

Energy-Efficient Real-Time Heart Monitoring on Edge–Fog–Cloud Internet of Medical Things

Berken Utku Demirel[✉], *Student Member, IEEE*, Islam Abdelsalam Bayoumy, and Mohammad Abdullah Al Faruque[✉], *Senior Member, IEEE*

Abstract—The recent developments in wearable devices and the Internet of Medical Things (IoMT) allow real-time monitoring and recording of electrocardiogram (ECG) signals. However, continuous monitoring of ECG signals is challenging in low-power wearable devices due to energy and memory constraints. Therefore, in this article, we present a novel and energy-efficient methodology for continuously monitoring the heart for low-power wearable devices. The proposed methodology is composed of three different layers: 1) a noise/artifact detection layer to grade the quality of the ECG signals; 2) a normal/abnormal beat classification layer to detect the anomalies in the ECG signals; and 3) an abnormal beat classification layer to detect diseases from ECG signals. Moreover, a distributed multioutput convolutional neural network (CNN) architecture is used to decrease the energy consumption and latency between the edge–fog/cloud. Our methodology reaches an accuracy of 99.2% on the well-known MIT-BIH Arrhythmia Data Set. Evaluation on real hardware shows that our methodology is suitable for devices having a minimum RAM of 32 kb. Moreover, the proposed methodology achieves 7x more energy efficiency compared to state-of-the-art works.

Index Terms—Arrhythmia, electrocardiogram (ECG), heart monitoring, Internet of Medical Things (IoMT), wearable systems.

I. INTRODUCTION

LECTROCARDIOGRAM (ECG) signals are widely used to detect cardiovascular diseases, which are the leading cause of death globally [1]. Moreover, according to the American Heart Association, the early detection of these diseases is crucial for patients' health [2]. Clinical ECG is the primary tool for monitoring cardiac activity. However, it can only be used for a limited time, and continuous monitoring of the patients' condition is still required outside clinical hours. Traditionally, ambulatory ECG devices are used to monitor the cardiac activity for a long duration to be further investigated by clinicians. For example, the Holter [3], a battery-operated portable device, is used to record and store long-term ECG signals. However, these devices cannot provide real-time feedback to users, and cardiologists need to analyze long-term recordings, which is a very time consuming and expensive process.

Manuscript received April 4, 2021; revised June 18, 2021; accepted December 14, 2021. Date of publication December 27, 2021; date of current version July 7, 2022. This work was supported in part by the National Science Foundation (NSF) under Award CMMI-1739503. (Corresponding author: Berken Utku Demirel.)

The authors are with the Department of Electrical Engineering and Computer Science, University of California at Irvine, Irvine, CA 92697 USA (e-mail: bdemirel@uci.edu).

Digital Object Identifier 10.1109/JIOT.2021.3138516

To solve this issue, several heart monitoring devices and solutions have been proposed and developed both in academia and industry [4], [5], thanks to rapid development on the Internet of Medical Things (IoMT) and smart health care systems. These heart monitoring devices or systems can be categorized into two different groups according to their methods. The first group [6], [7] analyzes long-term recorded ECG signals offline by using remote cloud servers. The utilized algorithms in the cloud provide a powerful classification performance. However, they cannot be implemented on the edge node due to their memory requirements and high energy consumption. Moreover, since all computing occurs in the cloud, the latency of the system increases, which weakens the user experience [8]. The latency is an essential factor for heart monitoring applications because the rapid detection of cardiovascular diseases is critical for people's lives. The second group [9], [10] provides a real-time solution by doing computation on the edge side rather than the cloud; however, the amount of time the device is monitoring the cardiac activity is limited due to constraints on battery life, which is the most valuable resource for the edge of the network [11].

Due to these problems, many people live for years unaware of their illness [12]. Some cases reported that deaths due to cardiovascular disease could have been prevented if the disease was detected earlier [13]. Therefore, continuous real-time ECG monitoring can be a vital solution for people with cardiovascular diseases.

The main challenge of designing a continuous real-time monitoring system is adhering to the devices' energy and memory constraints since processing requires lots of memory and is computationally intensive. To overcome this problem, existing works have proposed performing the computation at the proximity of data generation sources, which are the edge devices in this case, using fog or edge–cloud [14]–[16] architectures by transferring real-time signals to these nodes of network. However, this transfer operation requires tremendous communication power, which decreases the device's battery life and makes it difficult to sustain continuous monitoring over long periods. This kind of the IoMT system requires significant energy resources on the edge device, and is vulnerable to privacy issues [17], [18]. Several methods have been proposed to encrypt the ECG signals. However, they require additional energy and memory at the edge [19], [20]. Moreover, the transferred ECG signals might be contaminated with noise or artifacts caused by the users' mobility, which results in further unnecessary energy consumption. In

summary, the key research challenges associated with continuous heart monitoring are: 1) *developing resource-efficient algorithms on the edge* and 2) *detecting abnormalities in ECG signals as fast as possible to minimize latency*.

To address the above-mentioned challenges, this article proposes a novel heart monitoring methodology on a hybrid edge–fog/cloud IoMT. In this article, we define “edge” as the computing platform where the data acquisition is performed, and “fog” is defined as the device possessing further computing and network resources along the path between data sources and cloud data centers, e.g., a smartphone. Throughout this article, “edge–fog/cloud” is used since the proposed methodology is a solution for any two-tiered systems employing edge–cloud, edge–fog, or fog–cloud architectures. The proposed methodology delivers a layered software pipeline architecture by distributing the layers between edge and fog/cloud. The first layer running on the edge is designed to detect Noise/Motion artifacts. This detection aims to conserve energy resources through avoiding unnecessary artifacts transmission. The second layer, running as well on edge, classifies normal, and abnormal beats in the ECG. If the beats are classified as normal, the classification is considered complete, and only the heart rate value of that beat is transmitted instead of the raw signal. Else, the beats are classified as abnormal and are sent to the next layer in the hierarchy, which could be a fog or cloud, to be further classified. Moreover, these first two-layers can further reduce the energy consumption of the edge device through controlling the data sampling rate. If the signal is not classified as clean or the recorded ECG signal has no abnormal beats, the control unit changes the sampling rate to the degree that both classifiers still maintain high-performance classification. The last layer, running on fog or cloud, is a distributed multioutput convolutional neural network (CNN) to classify several cardiovascular diseases.

