IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 11, 2020, accepted May 27, 2020, date of publication May 29, 2020, date of current version June 10, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2998608

A Lossless Electrocardiogram Compression System Based on Dual-Mode Prediction and Error Modeling

MENGHAN JIA[1], FEITENG LI[1], YU PU[2], (Member, IEEE), AND ZHIJIAN CHEN[1]
[1]Institute of VLSI Design, Zhejiang University, Hangzhou 310027, China
[2]Alibaba DAMO Academy, Sunnyvale, CA 94085, USA

Corresponding author: Zhijian Chen (chenzj@vlsi.zju.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 61801425.

ABSTRACT Long-term electrocardiogram (ECG) monitoring requires high-ratio lossless compression techniques to reduce data transmission energy and data storage capacity. In this paper, we have proposed a high-ratio ECG compression system with low computational complexity. Firstly, as the morphologies of the ECG change over time, we divide the signal of each heartbeat cycle into two regions. To achieve high prediction accuracy, a 1st order linear predictor and a combination of the template predictor and 3rd order linear predictor are applied in the two regions respectively. Secondly, we introduce a context-based error modeling module to the system, which cancels the statistical bias of the prediction algorithm and further improves the prediction accuracy. Thirdly, we modify the Golomb-Rice encoding algorithm to adaptively encode the prediction errors, while preserving a code space for packaging the information that is necessary for prediction. We evaluate the proposed system by using the MIT-BIH Arrhythmia Database (ARRDB). The experimental results show that with memory requirements as low as 444 to 14556 total variables this system achieves a compression ratio (CR) from 2.975 to 3.040, suggesting that it is highly applicable to both the low-power design and the cloud.

INDEX TERMS Electrocardiogram, lossless compression, error modeling.

I. INTRODUCTION

Electrocardiogram (ECG) indicates the electrical activity of the heart and it is the most commonly used method to monitor the heartbeat. With wireless and wearable healthcare devices, long-term ECG data can be recorded continuously for monitoring and diagnosis. However, collecting such a large amount of data requires excessive transmission energy or storage capacity, which significantly increases the cost of long-term ECG applications [1], [2]. Therefore, an effective and efficient data compression method for ECG signals is required.

ECG compression methods include lossless compression and lossy compression. In lossless systems, the reconstructed ECG can be exactly the same as the original ECG, which is generally more useful for cardiac disease diagnosis. Because the lossy compression discards some morphological information, and it has not been approved by medical bodies in most countries and cannot be used in commercial devices [3]. Therefore, this paper focuses on lossless ECG compression.

A good lossless ECG compression algorithm must achieve a high CR with low computational complexity and a low number of variables, which are closely related to the hardware requirement of the algorithm. Especially for low-power application-specific integrated circuit (ASIC) designs or embedded system designs, low computational complexity and a low number of variables can decrease the chip area, thus reducing the chip cost and power consumption. Conventionally, the lossless ECG compression technique first predicts or transforms the signal to reduce the signal entropy, then entropy encodes the signal to remove redundant bits [4]. To date, various lossless ECG compression methods have been proposed, including algorithms such as data pulse code modulation (DPCM) + Golomb-Rice encoding [5], adaptive linear predictor + modified Huffman encoding [6], [7], adaptive

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

linear predictor + variable-length encoding [8], short term linear predictor + fixed-length encoding [3], and adaptive linear predictor + Golomb-Rice encoding [9]. These algorithms are relatively simple, but they do not utilize sufficient ECG morphologies hence causing poor CRs.

