
False peaks	d_{01}, d_{02},		3		4	
Total (Our Work)		4	13	11	7	2
Elgendi's algorithm [29]		6	15	11	4	-
Pan and Tompkins [16]		10	24	23	2	-

R: Registers, A: Adders, M: Multipliers, C: Comparators, and M: Multiplexers

IV. COMPLEXITY ANALYSIS AND HARDWARE IMPLEMENTATION

The block diagram for the implementation of the proposed QRS detection algorithm is shown in Fig. 9a. The hardware resources are summarized in Table II, and compared with existing methods. The cost of implementing the classifier depends on the input data size and shape, the number of layers, kernel size, and the number of kernels for each convolution (Conv) layer. To find an optimal solution, we build 170 models by varying the number of kernels in three Conv layers of the proposed structure and evaluate each model using overall accuracy (Overall_Acc), accuracy (Acc), sensitivity (SN), specificity (Spec), precision (Prec), and F1-score [30]. Fig. 9b shows the performance against the complexity of the different models. It is observed that the best results of SN (98.2%), F1-score (97.05%), and Prec (96%) are obtained when the kernels' number of the three Conv layers are set by 16, 32, and 128 kernels respectively, resulting in 59,141 parameters. To trade-off between the performance and the complexity, the model with a configuration of 8, 16, and 32 kernels is chosen. The optimized model has only 9,701 parameters and achieved SN of 97.68%, F1-score of 95.98%, and Prec of 94.52%.

In order to show the simplicity of the proposed NEO-CCNN algorithm, a simple microcontroller (STM32F407 MCU) with SRAM of 192 Kbytes is used for implementing the algorithm. As the FIFO size is only 2Kbytes and supports only uint16 data type, it can only store 4s data blocks at f_s=250 Hz. This is not sufficient for processing QRS detection. Thus, the ECG data are down-sampled to 125Hz to allow for 8s data block. To make use of uint16 data, ECG data block $x(n)$ is quantized as: 量化

$$\tilde{x}(n) = \frac{x(n) - \min(x(n))}{\Delta};\ \Delta = \frac{V_{pp}}{2^{16}-1}, \quad (9)$$

where V_{pp} is the peak-to-peak voltage of $x(n)$, Δ is the quantization step and $\tilde{x}(n)$ is the quantized signal. The shifted value and avg_{RR} (of detected QRS peaks of previous blocks) are sent as overhead with each data block $x(n)$.

Table III compares the performance of the hardware system and the software implementation. With the proposed quantization process, the impact of truncation error is eliminated 截断误差 resulting same performance in hardware and software implementations. The hardware system is run at three different clock frequencies of 8, 64, and 168 MHz. The execution time (Exe. time) of detecting and classifying R peaks into 5 classes (i.e. N, L, R, A, and V) of a 30 min record is 18 min, 2.2 min, and 46.61 s; and the consumption power is 6.24 mW, 28.57 mW, and 41.71 mW at clock frequencies of 8, 64, and 168 MHz, respectively. This means that the processing time is 103.67 µs/sample at a clock speed of 64 MHz, which makes the proposed system suitable for wearable ECG devices for detecting arrhythmia in real-time.

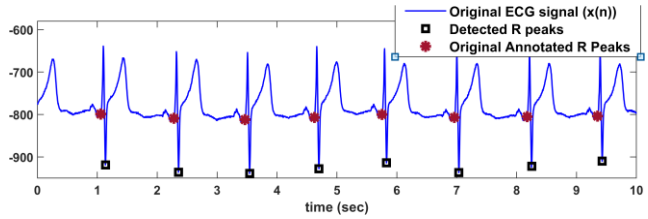


Fig. 10. Example of R peak detection in Record sele0129 of QTDB.

V. EVALUATION RESULTS

The NEO-CCNN algorithm is implemented using MATLAB®2019a software on a Laptop with an Intel® Core™ ™ i5 8th GEN 1.6 (up to 1.80) GHz processor and 8 GB RAM. Different metrics such as sensitivity *(SN)*, detection rate *(DR)*, positive predictivity (P^+) and the detection error rate (ER) [3, 31] are considered for evaluating the proposed QRS detection algorithm. Moreover, the root mean square (RMS) of R-peak Location Error (RLE_{RMSE}) and standard deviation error of heart rate variability (E_{HRV-SD} *sec)* metrics [3], defined by (10) and (11) respectively, are also used.

Definition 1 *RLE_{RMS}:* is the RMS of the time difference between the location of the detected peak (*DP*) and annotated peak (*AP*).

$$RLE_{RMS}\ (sec) = \sqrt{\frac{1}{TP}\sum_{i=1}^{TP}(AP_i - DP_i)^2}. \quad (10)$$

Definition 2 *E_{HRV-SD} sec :* is the RMS of the difference between SDNN of the detected peaks ($SDNN_{DP}$) and SDNN of the annotated peaks ($SDNN_{AP}$). $SDNN = \sqrt{\frac{1}{N_{RR}}\sum_{i=1}^{N_{RR}}(RR_i - \overline{RR_i})^2}$ is the standard deviation of all RR intervals.

$$E_{HRV-SD} = \sqrt{\frac{1}{N_{sub}}\sum_{i=1}^{N_{sub}}\left(SDNN_{AP_i} - SDNN_{DP_i}\right)^2}, \quad (11)$$

where *TP* is the number of true positive detected/classified R peaks. N_{sub} is the total number of subjects, and N_{RR} is the total number of RR intervals for each subject.