A. Motivational Example

We have done several experiments to show the advantages of the proposed methodology for continuous heart monitoring. In Case I, the 1-h raw ECG, which are acquired digital signals from sensors, are transferred to the fog/cloud server without any investigation or operation on the edge for noise and artefacts. Then, the energy consumption of these transfer operations is calculated for four different communication technologies: 1) *Wi-Fi*; 2) *LTE*; 3) *3G*; and 4) *BLE*. For Case II, the 1-h ECG record is partitioned into two parts: 1) a 50-min segment of clean ECG which has no artifacts with just regular beats and 2) a 10-min segment of the ECG signal, simulated as a noisy signal. The required communication energy for transferring this record is calculated again. Finally, in Case III the ECG signal is divided into three parts: 1) a 40 min regular beats segment; 2) a 10 min noisy signal; and 3) 10 min of the recording including several arrhythmias. The energy consumption for all cases’ transfer operation is shown in Fig. 1. While calculating the energy consumption of these cases, we have followed the $\mu\text{J}/\text{bit}$ values given in [23] for the *Wi-Fi*, *LTE*, and *3G*. For *BLE* protocol energy consumption, we performed the profiling for data exchange on an EFR32BG13 Blue Gecko

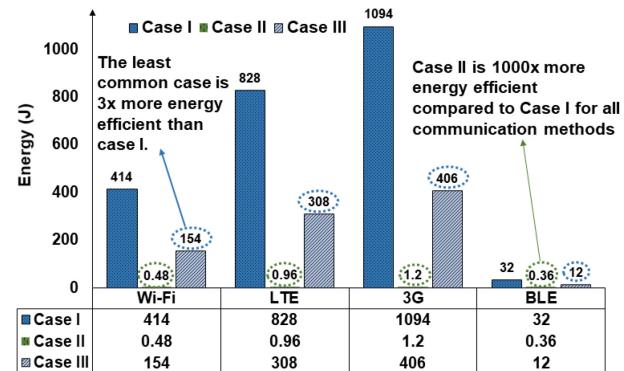


Fig. 1. Energy consumption of Cases I-III.

Bluetooth Low Energy SoC which has 32 bit ARM Cortex-M4 core with a 40-MHz maximum operating frequency.

Since the raw ECG data is transmitted without any classification on edge in Case I, its communication energy is the highest among the three cases regardless of the communication methods as shown in Fig. 1. However, when we applied our proposed methodology to detect the normal and noisy beats in the ECG signal and filtered out them before transmitting to the fog/cloud node (Cases II and III), the communication power can be decreased by $1000\times$ and $3\times$ for Cases II and III, respectively. Also, to make a fair comparison with different cases, we measure the additional computational energy consumption of our proposed methodology to detect the artifacts and abnormal beats in the ECG signals for Cases II and III. We observe that the algorithm’s energy consumption approximately 1.5 J for an hour, which makes Cases II and III still much more energy-efficient than Case I. Moreover, it is known that arrhythmias are not as frequent in people like Case III (10 min of an hour), so Case II will be more common for most of the people. Therefore, we observe that detecting the normal and noisy ECG beats in edge devices could save energy and time while reducing the communication channel usage and fog/cloud server load.

B. Novel Contributions

The novel contributions of this article are as follows.

- 1) A real-time continuous heart monitoring system that runs on a hybrid edge–fog/cloud architecture while being energy and memory efficient.
- 2) A novel layer-wise distributed multioutput CNN architecture that is optimized for decreasing the energy consumption and latency between edge and fog/cloud. To the best of our knowledge, we are the first to investigate the signal’s physiology in distributing the computational complexity of architecture between the different nodes of a network.
- 3) Evaluation of our proposed methodology on the well known MIT-BIH data sets [22], and PhysioNet/CinC (PICC) 2011 challenge [21], [25]. It shows that, our proposed methodology reaches or outperforms the current state-of-the-art works in terms of classification performance [26], [27], [34]–[36].

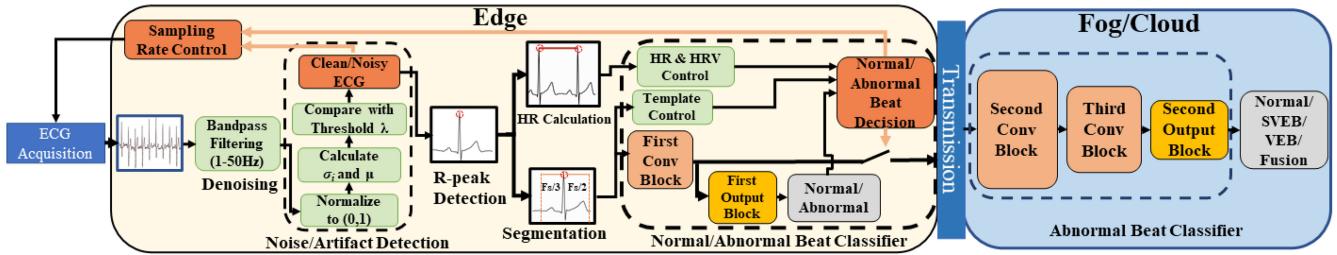


Fig. 2. Overview of our proposed methodology.

- 4) Evaluation on real hardware shows that each layer of the proposed methodology achieves up to 7× more energy efficiency compared to the state-of-the-art works [34]–[36].

The remainder of this article is organized as follows. In Section II, we review related works for heart monitoring systems. Section III describes the proposed methodology. Section IV shows the experimental setup. Section V discusses the results. Finally, a conclusion is drawn in Section VII.

II. RELATED WORK

A. Quality Assessment of ECG

Signal quality assessment (SQA) is the critical first step in continuous heart monitoring since it eliminates the noisy signals before the classification. Mostly, SQA methods grade ECG into two groups: 1) *acceptable* and 2) *unacceptable*. Existing methods extracted several features from the ECG signals and graded them using heuristic rules [26] or machine learning-based classifiers such as Support Vector Machine [27], [28]. For example, Clifford *et al.* [27] proposed extracting six different features based on time and frequency domains, whereas the extraction of these features depends on the accurate and reliable detection of the QRS complex in noisy ECG signals, which is a challenging task. To avoid detection of QRS complex, Satija *et al.* [26] examined three main causes of noise in ECG: 1) abrupt change; 2) signal absence; and 3) high-frequency noise. Since the authors do not detect the QRS complexes in ECG signals, they investigated the 10-s windows based on time and frequency. However, since the number of samples increases with a longer duration of ECG recording, the feature extraction-related algorithms' computation requirements increase, especially when the frequency-domain features are used. Since these SQA algorithms run in the proximity of data sources (edge of the networks) during acquisition, they need to be energy and memory efficient. Therefore, in this article, we proposed a lightweight SQA which can detect noise and artifact.

B. Heartbeat Classification of ECG Signals

A wide range of automatic heartbeat classification methods has been proposed, and these existing methods may be categorized into two groups. The first method [7], [9] extracts some handcrafted features from the ECG signals and feeds them to a classifier like SVM. For example, Venkatesan *et al.* [7], extract six frequency-domain features from the heart rate for

arrhythmic beat classification. Similarly, Tang *et al.* [9] used 32 time-domain features with linear kernel SVM to classify heartbeats. However, this feature extraction process increases the computational complexity and memory. Moreover, these extracted features may not represent the complete characteristics of the ECG and restrict the performance.