Some other methods achieve high CRs. Among which, Miaou and Chao [10] applied the combination of distortion-constrained codebook replenishment (DCCR), set partitioning in hierarchical tree (SPIHT), and bit-plane coding (BPC) algorithms, but this method contains recursive operation and requires many variables. Zhou [11] applied k-means cluster predictor + Huffman encoding, and Tseng *et al.* [12] applied Takagi-Sugeno fuzzy neural network + arithmetic encoding, but they require multiply-accumulate operation many times when predicting each point. Tsai and Tsai [13] applied adaptive linear predictor + Golomb-Rice encoding, and Rzepka [14] applied selective and multichannel linear predictor + asymmetric numeral systems encoding, but they all require integer division operation hence increasing the computational complexity. Therefore, the methods of [10]–[14] are difficult to be implemented on wearable ECG monitoring devices.

This study proposes a lossless ECG compression algorithm based on dual-mode prediction with context error modeling. The features of the proposed method are:

1) Given the feature that the morphologies of the ECG change over time, we divide the signal of each heartbeat cycle into two regions. A 1st order linear predictor and a combination of the template predictor and 3rd order linear predictor are applied respectively to achieve high prediction accuracy.
2) A context-based error modeling module is added after the predictors to cancel the statistical bias of the prediction errors and further improve the prediction accuracy.
3) A modified Golomb-Rice encoding algorithm is designed for entropy encoding. This algorithm adaptively encodes the prediction error according to the current error amplitudes to improve the CR. Besides, it preserves a code space for packaging the information which is necessary for prediction.
4) By adjusting the size of the templates and contexts, this system can balance the number of variables and compression performance, in this way enabling compression applications from the edge to the cloud.

The remainder of this paper is organized as follows. Section II briefly introduces the database that applies to this research. Section III presents the proposed lossless ECG compression system. The proposed method is evaluated in Section IV. Section V discusses the experimental results. Finally, section VI draws the conclusion.

II. DATABASE

This paper uses the MIT-BIH Arrhythmia Database (ARRDB) to evaluate the proposed lossless ECG compression system. The ARRDB contains 48 excerpts of two-channel ambulatory ECG recordings, which are obtained from 47 subjects studied by the BIH Arrhythmia Laboratory between 1975 and 1979. 23 recordings were chosen at random from a set of 4000 24-hour ambulatory ECG recordings collected from a mixed population of inpatients (about 60%) and outpatients (about 40%) at Boston's Beth Israel Hospital; the remaining 25 recordings were selected from the same set to include less common but clinically significant arrhythmias that would not be well-represented in a small random sample. The recordings were digitized at 360 samples per second per channel with an 11-bit resolution over a 10 mV range, and there are in total 650,000 points in each channel. Two or more cardiologists independently annotated each record; disagreements were resolved to obtain the computer-readable reference annotations for each beat (approximately 110,000 annotations in all) included in the database [15], [16].

III. METHOD

Figure 1 shows the block diagram of the proposed method. In the compression part, this method split each sampled ECG point into the prediction value and error value, i.e.

$$\epsilon\,[n] = x\,[n] - \tilde{x}\,[n] \tag{1}$$

where $x\,[n]$ represents the ECG samples at the n^{th} point, $\tilde{x}\,[n]$ represents the prediction value which is derived from the past samples by using the dual-mode prediction and context-based error modeling methods, and $\epsilon\,[n]$ represents the error value. Then, we package the error codes with the information which is necessary for prediction to form the final bitstream. In the decompression part, we use the same prediction and error modeling methods to get the same prediction values. By adding the error values with the prediction values, the lossless reconstruction ECG can be obtained.

A. DUAL-MODE PREDICTION

The normal ECG comprises a sequence of P, Q, R, S, and T wave. Among them, the Q, R, S waves compose the QRS-complex which is the main spike seen on the ECG, while the waveform out of the QRS-complex is flatter as shown in Fig. 2.