A.Performance of Proposed QRS Detection Algorithm

The performance of the proposed QRS detection algorithm depends on the segmentation window size (T_B). To determine the optimum value of T_B, we vary it from 5 s to 40 s while examining the *SN*. It is found that a T_B of 15s makes a good trade-off between the buffer size and the performance. Table IV lists the obtained results for 48 records of the MITD-AR using the proposed detection algorithm. It is observed that record 105 has the highest FPs because it contains high-grade noise and QRS like artifacts. The low amplitude of some R peaks and a high number of multiform premature ventricular complexes (PVCs) in records 116 and 203, respectively, lead to the highest *FN*s. Moreover, *SN* of 21 records reaches 100% and P^+ of 29 records achieved 100%. Only three records, 105, 108, and 203 have *DR* less than 99%. Compared to the given annotation of MITD-AR, the proposed detection algorithm achieved an average RLE_{RMS} of 7.94 ms and an average E_{HRV-SD} of 18.55 ms. It is noticed that some very low amplitude R peaks in record 116 make it challenging to detect them and lead to high RMS error between $SDNN_{DP}$ and $SDNN_{AP}$.

TABLE III
PERFORMANCE COMPARISON BETWEEN SOFTWARE IMPLEMENTATION AND HARDWARE IMPLEMENTATION AT FS=125 HZ

Method	Clock Frequency	Exe. Time /30 min record	Detection Results								Classification Results		
			TP	FN	FP	SN %	P^+ %	DR %	DTE_{MAE}	E_{HRV-SD}	Over Acc	SN %	Spec %
Software	Laptop 1.80GHz	6.442 sec	109,342	152	112	99.87	99.90	99.77	9.28 ms	18.34 ms	97.83	96.46	99.25
Hardware	168, 64, and 8 MHz	46.605,131.59, and 1084.2 sec	109,342	152	112	99.87	99.90	99.77	9.28 ms	18.34 ms	97.83	96.46	99.25

Compared with other methods, Table V shows the obtained results of the proposed algorithm. **MITD-AR:** It is obvious that our simple detection algorithm approaches the performance of the complex deep network method in [31]. Moreover, the performance of our detection algorithm is superior to the other methods. The values of *SN*, P^+, *DR*, and *ER* are improved by 0.341%, 0.422%, 0.76, and 78.75% compared to [32], and by 0.05%, 0%, 0.046%, and 18.4% compared to the best results presented in [33]. Compared to Tompkins [16], our *SN*, P^+, *DR*, *ER*, $RLE_{RMS,}$ and E_{HRV-SD} are improved by 0.77%, 0.29%, 1.11%, 84.3%, 21.46%, and 93.38% respectively. In terms of *RLE* and *HRV* error, the proposed method has the smallest values compared to [3, 16].

QT Database: It is observed that our algorithm approaches the performance of [3], while it has the smallest RLE_{RMS} and *ER*. Moreover, it outperforms the other methods in terms of *SN*, P^+, *DR*, and *ER*. For example, *SN*, P^+, *DR*, and *ER* are improved over [34] by 0.05%, 0.2%, 0.24%, and 87.86%, respectively. It is noticed that the obtained *RLE* of the QTDB is high compared to that obtained in MITD-AR due to the inaccurate annotation of R peaks in QTDB as shown in Fig. 10. The obtained results illustrate that the proposed method can efficiently detect R peaks of different clinical ECG signals with minimum *RLE* and *HRV* errors compared to the existing methods. Based on our implementation using Matlab, the proposed detection algorithm requires 0.22 s compared to 0.27 s of the Tompkins algorithm [16] for detecting QRS complexes of a 30-min record at a sampling rate of 250 Hz. This means our pre-sample execution time is only 0.49 µs compared to 0.6 µs of the Tompkins algorithm that makes our algorithm is suitable for real-time applications and wearable ECG devices.

B. Performance of NEO-CCNN Classification Algorithm

To study the effect of the QRS detection algorithm on the classification accuracy, the classification model is tested using our detected QRS data compared to the annotated QRS data. As k_1 is set by 2, 50% of annotated data is used as DS1 for training, and three cases for DS2 are considered for testing namely, 1) DS2 is the rest 50% of the annotated QRS data, 2) DS2 is the detected QRS data of the rest 50% of annotated QRS data using our detection algorithm, and 3) DS2 is the detected QRS data of the rest 50% using Tompkins algorithm [16]. Table VI shows the results of the three cases for classifying the ECG beats of the MITD-AR database into 5 classes (i.e. N, L, R, A, and V). It is observed that the annotated QRS data (case 1) achieves the best results compared to cases 2 and 3 which are affected by the RLE error of the detection algorithms. As the QRS detection algorithm results smaller RLE error, the classification results are improved by 1.52%, 0.59%, 4.43%, 0.66%, 3.68%, and 3% in terms of *overall_Acc*, *average Acc*, *SN*, *Spec*, *F1-score*, and *Prec*, respectively.