The second method is to directly send the original waveform to a neural network for classification, known as the end-to-end classification method to avoid the feature extraction process since the neural networks do not require feature engineering as they automatically extract features. For example, in [34], authors use a combination of the bidirectional recurrent neural network (BRNN) and CNN model to detect four kinds of heartbeats. Compared to feature-based classification methods, this method can reach higher accuracy. However, this combined architecture has tens of millions of parameters, making it unsuitable for the edge node with memory and energy constraints. Similarly, Wang *et al.* [36] proposed a two-stage neural network, which combines a multilayer perceptron and a CNN. The first stage classifies the ECG beats as normal or abnormal, and the second stage classifies the abnormal beats to several arrhythmias. Moreover, their proposed solution uses an additional classifier for discriminating abnormal beats, which introduces more parameters and energy consumption to the system. However, our proposed methodology uses a distributed multioutput CNN architecture to detect and filter out the regular beats during run-time without using any additional features and classifiers.

III. OUR PROPOSED METHODOLOGY

The proposed methodology, shown in Fig. 2, consists of three main layers and several processing blocks whose components are detailed in the following section.

A. Processing and Control Units

- 1) *Filtering*: We use a fifth-order linear phase bandpass filter with a Hamming window cut-off frequencies ($f_1 = 1$ Hz and $f_2 = 50$ Hz).
- 2) *R-Peak Detection*: To detect the R-peaks, we have used the Pan–Tompkins algorithm [29], a real-time QRS complex-based heartbeat detection approach that has an accuracy of up to 99.5%.
- 3) *Beat Segmentation*: We take $F_s/3$ samples before and $F_s/2$ samples after the R-peak. The highest heart rate is chosen as 180 beats per minute (BPM) to avoid overlapping two beats in a window.

4) *Heart Rate Calculation*: The heart rate is calculated using the following equation:

$$HR_{(i)} = \frac{60}{(R_{(i)} - R_{(i-1)})/F_s} \quad (1)$$

where $HR_{(i)}$ is the heart rate of the i th heartbeat segment in BPM. $R_{(i)}$ is the location of the R-peak in the i th heartbeat segment, and F_s is the sampling frequency.

5) *Sampling Rate Control*: This unit controls the ECG acquisition sampling rate according to the output of two different decision units. The first decision unit checks the ECG signal's quality. If it detects a noisy ECG signal, the sampling rate is decreased to the degree that all components maintain the high classification performance. The second decision unit monitors the incoming beats and classify them as normal and abnormal beats. If the beats are classified as normal, this unit decreases the sampling rate to the same degree.

B. Noise/Artifact Detection Layer

The ambulatory ECG signals are mostly contaminated with low-frequency motion artifacts that cannot be removed using simple filtering. Therefore, the first layer is designed to detect these artifacts. Since this algorithm runs on the edge, we have focused on developing a lightweight and robust algorithm to increase the device's battery life. We have observed that a rule-based decision method is superior compared to the classical machine learning algorithms (SVM, RF) considering energy efficiency while maintaining the classification performance for this task. Therefore, a rule-based algorithm is used to classify signals into two groups *acceptable* and *unacceptable*. The ECG signals are divided into 10-s windows. First, the windows are normalized with respect to the maximum amplitude value. The mean of a normalized window is obtained and compared with a threshold (λ) to detect the absence of an ECG signal. If the mean is lower than the threshold, the window is classified as *unacceptable* and the signal is discarded.

The abrupt changes and baseline wander are investigated using a moving standard deviation of the 10-s ECG signal. In this method, a window of a specified length ($2F_s/5$) is moved over the signal with a 70% overlap. The deviation of the signal (σ_i) is computed over the data by

$$\sigma_i = \sqrt{\frac{1}{N-1} \sum_{n=1}^N |x_i[n] - \mu_i|^2} \quad (2)$$

where σ_i and μ_i are the corresponding standard deviation and mean of the window, respectively. An example of σ_i waveforms is shown in Fig. 3. When the ECG signal has an abrupt change, the algorithm suppresses the beats and brings the artifacts forefront. Finally, to detect the abrupt changes, the mean of the waveform (σ_i) is obtained and compared with a threshold, which is set to 0.2 based on acceptable level of noise. If the mean is higher than the threshold, the signal is classified as acceptable.

The proposed method has the following advantages: 1) it is sensitive to abrupt changes and baseline wanders where if a small portion of the signal is corrupted with an artifact or

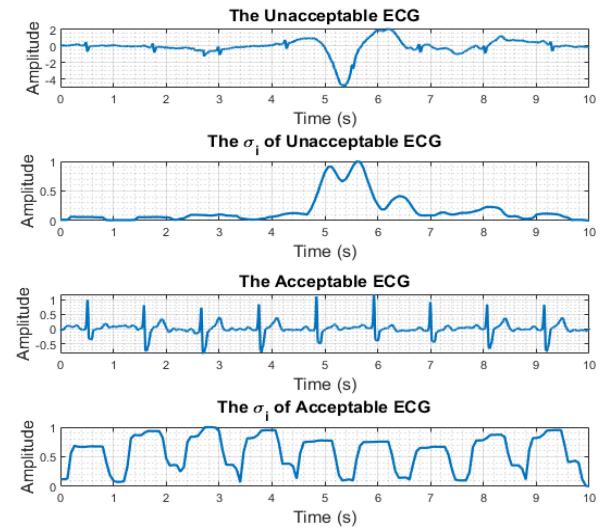


Fig. 3. *Acceptable* and the *Unacceptable* ECG signals from PICC [25] and obtained σ_i waveforms.

noise, it is detected and ignored and 2) it is energy efficient since it can run at a low sampling rate of 100–120 Hz without losing performance, and it does not need complex features to classify.

C. Normal/Abnormal Beat Classification

While classifying the normal and abnormal beats, we have followed the Association for the Advancement of Medical Instrumentation (AAMI) instructions, which are the golden standard for automatic heartbeats classification. According to the AAMI standard, heartbeats can be divided into N (normal), S (supraventricular ectopic beat), V (ventricular ectopic beat), F (fusion beat), and Q (unclassified beat) [30]. Therefore, the beats are divided into two groups for that layer. The first group only contains the normal beats N from the data sets, and the second group is composed of other types of heartbeats (S, V, F, and Q), which are the abnormal beats. If a beat is classified as normal in this stage, it is not transmitted to the fog/cloud node to save energy. Nevertheless, if a regular beat is mistakenly classified as abnormal, it can still be corrected by the next classifier. So, we need to ensure that the abnormal beats are classified with high sensitivity in that layer.