According to the different morphologies of the ECG in different periods, we divide the ECG signal into the QRS regions and the non-QRS regions first for separate prediction. The QRS duration of a normal ECG is between 60-109ms. However, when there are some abnormalities in the ECG, such as the premature ventricular contractions (PVCs), the QRS duration may be $\geq$120ms [17]. To obtain relatively accurate division results for both normal and abnormal ECG, we set the length of the QRS region to 100ms and the R-peak position as the midpoint of the QRS region. So the number of sampling points in the QRS region W_{qrs} is calculated according to

$$W_{qrs} = 0.1 f_s \tag{2}$$

where f_s represents the ECG sampling rate. We apply the R-peak detection algorithm proposed by Ieong *et al.* [2] to locate

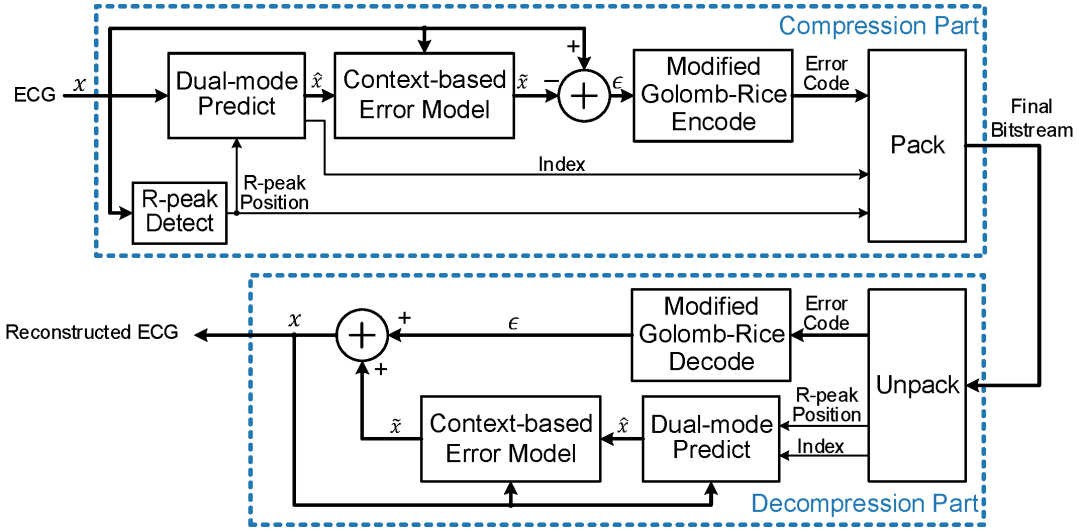


FIGURE 1. Overall block diagram of the proposed lossless ECG compression and decompression system.

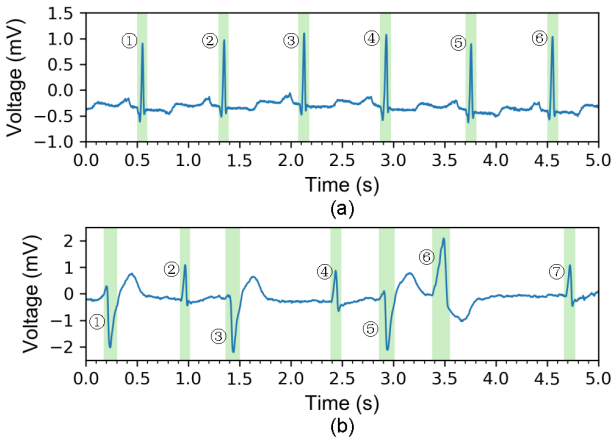


FIGURE 2. (a) A piece of the normal ECG with 6 cycles from the ARRDB recording 100. (b) A piece of the abnormal ECG with 7 cycles from the ARRDB recording 200, including occasional PVCs. Where the green background indicates the QRS-complex.

the QRS region since this algorithm achieves a high R-peak detection accuracy with low computational complexity.

For the flat non-QRS region, the difference between the two adjacent points is small. Therefore, we predict the current signal by using the previous sampled point, which is called 1st order linear prediction, i.e.

$$\hat{x}[n] = x[n-1] \tag{3}$$

where $\hat{x}[n]$ is the prediction value of the n^{th} point, and $x[n-1]$ is the sampled value of the $(n-1)^{\text{th}}$ point.