To improve the performance and avoid the effect of RLE, the QRS beats in DS1 are augmented before training by circular shifting to right and left. It is observed from Table VI that augmentation of the training data makes the model adaptive to RLE errors. This reduces the effect of the RLE compared to cases 2 and 3, which allows the performance approach that of the annotated QRS (case 1). Considering data augmentation with our QRS detected data (i.e. case 2 + Aug) improves the performance by 1.82 %, 0.72%, 2.30%, 0.62%, 4.49%, and 5.61% for *Overall_Acc*, *Acc*, *SN*, *Spec*, *F1-score*, and *Prec*, respectively. The confusion matrices before and after applying the augmentation for our algorithm are given in Table VI.

To compare the robustness of the proposed classification algorithm to the other algorithms, different classification tasks are considered (see Table VII) for classifying the ECG beats of the MITD-AR database into the following categories: **1)** 5 classes (i.e. N, L, R, A, and V); **2)** 5 classes (i.e. N, L, R, F, and V); **3)** 5-AAMI classes (i.e. NB, SB, VB, FB, and QB); **4)** 2 classes (i.e. SVEB+ {NB, SB} and VEB+ {VB, FB}); and **5)** 14 classes (i.e. N, A, V, Q, /, f, F, j, L, a, J, R, E, and e), where the class S is excluded because it has only two beats that are not enough for training. Note that the proposed algorithm is robust because its performance is measured based on all 48 records of the MITD-AR database compared to the other algorithms which excluded the poor-quality records. Furthermore, the performance of the proposed algorithm is similar or slightly less than those of complex structured networks [8-10]. It takes 2.28 ms for detection and classifying one QRS beat.

VI. CONCLUSION

A robust QRS detection and arrhythmia classification algorithm for portable embedded systems and wearable devices, called the NEO-CCNN algorithm, is proposed in this paper. Incorporating an adaptive time-dependent threshold with the nonlinear energy operator technique increases the algorithm's robustness, resulting in a high detection rate and minimum R-peak location error (RLE). Moreover, to reduce the effect of RLE on classification performance, a compact 1D-CCNN network is constructed based on data augmentation. A nested k_1k_2-fold cross-validation method is introduced to evaluate the performance of our algorithm using the MIT-BIH Arrhythmia and QT databases. Simulation results show that the proposed algorithm is competitive with, and in some cases superior to, the compared ones in terms of simplicity, efficiency, and robustness. It has the ability to detect more than 99.79% of R peaks with an RLE of 7.94 ms for the MIT-BIH database. The detected R peaks are classified into 5 classes with an accuracy of 99.43% and sensitivity of 97.68% using the optimized CCNN model with only 9,701 parameters. Our implementation of the NEO-CCNN algorithm takes 6.44 s and 46.61 s for detecting and classifying R beats of a 30 min record using a laptop with an i5 processor and the STM32F407 microcontroller, respectively. This makes our algorithm suitable for wearable ECG devices for classifying cardiac diseases. Future work should be continued to find a solution for

TABLE IV
OBTAINED RESULTS OF THE PROPOSED METHOD USING A 15 S SEGMENTATION WINDOW.