It is known that during arrhythmias, the heart rate deviates from its normal rhythm. These heart rate variations are common in the premature beats; as the name suggests, they occur when the ventricles or atrial contract too soon, out of sequence with the normal heartbeat. Moreover, as shown in Fig. 4, the waveform of the abnormal beats is different from the regular beats. Therefore, the heart rate variability (HRV) and correlation of the beats are indicators of abnormal beats. We have used these together with the result of the first output block to decide whether a beat is normal or abnormal.

If the first output block classifies a beat as abnormal, the output of the first convolutional is directly sent to the next classifier to be further examined. However, if a beat is classified as normal, other indicators just confirm the decision. The first

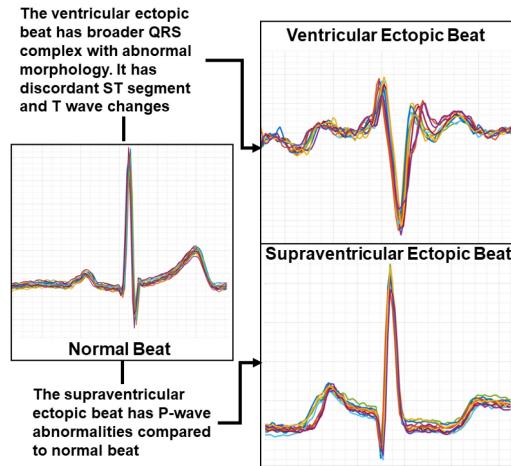


Fig. 4. Waveform of normal and abnormal beats.

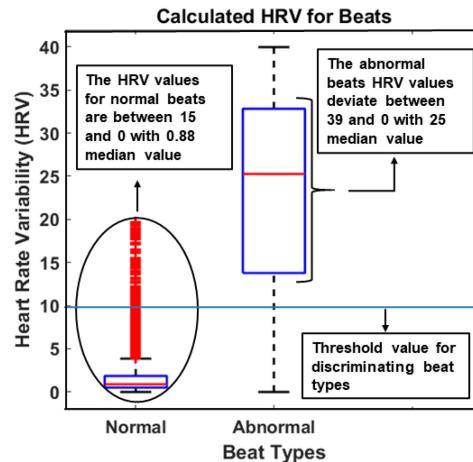


Fig. 5. HRV values for normal and abnormal beats.

indicator is the HRV between three consecutive beats, which is calculated as

$$F_S \times \left(\frac{1}{RR_{(i-1)}} - \frac{1}{RR_i} \right) \quad (3)$$

where RR_i is the latest interval and F_S is the signal's sampling rate. Fig. 5 is the whisker plot for HRV values of different beats. The threshold value is set to 10 based on variances of normal and abnormal beats. If the calculated HRV value is greater than 10, the last beat is classified as abnormal even though the first block classified it as a regular beat. It is also observed that most of the normal beats with HRV values greater than 10 are either before or after abnormal beats.

As a second indicator, we have calculated the correlation of a template and classified beat. While creating a template, we have used 20 different regular beats. These regular beats are aligned according to their R-peak and averaged, then the correlation is calculated as

$$\rho(T, x) = \frac{1}{N-1} \sum_{n=1}^N \left(\frac{T[n] - \mu_T}{\sigma_T} \right) \left(\frac{x[n] - \mu_x}{\sigma_x} \right) \quad (4)$$

where σ_T and μ_T are the standard deviation and mean of the template beat (T) and σ_x and μ_x are the standard deviation

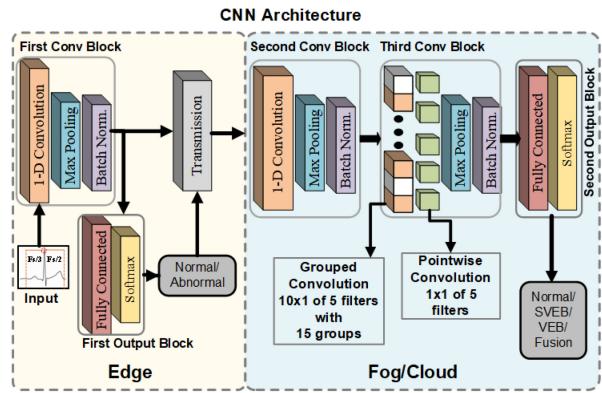


Fig. 6. Distributed multioutput CNN architecture.

TABLE I
MULTIOUTPUT CNN ARCHITECTURE DETAILS

Layer Name	Kernel Size	Stride Size	Activation Function	Output Shape	# of Param.
Input	-	-	-	105x1	0
Conv 1	64	2	ReLU	53x5	325
Pooling 1	2	2	-	27x5	0
Batch Norm.	-	-	-	27x5	10
Fc 1	-	-	Softmax	2x1	272
Total Number of Parameters on Edge					607
Conv 2	32	1	ReLU	27x15	2415
Pooling 2	2	2	-	14x15	0
Batch Norm	-	-	-	14x15	30
Grouped Conv	10	1	-	14x75	825
Pointwise Conv	1	1	Relu	14x5	380
Pooling 3	2	2	-	7x5	0
Batch Norm	-	-	-	7x5	10
Dropout	-	-	-	7x5	0
Fc 2	-	-	Softmax	4x1	144
Total Number of Parameters on Fog/Cloud					3804

and mean of the incoming beat, respectively. If the correlation coefficient is lower than 0.2, the normal beat label is changed to abnormal and forwarded to the next layer running in the fog/cloud for further classification.

D. Distributed Multioutput CNN

Our distributed multioutput CNN consists of three convolutions and two output blocks. The first and last convolution blocks are followed by an output block. Fig. 6 shows the layout of the designed CNN. The first and third convolution blocks consist of one convolution layer passed through ReLU activation, one max-pooling layer, and one batch normalization layer. The third convolution block is composed of depthwise and pointwise convolutions. The depthwise convolution block, which connects to each feature map, learns frequency-specific filters. The pointwise convolution block is placed after that to mix the feature maps. Also, closer inspection of the Table I shows that the kernel sizes of designed CNN vary from 64 to 1, which enables the model to learn different features from the heartbeats. Since the shorter filters can cover limited samples, they are used to extract more temporary and rapid oscillatory changes in the ECG, and the longer filters are used to extract more long-term features such as abnormalities in the T-wave. Before the second output block, a dropout value of 0.2 is used

to prevent overfitting during training. Each of the output blocks consists of one fully connected layer passed through the softmax activation. The details of the architecture parameters for each of the layers are given in Table I.

We design the distributed multioutput CNN considering the current limitations and resource constraints of the state-of-the-art works. It is known that the number of parameters and computational operations increases with the number of layers which leads to the consumption of more energy and memory resources. To solve these problems, we have distributed the CNN layers between edge and fog/cloud so that the introduced additional energy consumption and memory of the CNN to the edge are decreased. This fact may be seen from Table I. The total number of parameters before the second convolution block is 607, whereas CNN has 3804 more parameters after that. This clearly shows that by distributing the CNN over nodes, the memory requirement of the edge node can be decreased by 87%.