Koski [4] applied one template to predict the points belonging to the QRS complex since the morphologies such as the directions and the slope values of continuous heartbeats are similar. However, in general, more templates lead to higher prediction performance, because

1) Some ECG signals contain multiple QRS morphologies, as shown in Fig. 2 (b). More templates can cover more QRS morphologies.
2) Even if the morphology of two QRS-complexes is the same macroscopically, there may be some differences in details. For example, the amplitude of the 5th R-peak is smaller than that of the 4th R-peak in Fig. 2(a). More templates can help to reduce prediction errors.

In this system, we introduce multiple templates to predict the signal in the QRS regions. However, it should be noted that more templates will increase the number of variables and computation amount, thus making it difficult to be applied to low-power design. Therefore, it is necessary to use different numbers of templates to balance the memory requirement, computation amount, and CR for different scenarios. For narrative purposes, we assume that there are N_t templates in the system. So, the templates have a total of $W_{qrs} \times N_t$ variables. Since the ECG is vulnerable to the interference of the baseline drift noise, which causes some bias [18], to make the templates not affected by the signal bias, the templates only store the difference between two adjacent points, i.e.

$$v = \Delta x[n] = x[n] - x[n-1] \tag{4}$$

where v is the value that will be saved to the template. Besides, on considering the fluctuation feature of the QRS, we also use the 3rd order linear prediction as a supplementary predictor to predict the data of the QRS region more accurately when 1) there is no template for prediction in the initial stage, or 2) the templates do not contain this QRS morphology. The prediction formula is given in (5).

$$\begin{aligned}\hat{x}[n] &= x[n-1] + \Delta x[n-1] + \Delta(\Delta x[n-1]) \\ &= 3x[n-1] - 3x[n-2] + x[n-3]\end{aligned} \tag{5}$$

There are $N_t + 1$ predictors for the QRS region, including N_t template predictors and a 3rd order linear predictor.

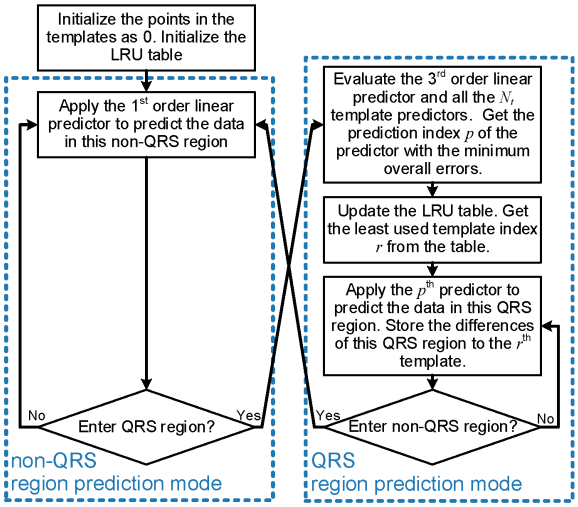


FIGURE 3. Flow chart of the proposed dual-mode prediction algorithm.

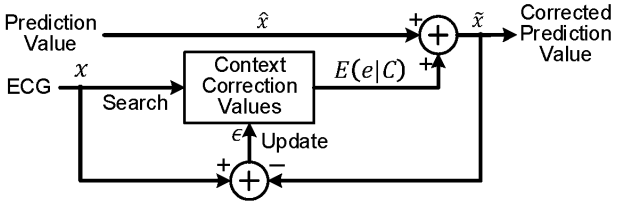


FIGURE 4. Block diagram of the context-based error modeling, where e is the prediction error, $E(e|C)$ is the expectation of the prediction errors, which is also used as the correction value for the current context, $\tilde{x}$ is the corrected prediction value, and ϵ is the prediction error after correction.