Record		Actual beats		*TP*		*FN*		*FP*		*SN %*		*P⁺ %*		*DR %*		*RLE$_{RMS}$ (ms)*		*|SDNN$_{DP}$ - SDNN$_{AP}$| (ms)*	
100	**201**	2273	1963	2273	1957	0	6	0	0	100	99.69	100	100	100	99.69	3.83	5.74	0.06	2
101	**202**	1865	2136	1865	2132	0	4	3	0	100	99.81	99.84	100	99.84	99.81	4.8	4.6	1.96	0.13
102	**203**	2187	2980	2187	2963	0	17	0	13	100	99.43	100	99.56	100	98.99	15.22	10.2	0.7	7.73
103	**205**	2084	2656	2084	2650	0	6	0	0	100	99.77	100	100	100	99.77	4.17	4.57	0.05	0.6
104	**207**	2229	1860	2228	1857	1	3	8	0	99.96	99.84	99.64	100	99.60	99.84	10.12	7.89	10.6	2.8
105	**208**	2572	2955	2565	2942	7	13	22	4	99.73	99.56	99.15	99.86	98.87	99.42	6.1	42.6	4.46	10.01
106	**209**	2027	3005	2026	3005	1	0	1	1	99.95	100	99.95	99.97	99.90	99.97	5.31	4.5	1.16	0.26
107	**210**	2137	2650	2135	2645	2	5	1	2	99.91	99.81	99.95	99.92	99.86	99.74	3.89	4.77	0.57	1.1
108	**212**	1763	2748	1756	2748	7	0	12	0	99.60	100	99.32	100	98.92	100	19.1	4.1	7.95	0.04
109	**213**	2532	3251	2532	3250	0	1	0	0	100	99.97	100	100	100	99.97	5.37	7.82	0.58	3.9
111	**214**	2124	2262	2123	2257	1	5	2	1	99.95	99.78	99.91	99.96	99.86	99.73	5.31	6.34	6.5	1.57
112	**215**	2539	3363	2539	3363	0	0	0	0	100	100	100	100	100	100	4.56	4.87	0.04	0.02
113	**217**	1795	2208	1795	2205	0	3	0	0	100	99.86	100	100	100	99.86	4.21	13	0.07	13.02
114	**219**	1879	2154	1877	2154	2	0	4	0	99.89	100	99.79	100	99.68	100	28.89	6.47	6.83	0.61
115	**220**	1953	2048	1953	2047	0	1	0	0	100	99.95	100	100	100	99.95	4.6	4.78	0.01	0.92
116	**221**	2412	2427	2392	2425	20	2	2	0	99.17	99.92	99.92	100	99.09	99.92	5.4	4.88	123.86	0.33
117	**222**	1535	2483	1535	2471	0	12	0	1	100	99.52	100	99.96	100	99.48	14.19	4.38	4.15	1.41
118	**223**	2278	2605	2278	2604	0	1	0	0	100	99.96	100	100	100	99.96	4.09	5.95	0.07	2.28
119	**228**	1987	2053	1987	2052	0	1	1	19	100	99.95	99.95	99.08	99.95	99.03	22.59	6.55	15.61	8.58
121	**230**	1863	2256	1862	2256	1	0	0	0	99.95	100	100	100	99.95	100	3.96	5.17	1.08	0.02
122	**231**	2476	1571	2476	1571	0	0	0	0	100	100	100	100	100	100	4.53	3.66	0.02	0
123	**232**	1518	1780	1518	1780	0	0	0	2	100	100	100	99.89	100	99.89	4.66	4.45	0.04	0.41
124	**233**	1619	3079	1619	3078	0	1	0	0	100	99.97	100	100	100	99.97	5.58	13.67	1.42	13.32
200	**234**	2601	2753	2598	2753	3	0	5	0	99.88	100	99.81	100	99.69	100	5.32	4.23	1.33	0.04
48 Records		**109494**		**109359**		**126**		**104**		**99.89**		**99.91**		**99.80**		**7.94**		***E$_{HRV-SD}$*=18.55 ms**	

TABLE V
COMPARISON OF THE PROPOSED DETECTION METHOD WITH OTHER METHODS, (*) THE GIVEN RESULTS FROM OUR IMPLEMENTATION

DB	Method	Actual beats	TP	FN	FP	SN%	P⁺ %	DR %	ER%	RLE$_{RMS}$ (ms)	E$_{HRV-SD}$ (ms)
MITD-AR	HRWPT-CNN [31]	109,494	-	-	-	99.89	99.90	99.80	0.20	-	-
	Our Work	**109,494**	**109,359**	**126**	**104**	**99.89**	**99.91**	**99.796**	**0.204**	**7.94**	**18.55**
	Kai Zhao et al. [3]	109,996	-	-	-	99.81	99.88	99.69	0.31	12.2	44.6
	AT-CNN [35]	105,078	104,837	241	99	99.77	99.91	99.68	0.32	-	-
	Elgendi's algorithm [29]	109,985	-	-	-	99.78	99.87	-	-	-	-
	P. Phukpattaranont [36]	109,483	109,281	202	210	99.82	99.81	99.62	0.38	-	-
	T. W. Bae et al. [18]	109,510	109,295	215	214	99.80	99.80	99.61	0.39	-	-
	T. Nguyen et al. [34]	109,494	109,270	224	314	99.80	99.71	99.51	0.49	-	-
	M. Yochum et al. [17]	109,491	109,331	160	574	99.85	99.48	99.33	0.67	-	-
	T. Sharma et al. [37]	109,494	109,353	131	111	99.88	99.90	99.88	0.22	-	-
	D. Castells et al. [38]	109,494	108,880	614	353	99.44	99.68	99.11	0.89	-	-
	Pan and Tompkins [16]	109,494	108,611	883	380	99.12	99.62	98.70	1.3	10.11	280.2
QTDB	**Our Work**	**86,995**	**86,983**	**12**	**17**	**99.99**	**99.98**	**99.966**	**0.034**	**26.3**	**3.23**
	Kai Zhao et al. [3]	86,995	-	-	-	99.99	99.98	99.96	0.04	30.6	1.7
	T. Nguyen et al. [34]	85,353	85,302	51	188	99.94	99.78	99.72	0.28	-	-
	Pan and Tompkins [16]	86,995	86,594	401	28	99.44	99.96	99.40	0.60	25.2	346.3

TABLE VI
EFFECT OF OUR DETECTION ALGORITHM ON THE CLASSIFIER

DS2 data		Overall Acc	Acc	SN	Spec	F1	Prec
Our Work	Case 1	98.39	99.36	97.28	99.42	95.42	93.90
	Case 2	**96.80**	**98.72**	**95.48**	**98.89**	**91.86**	**89.50**
	Case 2 + Aug	**98.56**	**99.43**	**97.68**	**99.50**	**95.98**	**94.52**
Tompkins algorithm [16]	Case 3 [16]	95.35	98.14	91.43	98.24	88.60	86.89
	Case 3 + Aug	97.67	99.06	95.01	99.22	93.39	92.14

	Case 2					Case 2+ Aug				
	N	V	A	R	L	N	V	A	R	L
N	72750	490	1419	90	183	73963	241	582	43	120
V	163	6792	71	32	26	116	6931	22	7	13
A	202	26	2297	11	0	143	12	2378	7	0
R	122	15	12	7009	3	29	10	10	7115	1
L	261	58	3	2	7738	48	29	1	0	7986

the imbalanced problem of the available datasets which affect the generalization capability of the trained model. Also, we will work on utilizing the work of this paper for developing a wearable ECG device for arrhythmia classification.