Another advantage of this proposed solution is that we decreased the communication energy requirements by $\sim 1000 \times$ using the first output block. The first output block is designed to classify beats as normal or abnormal during run-time using the extracted features from the first convolution block. By adding this block, normal and abnormal beats can be distinguished without the need to invoke the other two convolutional blocks. We have observed that this classification, which uses the first convolution block, can reach up to 95% accuracy. Therefore, further processing of those regular beats would be redundant. Through avoiding this redundant operation, the system's energy consumption and inference time are decreased from both communication and computation.

IV. EXPERIMENTAL SETUP

A. Training Distributed Multioutput CNN Classifier

We use data from MIT-BIH Arrhythmia Data Set [22], which contains 48 half-hours of two-channel ambulatory ECG recordings, digitized at 360 samples per second, obtained from 47 subjects, and MIT-BIH Supraventricular Arrhythmia Database [31] which includes 78 half-hour ECG recordings with digitized at 120 Hz. We combined these two data sets to increase the number of abnormal beats. For both data sets, only the lead-II ECG signals are used for experiments. For a fair comparison with published results, we follow the evaluation settings that was most frequent in the state-of-the-art-works. We have excluded the four paced records (102, 104, 107, and 217) from the MIT-BIH data set [22].

ECG beats in 22 recordings from the MIT-BIH data set are included to training set. Additionally, beats from MIT-BIH Supraventricular Arrhythmia Database [31] are added to the training set. In the training set, 80% of ECG beats are used for training, and 20% of ECG beats are used for validation (as shown in Fig. 7). Since the sampling rates of the two data sets are different, we have resampled them to 130 Hz. The 44 records (22 records from the training set and 22 records which the model has never seen before) from the MIT-BIH data set [22] are used as test data. The network was trained with Glorot initialization of the weights [37], we used the Adam optimizer [33] with the default parameters $\beta_1 = 0.9$

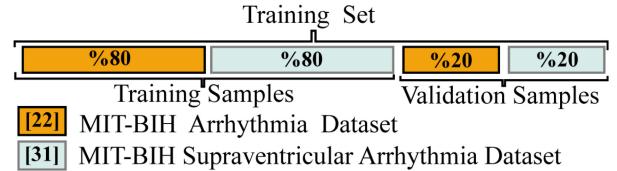


Fig. 7. Ratio of training and validation samples in training set.

and $\beta_2 = 0.999$. The learning rate is initialized to 0.001 and reduced by a factor of 10 when the validation accuracy stopped improving for 15 consecutive epochs. The training continues until 100 successive epochs without validation performance improvements with a maximum of 500 epochs. The best model is chosen as the highest accuracy rate on the validation data. After training the complete CNN architecture, the weights of the first convolution block are obtained to train the fully connected layer at the first output block. The same training data is given as input to the first convolution block again. Then, the output of the first batch normalization layer is fed to the fully connected layer instead of the second convolution block to classify beats as normal and abnormal. During backpropagation, only the weights and biases of the fully connected layer are calculated from the gradient of classification loss. In other words, while training the first output block, the parameters of the first convolutional block are not changed.

B. Target Wearable Device

Our work is designed for low-power and low-memory wearable devices. For example, SmartCardiaINYU [24] device is equipped with an ultra-low-power 32-bit microcontroller STM32L151 containing an ARM Cortex-M3 with a maximum clock rate of 32 MHz. It has a 48-kb RAM, 384-kb Flash, and a standard 710-mAh battery. The device captures ECG signals using a single lead ECG sensor. We have tested our work on a *BLE* standalone module similar to [24] with a 32-bit ARM Cortex-M4 core with 40-MHz maximum operating frequency and 64 kb of RAM. Our profiling showed that algorithms peak memory usage do not exceed 30 kb of RAM.

V. RESULTS

A. Performance of Noise/Artifact Detection

First, the MIT-BIH Arrhythmia Data Set is used to evaluate the performance. The recordings include clinically significant arrhythmias, which have quite different waveforms from the normal ECG, so it is essential to classify them as not noisy ECG signal to validate the algorithm. The performance of the proposed method is evaluated using four benchmark metrics, sensitivity (Se), accuracy (Ac), specificity (Sp), and positive predictivity (PPV) which are defined as follows:

$$Se = \frac{TP}{TP + FN} \quad (5)$$

$$Sp = \frac{TN}{TN + FP} \quad (6)$$

$$PPV = \frac{TP}{TP + FP} \quad (7)$$

$$Ac = \frac{TP + TN}{TP + TN + FN + FP} \quad (8)$$

TABLE II
PERFORMANCE OF THE ALGORITHM IN MIT-BIH FOR DETECTION OF NOISE/ARTIFACT

Segment Type	Total # of Segments	True Positives	False Negatives	Sp %	Ac %	Se %	PPV %
Clean	5000	4970	30	99.4	99.3	99.2	99.5
Noisy	5000	4959	41	99.2	99.3	99.4	99.4

TABLE III
PERFORMANCE COMPARISON OF RELATED WORKS IN PICC AND MIT-BIH DATA SET FOR DETECTION OF NOISE/ARTIFACT

Works	Sp	Ac	Se	Dataset	Methods	Features
[26]	94	-	99.74	PICC MIT-BIH	Rule Based	Time & Freq. Domain
	99.4	-	98.5			
[27]	96.5	97.1	97.7	PICC MIT-BIH	SVM	Time & Freq. Domain
	97.8	97.8	97.7			
Ours	96.3	96	94.5	PICC MIT-BIH	Rule Based	Time Domain
	99.4	98.3	99.2			

TABLE IV
PERFORMANCE EVALUATION OF THE ALGORITHM IN MERGED MIT-BIH FOR DETECTION OF NORMAL/ABNORMAL BEATS

Labels	Total # Beats	True Classified	False Classified	Beat Type	Se %	Sp %	Acc %
Abnormal	18000	17929	71	S,V,F	99.6	97	98.5
Normal	15000	14545	455	N	97	99.6	

where TP, TN, FP, and FN refer to True Positive, True Negative, False Positive, and False Negative, respectively.

A total of 10 000 segments are obtained from the MIT-BIH data set. As shown in Table II, the algorithm's accuracy can reach 99.3% by wrongly classifying only 71 segments in the MIT-BIH data set amongst the 10 000 segments.

The PICC 2011 challenge [25] is used as a second data set to validate our algorithm and compare it with the other related works. In the PICC, the ECG signals are standard 12-lead, and the leads are recorded simultaneously for 10 s; each lead is sampled at 500 Hz.