The predictor with the minimum total prediction error is used to predict this QRS region, then the difference data of the newest QRS region will be saved as a template to replace the ''Least Recently Used'' (LRU) template. Therefore, a real-time updated LRU table is required to record the order of the used templates. The detailed flow chart of the proposed dual-mode prediction algorithm is shown in Fig. 3. The index of the predictor used for each QRS region will be packed into the final bitstream to use the same predictor when decompressing. To make full use of the code space, the length of the index code B_{idx} is set according to (6).

$$B_{idx} = \lceil log_2 (N_t + 1) \rceil \tag{6}$$

B. CONTEXT-BASED ERROR MODELING

The context-based error modeling technique that captures the statistical information of the prediction errors can be used to improve the CR [19]–[23]. This technique first classifies the prediction errors based on their contexts. Then the expectation of the prediction errors from each category is determined and this value will be added to the prediction value to correct the prediction, as shown in Fig. 4. Compared with the original prediction values, the corrected prediction values are more accurate, thus increasing the CR.

A simple but effective context classification model is based on the binarized differences of past points, i.e.

$$c(x[n]) = \{Q(\Delta x[n - W_c]), \cdots, Q(\Delta x[n-1])\} \tag{7}$$

where W_c is the number of points used for context classification. $c(x[n])$ is the category index of the n^{th} point. $Q(\Delta x[n])$ is the binarized differences of the n^{th} point, and the formula is given in (8).

$$Q(\Delta x[n]) = \begin{cases} 1, & \Delta x[n] \geq 0 \\ 0, & \Delta x[n] < 0 \end{cases} \tag{8}$$

The total number of contexts is 2^{W_c}. In general, more contexts capture more accurate statistical information, thus improving the CR. However, more context also increases the number of variables. So, it is necessary to use different numbers of contexts for different scenarios.

Memon *et al.* [22], Sriraam and Eswaran [23], Chua and Fang [5] also applied the context-based error modeling technique for lossless biosignal compression. They estimate the expectation of each context category by dividing the sum of the prediction errors with the occurrence counter. However, this algorithm has two drawbacks. First, it requires the ''divide'' operation, which is difficult to be implemented on the low-power ASICs or embedded systems. Second, it is sensitive to outliers, thus reducing the accuracy of the expectations. Weinberger *et al.* [24] designed another error modeling algorithm for image compression. It only requires the ''add'' and ''compare'' operations and reduces the influence of the outliers. Therefore, we select this algorithm for error modeling to achieve higher CRs with lower computational complexity. In this algorithm, there are 3 variables in each context category, i.e. *COR*, *CNT*, *RES*. *COR* stores the expectation and also the correction values of the prediction errors; *CNT* stores the occurrence number; and *RES* stores the sum of errors after bias cancellation. By limiting the range of *RES*, this algorithm reduces the influence of outliers. *COR* is updated by comparing *CNT* and *RES* to make the mean value of the corrected prediction errors close to ''0''. The specific pseudo-code of this error modeling algorithm is shown in Fig 5.

C. ENCODING

Figure 6 shows the prediction errors after dual-mode prediction and context-based error modeling. It can be seen that the prediction errors fluctuates around ''0'', and the occurrence probability of small values is much higher than large values. Golomb-Rice code is quite useful to encode such prediction errors as it has optimal prefix code for this distribution [25] and it has low computational complexity.