对于现有数据集的不平衡问题，影响训练后模型的泛化能力，未来的工作还需要继续寻找解决方法

References

[1] (2019). *Cardiovascular diseases (CVDs)*. Available: https://www.who.int/en/newsroom/fact-sheets/detail/cardiovascular-diseases-(cvds)

[2] D. Gupta and N. Roistacher, "Arrhythmia Diagnosis and Management," in *Critical Care*, J. M. Oropello, S. M. Pastores, and V. Kvetan, Eds., ed New York, NY: McGraw-Hill Education.

[3] K. Zhao, Y. Li, G. Wang, Y. Pu, and Y. Lian, "A robust QRS detection and accurate R-peak identification algorithm for wearable ECG sensors," *Science China Information Sciences*, vol. 64, p. 182401, 2021/05/08 2021.

[4] X. Zhang and Y. Lian, "A 300-mV 220-nW Event-Driven ADC With Real-Time QRS Detection for Wearable ECG Sensors," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 8, pp. 834-843, 2014.

[5] Y. Zhao, Z. Shang, and Y. Lian, "A 13.34 μW Event-Driven Patient-Specific ANN Cardiac Arrhythmia Classifier for Wearable ECG Sensors," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 14, pp. 186-197, 2020.

[6] Q. Cai, X. Xu, Y. Zhao, L. Ying, Y. Li, and Y. Lian, "A 1.3 μW Event-Driven ANN Core for Cardiac Arrhythmia Classification in Wearable Sensors," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, pp. 3123-3127, 2021.

[7] M. I. o. Technology. *MIT-BIH ECG database*. Available: https://archive.physionet.org/cgi-bin/atm/ATM

[8] J. P. Allam, S. Samantray, and S. Ari, "SpEC: A system for patient specific ECG beat classification using deep residual network," *Biocybernetics and Biomedical Engineering*, vol. 40, pp. 1446-1457, Oct. 2020.

[9] Y. Ji, S. Zhang, and W. Xiao, "Electrocardiogram Classification Based on Faster Regions with Convolutional Neural Network," *Sensors*, vol. 19, pp. 1-18, June 2019.

[10] A. M. Alqudah, A. Albadarneh, I. Abu-Qasmieh, and H. Alquran, "Developing of robust and high accurate ECG beat classification by combining Gaussian mixtures and wavelets features," *Australas Phys Eng Sci Med*, vol. 42, pp. 149-157, Mar Jan 2019.

[11] J.-S. Huang, B.-Q. Chen, N.-Y. Zeng, X.-C. Cao, and Y. Li, "Accurate classification of ECG arrhythmia using MOWPT enhanced fast compression deep learning networks," *Journal of Ambient Intelligence and Humanized Computing*, May 2020.

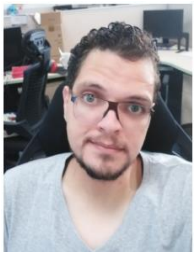
[12] U. R. Acharya, S. L. Oh, Y. Hagiwara, J. H. Tan, M. Adam, A. Gertych, *et al.*, "A deep convolutional neural network model to classify heartbeats," *Computers in Biology and Medicine*, vol. 89, pp. 389-396, Oct. 2017.

[13] J. Huang, B. Chen, B. Yao, and W. He, "ECG Arrhythmia Classification Using STFT-Based Spectrogram and Convolutional Neural Network," *IEEE Access*, vol. 7, pp. 92871-92880, 2019.