The proposed solution to detect the noise and artifacts in the ECG signals outperforms the other related works in the MIT-BIH data set (see Table III). This advantage of the algorithm is crucial since any other method that classifies arrhythmias as noisy would lead to a rejection of the data. Another advantage is that our algorithm does not require computationally heavy features to classify.

B. Performance of Normal/Abnormal Beat Classification

The performance of this layer is evaluated using the AAMI instructions for beat type classification. However, unclassified beats (Q) are excluded from the abnormal beat types since they are very rare in the data set. As shown in Table IV, this layer's classification accuracy is 98.5% with a 99.6% sensitivity for abnormal beats.

The goal in this layer is to achieve high sensitivity for abnormal beats classification because if a beat is classified as normal, it would not be transmitted to the fog/cloud node. Table IV shows that amongst the 18 000 abnormal beats, only 71 of them are misclassified, which indicates a very high sensitivity for the abnormal beat classification.

TABLE V
PERFORMANCE COMPARISON OF RELATED WORKS WITH OURS FOR ABNORMAL BEAT DETECTION

Works	Methods	Se	PPV	Acc
[7]	SVM	-	-	96
[32]	Tree	96.4	92	-
Ours	CNN & Rule based	97	99.6	98.5

Table V compares the performance of related works with the proposed solution on the MIT-BIH data set. The proposed algorithm outperforms others in terms of classification performance. Moreover, these methods use feature-based classifiers, such as SVM and tree, so they need to extract different features from the ECG signal. For example, Venkatesan *et al.* [7] extract 14 different time-frequency domain features from ECG, requiring additional memory and a computational overhead which does not suit the edge device. We tried to evaluate the memory and energy consumption of the state-of-the-art works in our target wearable device, which is explained in Section IV-B. However, the target device's memory overflowed more than $1.5 \times$ of its maximum memory for all algorithms [7], [32] due to huge memory requirements for the feature extraction and classification. On the other hand, our proposed solution calculates the correlation between template and input beats, checks the heart rate variability; no additional features are extracted from a beat specific to that layer. Moreover, unlike the related works, we do not use a feature-based classifier since the first convolution block is used to classify the beat types (N, S, V, F) for complete CNN architecture. The designed first convolution and output blocks consist of 607 parameters that allow us to run that algorithm on edge.

C. Performance of CNN Classifier

The overall performance is evaluated according to the second output block, which classifies beats into Normal, SVEB (supraventricular ectopic heartbeats), ventricular ectopic heartbeats (VEB), and Fusion beat. To make a fair comparison with other works, the performance of *Normal/Abnormal Beat Classifier* is also considered. So, if a beat is misclassified as normal in that layer, it is not transmitted to further CNN layers. The classification performance (Acc, Sen, Spe, Ppr) of VEB and SVEB are also investigated to be consistent with related works and given in Fig. 8.

Fig. 8 shows that the proposed algorithm's performance is the highest for *PPV* of SVEB; and for the other metrics, it is 0.4%–1% below from the best one. Also, closer inspection of the table shows that the performance of [35] reaches more than 0.999 for three different metrics which is practically hard to outperform. However, our algorithm's worst performance is 97.8% whereas it is 80.2% and 88% for [34] and [36], respectively. Since the number of normal beats in MIT-BIH is approximately ten times that of abnormal data, it is easy for models to achieve high accuracy. However, it is harder to achieve high performance on all the metrics. Also, the other models, in general, are not suitable for the edge devices as

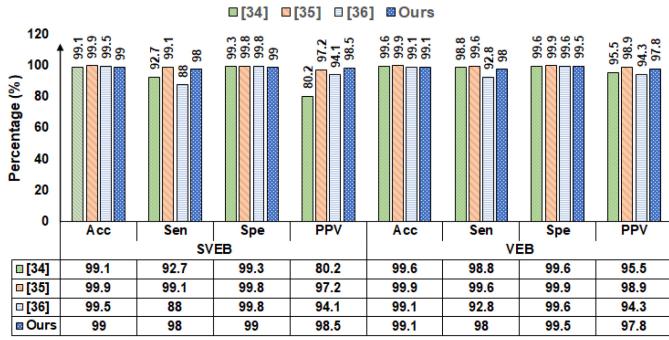


Fig. 8. Performance comparison between the proposed methodology and related works.

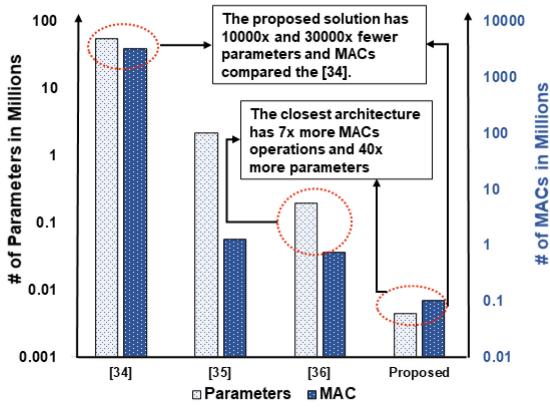


Fig. 9. Comparison of computational complexity and memory with related works.

their architecture has many parameters. To compare models in terms of memory and energy, we estimated the number of parameters and multiply accumulate operations (MACs) of each architecture. Equation (9) is used for the calculation of MACs

$$\text{MACs} = \frac{C_{\text{in}} \times C_{\text{out}} \times K_h \times K_w \times H_{\text{out}} \times W_{\text{out}}}{g} \quad (9)$$

where C_{in} is the number of input channels, C_{out} is the number of output channels, H_{out} and W_{out} are the height and width of the layer's output, respectively. $K_h \times K_w$ is the kernel size of each convolution, and g is the number of groups if there are any. When the MACs are calculated for different works, we observed that the proposed solution has $30,000 \times$ fewer operations compared to [34]. Therefore, for better visualization, Fig. 9 is given in logarithmic form where the two y-axes are in millions. Fig. 9 shows that the proposed solution is much more energy and memory-efficient compared to other existing work. Even the closest CNN architecture performs $7 \times$ more MAC operations and requires $40 \times$ more memory compared to ours.

The proposed CNN has very few parameters and MACs compared to related works because of two reasons. First, in the third convolution block of the proposed CNN, the grouped and pointwise convolutions are performed instead of 1-D convolutions to mix the high-level features optimally. These convolution operations are also more efficient compared to the 1-D convolutions since the number of MACs are decreased

TABLE VI
MEMORY AND ENERGY CONSUMPTION ON BLUE GECKO

Layers	Operation	Exe. Time (ms)	Average Power (mW)	Energy (μJ)	Compatible RAM
Noise/Motion Artifact	Feature Extraction	13.5	14.3	193.05	≥ 32 KB
Normal/Abnormal beat	Template Check & HRV	1.8	23	41.4	≥ 8 KB
CNN	First Convolution Block	20	14.1	282	≥ 32 KB
	First Output Block	1.2	7.5	9	

with increasing groups in convolutions. Second, as we applied a bandpass filter with cut-off frequencies $f_1 = 1$ Hz and $f_2 = 50$ Hz at the beginning of the process, the ECG signal's sampling rate, which directly affects the input size of CNN, is downsampled to 130 Hz without losing any information.