The Golomb-Rice encoding algorithm requires mapping the integer prediction errors into the non-negative integer first, i.e.

$$M[n] = \begin{cases} 2\epsilon[n], & \epsilon[n] \geq 0 \\ -2\epsilon[n] - 1, & \epsilon[n] < 0 \end{cases} \tag{9}$$

Initialization:
$COR\left[0\cdots 2^{W_c}-1\right]=0$; //context correction value
$CNT\left[0\cdots 2^{W_c}-1\right]=0$; //occurrence counter
$RES\left[0\cdots 2^{W_c}-1\right]=0$; //sum of residuals after bias cancellation
For each sample $x[n]$:
1. $i_c=c(x[n])$; //get context index
2. $\tilde{x}[n]=\hat{x}[n]+COR[i_c]$; //bias cancellation
3. $\epsilon[n]=x[n]-\tilde{x}[n]$; //get the residual after bias cancellation
4. $CNT[i_c]=CNT[i_c]+1$; //update occurrence counter
5. $RES[i_c]=RES[i_c]+\epsilon[n]$; //accumulate prediction residual
6. //update correction value

```
if(RES[i_c] ≤ −CNT[i_c]){
  COR[i_c] = COR[i_c] − 1;
  RES[i_c] = RES[i_c] + CNT[i_c];
  if(RES[i_c] ≤ −CNT[i_c]) RES[i_c] = −CNT[i_c] + 1;
}else if(RES[i_c] > 0){
  COR[i_c] = COR[i_c] + 1;
  RES[i_c] = RES[i_c] − CNT[i_c];
  if(RES[i_c] > 0) RES[i_c] = 0;
}
```

FIGURE 5. Pseudo-code of the error modeling algorithm used in the proposed ECG compression system.

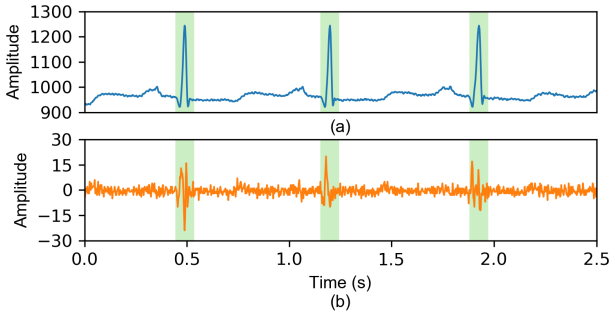


FIGURE 6. (a) An example of the typical ECG from the recording 100 of the ARRDB. (b) The final prediction errors ϵ of (a), where the prediction errors of neighbor points distribute in similar ranges.

where $M[n]$ is the error after mapping. $M[n]$ is then divided by another integer k to obtain the quotient part and the remainder part. The unary and binary codes are used to encode the quotient and the remainder, and a bit "0" is inserted between the unary code and binary code to obtain the final code, as follows

$$\begin{aligned}&\text{golomb\_rice\_code}\left(M[n]\right)\\&\quad=\text{group}\Big(\\&\qquad\text{unary\_code}\left(\left\lfloor\frac{M[n]}{2^k}\right\rfloor\right),\\&\qquad 0,\\&\qquad\text{binary\_code}\left(M[n]\bmod 2^k\right)\\&\qquad\Big)\end{aligned}\tag{10}$$

where $\left\lfloor\frac{M[n]}{2^k}\right\rfloor$ can be calculated by using the "shift right" operation in the ASIC or embedded system.

The efficiency of the code is sensitive to k value. Specifically, when k is set to $\lfloor\log_2 M[n]\rfloor$, the code length of $M[n]$ is the shortest. Considering that the prediction errors of neighbor points distribute in similar ranges, as shown in Fig. 6, we propose a k value prediction mechanism. First, we add a variable t to predict the value of $M[n]$. Actually, since only integer operations can be performed in low-power devices, we set t to 4 times the predicted value to improve the precision. Second, k is calculated by (11) to encode $M[n]$. For low power ASICs or embedded systems, $\lfloor\log_2\frac{t}{4}\rfloor$ can be calculated by searching the most significant bit of t. When $k=0$, the code length increases significantly with the increase of $M[n]$, thus causing a great penalty when $\frac{t}{4}$ is lower than $M[n]$. Therefore, we force $k\geq 1$ to reduce the penalty. Third, after coding, t is updated by (12) to predict the next data. This update method gives a higher weight to the data closer to the uncoded data and improves the prediction accuracy. The initial values of t and k are set to 64 and 4 to avoid excessive code length due to the large value of the first uncoded data.