[14] S. Nabil, L. Yongfu, Z. Zhe, P. Yu, W. Guoxing, and L. Yong, "Detection of the Interictal Epileptic Discharges based on Wavelet Bispectrum Interaction and Recurrent Neural Network," *SCIENCE CHINA Information Sciences*.
[15] N. Sabor, H. Mohammed, Z. Li, and G. Wang, "BHI-Net: Brain-Heart Interaction-Based Deep Architectures for Epileptic Seizures and Firing Location Detection," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 30, pp. 1576-1588, 2022.
[16] J. Pan and W. J. Tompkins, "A Real-Time QRS Detection Algorithm," *IEEE Transactions on Biomedical Engineering*, vol. BME-32, pp. 230-236, 1985.
[17] M. Yochum, C. Renaud, and S. Jacquir, "Automatic detection of P, QRS and T patterns in 12 leads ECG signal based on CWT," *Biomedical Signal Processing and Control*, vol. 25, pp. 46-52, March 2016.
[18] T. W. Bae and K. K. Kwon, "Efficient Real-Time R and QRS Detection Method Using a Pair of Derivative Filters and Max Filter for Portable ECG Device," *Applied Sciences*, vol. 9, pp. 1-19, 2019.
[19] N. Ravanshad, H. Rezaee-Dehsorkh, R. Lotfi, and Y. Lian, "A Level-Crossing Based QRS-Detection Algorithm for Wearable ECG Sensors," *IEEE Journal of Biomedical and Health Informatics*, vol. 18, pp. 183-192, 2014.
[20] C. J. Deepu and Y. Lian, "A Joint QRS Detection and Data Compression Scheme for Wearable Sensors," *IEEE Transactions on Biomedical Engineering*, vol. 62, pp. 165-175, 2015.
[21] P. Laguna, R. G. Mark, A. Goldberg, and G. B. Moody, "A database for evaluation of algorithms for measurement of QT and other waveform intervals in the ECG," presented at the Computers in Cardiology 1997, Lund, Sweden, 1997.
[22] R. Mark, "AAMI-recommended practice : Testing and reporting performance results of ventricular arrhythmia detection algorithms," *Association for the Advancement of Medical Instrumentation, Arrhythmia Monitoring Subcommittee, AAMI ECAR, 1987*, 1987 1987.
[23] S. Mukhopadhyay and G. C. Ray, "A new interpretation of nonlinear energy operator and its efficacy in spike detection," *IEEE Transactions on Biomedical Engineering*, vol. 45, pp. 180-187, 1998.
[24] T. Teijeiro, P. Félix, J. Presedo, and D. Castro, "Heartbeat Classification Using Abstract Features From the Abductive Interpretation of the ECG," *IEEE Journal of Biomedical and Health Informatics*, vol. 22, pp. 409-420, 2018.
[25] D. M. Shindler and J. B. Kostis, "Chapter 13 - Electrocardiographic Technology of Cardiac Arrhythmias," in *Sleep Disorders Medicine (Third Edition)*, S. Chokroverty, Ed., ed Philadelphia: W.B. Saunders, 2009, pp. 182-187.
[26] S. Raschka, "Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning," *ArXiv*, vol. abs/1811.12808, 2018.
[27] J. Wainer and G. Cawley, "Nested cross-validation when selecting classifiers is overzealous for most practical applications," *Expert Systems with Applications*, vol. 182, p. 115222, 2021/11/15/ 2021.
[28] A. Estabrooks, T. Jo, and N. Japkowicz, "A Multiple Resampling Method for Learning from Imbalanced Data Sets," *Computational Intelligence*, vol. 20, pp. 18-36, Jan. 2004.
[29] M. Elgendi, "Fast QRS Detection with an Optimized Knowledge-Based Method: Evaluation on 11 Standard ECG Databases," *PLOS ONE*, vol. 8, pp. 1-18, 2013.
[30] Z. Chen, J. Luo, K. Lin, J. Wu, T. Zhu, X. Xiang, *et al.*, "An Energy-Efficient ECG Processor With Weak-Strong Hybrid Classifier for Arrhythmia Detection," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, pp. 948-952, 2018.
[31] M. Jia, F. Li, J. Wu, Z. Chen, and Y. Pu, "Robust QRS Detection Using High-Resolution Wavelet Packet Decomposition and Time-Attention Convolutional Neural Network," *IEEE Access*, vol. 8, pp. 16979-16988, 2020.
[32] S.-W. Chen, H.-C. Chen, and H.-L. Chan, "A real-time QRS detection method based on moving-averaging incorporating with wavelet denoising," *Computer Methods and Programs in Biomedicine*, vol. 82, pp. 187-195, June 2006.
[33] Z. Zidelmal, A. Amirou, D. Ould-Abdeslam, A. Moukadem, and A. Dieterlen, "QRS detection using S-Transform and Shannon energy," *Comput Methods Programs Biomed*, vol. 116, pp. 1-9, Aug 2014.
[34] T. Nguyen, X. Qin, A. Dinh, and F. Bui, "Low Resource Complexity R-peak Detection Based on Triangle Template Matching and Moving Average Filter," *Sensors (Basel, Switzerland)*, vol. 19, pp. 1-17, 2019.
[35] Y. Xiang, Z. Lin, and J. Meng, "Automatic QRS complex detection using two-level convolutional neural network," *Biomed Eng Online*, vol. 17, pp. 1-13, Jan. 2018.
[36] P. Phukpattaranont, "QRS detection algorithm based on the quadratic filter," *Expert Systems with Applications*, vol. 42, pp. 4867-4877, July 2015.
[37] T. Sharma and K. K. Sharma, "A new method for QRS detection in ECG signals using QRS-preserving filtering techniques," *Biomed Tech (Berl)*, vol. 63, pp. 207-217, March 2018.
[38] D. Castells-Rufas and J. Carrabina, "Simple real-time QRS detector with the MaMeMi filter," *Biomedical Signal Processing and Control*, vol. 21, pp. 137-145, Aug. 2015.
[39] L. El Bouny, M. Khalil, and A. Adib, "ECG Beat Classification Based on Stationary Wavelet Transform," presented at the Mobile, Secure, and Programmable Networking, Cham, 2019.
[40] N. Kar, B. Sahu, S. Sabut, and S. Sahoo, "Effective ECG Beat Classification and Decision Support System Using Dual-Tree Complex Wavelet Transform," in *Advances in Intelligent Computing and Communication*, Singapore, 2020, pp. 366-374.
[41] Ö. Yıldırım, P. Pławiak, R.-S. Tan, and U. R. Acharya, "Arrhythmia detection using deep convolutional neural network with long duration ECG signals," *Computers in Biology and Medicine*, vol. 102, pp. 411-420, Nov. 2018.
[42] S. S. Xu, M. W. Mak, and C. C. Cheung, "Towards End-to-End ECG Classification With Raw Signal Extraction and Deep Neural Networks," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, pp. 1574-1584, 2019.
[43] M. Alfaras, M. C. Soriano, and S. Ortín, "A Fast Machine Learning Model for ECG-Based Heartbeat Classification and Arrhythmia Detection," *Frontiers in Physics*, vol. 7, pp. 1-11, July 2019.