D. Memory and Energy Consumption Evaluation

We evaluate the memory footprint and energy consumption of our proposed methodology on the target device mentioned in Section IV-B. Table VI shows the execution time, energy consumption, and required memory for each layer that runs on the edge device. When we evaluate the power and execution time of each layer, we perform multiple experiments to take the average of them. In the end, we observe $\pm 1\%$ mW and $\pm 0.5\%$ ms deviation from the average of all trials. For example, we observe a maximum 20.12-ms execution time and 14.13-mW average power consumption for the First Convolutional Block. The model is implemented and deployed to the target device using MATLAB (MATLAB and Coder Toolbox Release R2020b, The MathWorks, Inc, USA).

The overall execution time for a heartbeat takes 36 ms in the edge device with 55-mW power consumption. Also, our proposed methodology is compatible with any devices with a minimum RAM of 32 kb. As a result, our methodology guarantees high classification performance while maintaining the low-power wearable devices requirements of being resource-efficient in terms of energy and memory.

VI. DISCUSSION AND FUTURE WORK

In this article, we present a novel and energy-efficient methodology that runs on a hybrid edge–fog/cloud architecture for continuously monitoring the heart at low-power wearable devices. To evaluate our methodology's performance, we compare our approach with several state-of-the-art methods that evaluate their classification results on the same data sets. We show that our proposed methodology reaches or outperforms the current state-of-the-art works in terms of classification performance for three different tasks (noise/artifact detection, normal/abnormal beat detection, and abnormal beat classification) while being energy and memory efficient. However, despite these promising results, questions remain about whether the proposed approach's performance is excellent. Therefore, it is important to evaluate the limitations of our methodology.

First, in our proposed methodology, the abnormal beat detection and classification layers heavily depend on the R-peak detection performance. It is observed that when the detected R-peaks are wrong, the classification performance decreases severely due to wrong segmentation and HRV calculation. Second, even though the MIT-BIH data set is widely used in literature, most state-of-the-art works and our proposed methodology focus on identifying small numbers of cardiac abnormalities (VEB, SVEB) that do not represent the complexity and difficulty of ECG interpretation. Therefore, we believe that there is abundant room for further progress in beat classification in wearable devices. For example, this article showed that the HRV is a helpful feature to classify beats as normal or abnormal. In future investigations, it might be helpful to use different machine learning structures such as neural graph learning to integrate HRV features into an end-to-end model. Since our proposed methodology is a distributed neural architecture between nodes of the network, a further study with more focus on federated learning can be employed to increase performance while preserving privacy [38]. Also, the performance of the transmission decision unit (normal/abnormal beat classification) can be further studied using a more comprehensive ECG data set with different arrhythmias and abnormalities.

VII. CONCLUSION

This article proposes a methodology for real-time continuous heart monitoring using distributed multioutput CNN. The neural network layers are distributed between edge–fog/cloud so as to save energy and time while reducing the communication channel usage and server load. Moreover, the proposed methodology requires $40\times$ less memory compared to state-of-the-art works while maintaining high accuracy. To the best of our knowledge, our methodology achieves the best performance on heartbeat classification while being $7\times$ more energy efficient for devices with a minimum of 32 kb of RAM.

ACKNOWLEDGMENT

Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

REFERENCES

- [1] H. Wang *et al.*, “Global, regional, and national life expectancy, all-cause mortality, and cause-specific mortality for 249 causes of death, 1980–2015: A systematic analysis for the Global Burden of Disease Study 2015,” *Lancet*, vol. 388, no. 10053, pp. 1459–1544, Oct. 2016, doi: [10.1016/S0140-6736\(16\)31012-1](https://doi.org/10.1016/S0140-6736(16)31012-1).
- [2] E. J. Benjamin *et al.*, “Heart disease and stroke statistics-2018 update: A report from the American heart association,” *Circulation*, vol. 137, pp. e67–e492, Mar. 2018.
- [3] D. Jabaoudon, J. Sztajzel, K. Sievert, T. Landis, and R. Sztajzel, “Usefulness of ambulatory 7-day ECG monitoring for the detection of atrial fibrillation and flutter after acute stroke and transient ischemic attack,” *Stroke*, vol. 35, pp. 1647–1651, Jul. 2004.
- [4] A. H. Ribeiro *et al.*, “Automatic diagnosis of the 12-lead ECG using a deep neural network,” *Nat. Commun.*, vol. 11, no. 1, p. 1760, Apr. 2020.
- [5] S. B. Baker, W. Xiang, and I. Atkinson, “Internet of Things for smart healthcare: Technologies, challenges, and opportunities,” *IEEE Access*, vol. 5, pp. 26521–26544, 2017.
- [6] A. Y. Hannun *et al.*, “Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network,” *Nat. Med.*, vol. 25, no. 1, pp. 65–69, Jan. 2019.
- [7] C. Venkatesan, P. Karthigaikumar, A. Paul, S. Satheeskumaran, and R. Kumar, “ECG signal preprocessing and SVM classifier-based abnormality detection in remote healthcare applications,” *IEEE Access*, vol. 6, pp. 9767–9773, 2018.
- [8] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, “Edge computing: Vision and challenges,” *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.
- [9] X. Tang, Z. Ma, Q. Hu, and W. Tang, “A Real-time arrhythmia heartbeats classification algorithm using parallel delta modulations and rotated linear-kernel support vector machines,” *IEEE Trans. Biomed. Eng.*, vol. 67, no. 4, pp. 978–986, Apr. 2020.
- [10] J. M. Bumgarner *et al.*, “Smartwatch algorithm for automated detection of atrial fibrillation,” *J. Amer. College Cardiol.*, vol. 71, no. 21, pp. 2381–2388, 2018.
- [11] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, “Mobile edge computing: A survey,” *IEEE Internet Things J.*, vol. 5, no. 1, pp. 450–465, Feb. 2018.
- [12] A. N. Nowbar, M. Gitto, J. P. Howard, D. P. Francis, and R. AlLamee, “Mortality from ischemic heart disease analysis of data from the world health organization and coronary artery disease risk factors from NCD risk factor collaboration,” *Circ. Cardiovasc. Qual. Outcomes*, vol. 12, no. 6, Jun. 2019, Art. no. e005375.
- [13] H. C. McGill, C. A. McMahan, and S. S. Gidding, “Preventing heart disease in the 21st century: Implications of the pathobiological determinants of atherosclerosis in youth (PDAY) study,” *Circulation*, vol. 117, no. 9, pp. 1216–1227, 2008.
- [14] J. Pan and J. McElhannon, “Future edge cloud and edge computing for Internet of Things applications,” *IEEE Internet Things J.*, vol. 5, no. 1, pp. 439–449, Feb. 2018.
- [15] Y. Liu, M. Peng, G. Shou, Y. Chen, and S. Chen, “Toward edge intelligence: Multiaccess edge computing for 5G and Internet of Things,” *IEEE Internet Things J.*, vol. 7, no. 8, pp. 6722–6747, Aug. 2020.
- [16] N. Rashid, M. Dautta, P. Tseng, and M. A. A. Faruque, “HEAR: Fog-enabled energy aware online human eating activity recognition,” *IEEE Internet Things J.*, vol. 8, no. 2, pp. 860–868, Jan. 2021, doi: [10.1109/JIOT.2020.3008842](https://doi.org/10.1109/JIOT.2020.3008842).
- [17] F. Meneghelli, M. Calore, D. Zucchetto, M. Polese, and A. Zanella, “IoT: Internet of Threats? A survey of practical security vulnerabilities in real IoT devices,” *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8182–8201, Oct. 2019.
- [18] S. Ramesh and M. Govindarasu, “An efficient framework for privacy preserving computations on encrypted IoT data,” *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8700–8708, Sep. 2020.
- [19] S. J. Lee, J. Kim, and M. Lee, “A real-time ECG data compression and transmission algorithm for an e-health device,” *IEEE Trans. Biomed. Eng.*, vol. 58, no. 9, pp. 2448–2455, Sep. 2011.
- [20] P. Huang, L. Guo, M. Li, and Y. Fang, “Practical privacy-preserving ECG-based authentication for IoT-based healthcare,” *IEEE Internet Things J.*, vol. 6, no. 5, pp. 9200–9210, Oct. 2019.
- [21] A. L. Goldberger *et al.*, “PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals,” *Circulation*, vol. 101, no. 23, pp. e215–e220, Jun. 2000.
- [22] 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/Jun. 2001.
- [23] J. Huang, F. Qian, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck, “A close examination of performance and power characteristics of 4G LTE networks,” *Proc. ACM MobiSys*, Jun. 2012, pp. 225–238.
- [24] S. Murali, F. Rincon, and D. Atienza, “A wearable device for physical and emotional health monitoring,” in *Proc. Comput. Cardiol. Conf. (CinC)*, 2015, pp. 121–124.
- [25] I. Silva, G. B. Moody, and L. Celi, “Improving the quality of ECGs collected using mobile phones: The PhysioNet/computing in cardiology challenge 2011,” *Comput. Cardiol.*, vol. 38, no. 1, pp. 273–276, 2011.
- [26] U. Satija, B. Ramkumar, and M. S. Manikandan, “Real-time signal quality-aware ECG telemetry system for IoT-based health care monitoring,” *IEEE Internet Things J.*, vol. 4, no. 3, pp. 815–823, Jun. 2017.
- [27] G. D. Clifford, J. Behar, Q. Li, and I. Rezek, “Signal quality indices and data fusion for determining acceptability of electrocardiograms collected in noisy ambulatory environments,” *Phys. Meas.*, vol. 33, no. 9, pp. 1419–1433, Sep. 2012.
- [28] C. Liu *et al.*, “Signal quality assessment and lightweight QRS detection for wearable ECG SmartVest system,” *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1363–1374, Apr. 2019

- [29] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," *IEEE Trans. Biomed. Eng.*, vol. BME-32, no. 3, pp. 230–236, Mar. 1985.
- [30] *Recommended Practice for Testing and Reporting Performance Results of Ventricular Arrhythmia Detection Algorithms*, AAMI ECAR, Assoc. Adv. Med. Instrum., Arlington, VA, USA, 1987
- [31] S. D. Greenwald, "Improved detection and classification of arrhythmias in noise-corrupted electrocardiograms using contextual information," Ph.D. dissertation, Massachusetts Inst. Technol. Divis. Health Sci. Technol., Harvard Univ., Cambridge, MA, USA, 1990.
- [32] D. Pandit, L. Zhang, N. Aslam, C. Liu, and S. Chattopadhyay, "Improved abnormality detection from raw ECG signals using feature enhancement," in *Proc. 12th Int. Conf. Nat. Comput. Fuzzy Syst. Knowl. Disc.*, 2016, pp. 1402–1406.
- [33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
- [34] P. Xie *et al.*, "Bidirectional recurrent neural network and convolutional neural network (BiRCNN) for ECG beat classification," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2018, pp. 2555–2558.
- [35] Y. Xia and Y. Xie, "A novel wearable electrocardiogram classification system using convolutional neural networks and active learning," *IEEE Access*, vol. 7, pp. 7989–8001, 2019.
- [36] N. Wang, J. Zhou, G. Dai, J. Huang, and Y. Xie, "Energy-efficient intelligent ECG monitoring for wearable devices," *IEEE Trans. Biomed. Circuits Syst.*, vol. 13, no. 5, pp. 1112–1121, Oct. 2019.
- [37] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Stat. Mach. Learn. Res.*, 2010, pp. 249–256.
- [38] S. Deng, H. Zhao, W. Fang, J. Yin, S. Dustdar, and A. Y. Zomaya, "Edge intelligence: The confluence of edge computing and artificial intelligence," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7457–7469, Aug. 2020, doi: [10.1109/JIOT.2020.2984887](https://doi.org/10.1109/JIOT.2020.2984887).

Berken Utku Demirel (Student Member, IEEE) received the bachelor's degree in electrical and electronics engineering from Middle East Technical University, Ankara, Turkey, in 2020. He is currently pursuing the Ph.D. degree in computer engineering with the University of California at Irvine, Irvine, CA, USA.

His research is focused on bio-signal processing, energy-efficient embedded systems and cyber-physical systems with special interest on mobile healthcare, wearable devices, and edge machine learning.

Islam Abdelsalam Bayoumy received the bachelor's degree in computer engineering from Al-Azhar University, Cairo, Egypt, in 2020. He is currently pursuing the M.S. degree in computer engineering with the University of California at Irvine, Irvine, CA, USA.

His research is focused on embedded systems.

Mohammad Abdullah Al Faruque (Senior Member, IEEE) received the Ph.D. degree in computer science from Karlsruhe Institute of Technology, Karlsruhe, Germany, in 2009.

He is currently an Associate Professor with the University of California at Irvine (UCI), Irvine, CA, USA. He is directing the Embedded and Cyber-Physical Systems Lab, UCI. His current research is focused on the system-level design of embedded and cyber-physical systems.

Dr. Al Faruque has been the IEEE CEDA Distinguished Lecturer since 2022 and the ACM Senior Member.