Nabil Sabor (M'19) received his B.S.E.E and M.S.E.E. degrees in electrical engineering in 2006 (excellent with honors) and 2011 respectively, both from Assiut University, Egypt. In 2016, he received Ph. D. degree from the Niigata University, Japan, and Assiut University (channel system). In addition, he got another Ph.D. degree from Niigata University, Japan in 2017. Nabil Sabor is an Assistant Professor with the Department of Electronics and Communication Engineering, Assiut University. Currently, he is a postdoctoral researcher with the Department of Micro and Nano Electronics Engineering and MoE Key Lab of Artificial Intelligence, Shanghai Jiao Tong University, China. His research interests include signal and image processing, data compression, biomedical signal processing, wavelet-transforms, digital filters, genetic algorithms, immune algorithms, wireless sensor networks, compressive sensing, machine and deep learning techniques. He is a member of IEEE Circuits and Systems Society (IEEE CAS).

Garas Gendy received his B. Sc. and M.Sc. degrees in 2009 and 2016, respectively, from Assiut University, both in Electrical and Electronic Engineering. He is currently a Ph.D. student at Shanghai Jiao Tong University, Shanghai, China. His research interests include artificial intelligence, deep learning, image processing, computer vision, fuzzy systems. He is a member of IEEE Circuits and Systems Society (IEEE CAS).



Hazem Mohammed received the B.S. and M.S. degrees in electrical engineering in 2010 and 2014, respectively, from Assiut University, Assuit, Egypt. He is currently Ph.D. student at Shanghai Jiao Tong University, Shanghai, China. His research includes signal and image processing, artificial intelligence, machine learning, wireless cooperative communication. He is a student member of IEEE Circuits and Systems Society (IEEE CAS).

Guoxing Wang, (M'06, SM'13) received his Ph.D. in electrical engineering from the University of California at Santa Cruz, US, in 2006. He was a Member of the Technical Staff in Agere Systems, San Jose, California, from 2006 to 2007. In 2007-2009, he joined the Second Sight Medical Products, Sylmar California, where he designed the integrated circuits chip that went into the eyes of patients to restore vision. Currently, he is a Professor in the School of Microelectronics, Shanghai Jiao Tong University, Shanghai, China. He has published in various peer-reviewed journals and conferences. His current research interests include biomedical electronics, bio-inspired circuits and systems. Dr. Wang serves as Editor-in-Chief for IEEE Transactions on Biomedical Circuits and Systems (TBioCAS) (2020-2021). He is a member of the IEEE Biomedical Circuits Systems Technical Committee (BioCAS). He served as an Associate Editor for IEEE Transactions on Circuits and

Systems II (2012-2015), Guest Editor for IEEE Journal on Emerging and Selected Topics in Circuits and Systems (JETCAS) and Guest Editor and deputy Editor-in-Chief for IEEE Transactions on Biomedical Circuits and Systems. He served as the technical program chair for IEEE Conference on Biomedical Circuits and Systems in 2016 and IEEE International Symposium on Integrated Circuits and Systems in 2020. He was the local chair for the first IEEE Green Circuits and Systems (ICGCS) in 2010 and for the second Asia Pacific Conference on Postgraduate Research in Microelectronics & Electronics (PrimeAsia), in 2010. He serves as Vice President for IEEE Circuits and Systems Society (2019-2022).

Yong Lian (M'90–SM'99–F'09) has research interests that include biomedical circuits and systems and signal processing. He was the recipient of more than 15 awards for his research, including IEEE Circuits and Systems Society's Guillemin–Cauer Award, IEEE Communications Society Multimedia Communications Best Paper Award, Institution of Engineers Singapore Prestigious Engineering Achievement Award, and Winning of the Design Contest Award in ISLPED 2015. He is the President of the IEEE Circuits and Systems (CAS) Society, Chair of the IEEE Periodicals Partnership Opportunities Committee, member of the IEEE Periodicals Committee, member of the IEEE Biomedical Engineering Award Committee, and member of Steering Committee of the IEEE Transactions on biomedical circuits and systems. He was the Editor-in-Chief for the IEEE Transactions on circuits and systems Part II for two terms. He served as the VP for Publications and VP for Region 10 of the IEEE CAS Society, and many other roles in IEEE. He is the Founder of the IEEE Biomedical Circuits and Systems Conference and the Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics (PrimeAsia). He is a Fellow of the Academy of Engineering Singapore.

TABLE VII
COMPARISON OF THE PROPOSED CCNN NETWORK WITH OTHER METHODS

Method	Classes {classes names}	Records (beats No.)	Validation Method	Classifier's Input	Classifier (Parameter)	Over Acc %	Avg Acc%	SN %	Spec %
Our Work	**5 classes {N (75052), L (8075), R (7259), A (2546), V (7130)}**	**48 Records (100,062 beats)**	**Nested (2,10)-fold {50% (10-fold) Tn & Vd, 50% Tst}**	**76** samples	**1D-CCNN (9,701)**	**98.56**	**99.43**	**97.68**	**99.50**
SWT-SVM [39]	5 classes {N (3150), L (300), R (300), A (300), V (395)}	42 Record, excluded {102,104,107, 108, 203, & 208} (4,445 beats)	10-folds {90% Tn, 10% Tst)	250 samples	SVM	98.74	99.49	97.27	99.54
MOWPT / FCResNet [11]	5 classes {N (540), L (540), R (540), A (540), V (360)}	30 records (2,520 beats)	1-fold {80% Tn, 20% Tst}	3600 samples	FCResNet	-	98.79	95.16	-
DT-CWT method [40]	5 classes {N, L, R, A, V}	-	10-fold	-	KNN	-	98.92	96.45	96.16
Gaussian mixtures and DWT [10]	6 classes {N, L, R, a, A, and V}	30 records (10,502 beats)	1-fold {75% Tn, 25% Tst}	360 samples	RF PNN	- -	99.97 99.99	99.81 99.99	99.96 99.97
Our Work	**5 classes {N (75052), L (8075), R (7259), V (7130), F (803)}**	**48 Records (98,319 beats)**	**Nested (2,10)-fold {50% (10-fold) Tn & Vd, 50% Tst}**	**76** samples	**1D-CCNN (9,701)**	**98.92**	**99.57**	**97.25**	**99.61**
Faster R-CNN [9]	5 classes {N (8500), L (8075), R (6857), V (6900), and F (803)}	40 Record, excluded {102,104,107, 209, 217, 220, 222 & 232} (31,135 beats)	1-folds {50% Tn ,50% Tst}	500×375 samples	CNN net. RPN net, and 2D R-CNN net.	98.04	99.21	98.06	99.45
Our Work	**5-AAMI classes {NB (90631), SB (2781), VB (7236), FB (803), QB (8043)}**	**48 Records (109,494 beats)**	**Nested (2,10)-fold {50% (10-fold) Tn & Vd, 50% Tst}**	**76** samples	**1D-CCNN (9,701)**	**96.97**	**98.79**	**95.91**	**99.10**
9-layer-CNN [12]	5-AAMI classes {NB (90592), SB, VB (7235), FB (802), QB (8039)}	48 Records (109,449 beats)	10-folds {90% Tn, 10% Tst)	260 samples	1D-CNN (19,805)	-	94.03	96.71	91.54
ST-based 2D ResNet [8]	5-AAMI classes {NB (46074), SB (3871), VB (6816), FB (1413), QB (62)}	44 Record, excluded {102,104,107, and 217} (58,236 beats)	1-fold {15% Tn, 85% Tst}	256×256 samples	**2D-ReNet with 12 residual paths**	**99.73**	**98.86**	**98.84**	**99.5**
Our Work	**14 classes**	**48 Records (109,492 beats)**	**Nested (2,10)-fold {50% (10-fold) Tn & Vd, 50% Tst}**	**76** samples	**1D-CCNN (15,470)**	**95.11**	**99.30**	**86.1**	**99.6**
1D-CNN [41]	13 15 17	1000 segments	1-fold {70% Tn, 15% Vd, 15% Tst}	3600 samples	2,270,991	95.20 92.51 91.33	- - -	93.52 88.57 83.91	99.61 99.39 89.52
Our Work	**2 classes SVEB+ {NB (90631), SB (2781)}, VEB+ {VB (7236), FB (803)}**	**48 Records (101,451 beats)**	**Nested (2,10)-fold {50% (10-fold) Tn & Vd, 50% Ts}**	**76** samples	**1D-CCNN (7,778)**	**99.19**	**99.19**	**99.27**	**98.25**
Abstract features-based method [24]	2 classes SVEB+ , VEB+	44 Record	-	-	Clustering Algorithm	-	-	85.10	92.82
End-to-End DNN [42]	2 classes SB (2779) , VB (7007)	44 Record, excluded {102,104,107, and 217} (9,786 beats)	{50% (8-fold) Tn & Vd, 50% Tst}	417 samples	DNN (62,202)	94.7	94.7	77.3	93.7
Fast Machine Model [43]	2 classes SVEB+ {NB (89962), SB (2777)}, VEB+ {VB (7001), FB (802)}	44 Record, excluded {102,104,107, and 217} (100,542 beats)	{50% (5-fold) Tn & Vd, 50% Tst}	60 samples	Echo State Network (ESN)	98.6	98.6	84.4	99.7

CCNN: compact conventional neural network, RF: random forest, PNN: probabilistic neural networks, RPN: region proposal network, SVM: support vector machine, KNN: K-nearest neighbor, Tn: Train, Vd: Validation, Tst: Test.