$$k=\max\left(\left\lfloor\log_2\frac{t}{4}\right\rfloor,1\right)\tag{11}$$

$$t=\left\lfloor\frac{3t}{4}\right\rfloor+M[n]\tag{12}$$

Besides, the Golomb-Rice code has covered all the coding space and made it impossible to add the R-peak position information to the code. Therefore, we add a bit "1" to all the unary codes with quotient ≥ 8 to reserve the code "111111110" as the R-peak indication code. Table. 1 shows several comparison examples between the Golomb-Rice code and the modified Golomb-Rice code. This encoding method makes the QRS code length relatively short while only increasing the code length of data with large quotient (which rarely appears) by 1 bit.

D. PACKAGING

In the packaging process, all the information needed for the decoder to reconstruct the ECG will be packaged into the final bitstream. It should be noted that the sampling frequency and resolution of the ECG sensor, the template number and context width of the proposed compression system, and the initial values of all the variables should already be known to the decoder. Therefore, the final bitstream doesn't contain this information. Figure 7 shows the packaging format. First, we package the first three signals by using the original binary codes so that the decoder can predict the following points according to the known points. After that, for each heartbeat cycle, the data is packaged in the following order to compress and decompress the signal in real-time:

1) Package the modified Golomb-Rice codes of the current non-QRS region.
2) When entering the QRS region, the R-peak indication code "111111110" and the prediction index for current QRS region are first packaged to inform the decoder.
3) Package the modified Golomb-Rice codes of the current QRS-region.

TABLE 1. Several examples of comparison between the Golomb-Rice code and the modified Golomb-Rice code.

$\epsilon[n]$	$M[n]$	k	2^k	$\lfloor \frac{M[n]}{2^k} \rfloor$	$M[n] \bmod 2^k$	Golomb-Rice encoding		modified Golomb-Rice encoding	
						code	length	code	length
0	0	1	2	0	0	0 0	2	0 0	2
-9	17	1	2	8	1	11111111 0 1	10	111111111 0 1	11
2	4	2	4	1	0	1 0 00	4	1 0 00	4
19	38	2	4	9	2	111111111 0 10	12	1111111111 0 10	13
28	56	3	8	7	0	1111111 0 000	11	1111111 0 000	11
-43	85	3	8	10	5	1111111111 0 101	14	11111111111 0 101	15

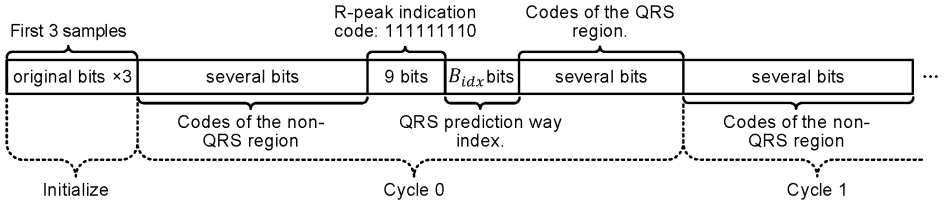


FIGURE 7. Packaging format for the ECG.

Table. 2 shows an example of using the proposed packaging format to package the ECG. The sampling frequency and resolution of the example are 360Hz and 11-bit respectively, which are the same as those of the ARRDB. The N_t value for the compression system used in Table. 2 is set to 7. The 51st to 86th sampling points in this example belongs to the QRS region of cycle 0.

IV. EVALUATION

A. EVALUATION CRITERIA

We take all the recordings from the ARRDB as our test set. CR is used as the evaluation criteria for each ECG recording, which is calculated according to

$$CR = \frac{B_o}{B_c} \tag{13}$$

where B_o is the total bit number of the original ECG recording,

$$B_o = N_p \times R \tag{14}$$

N_p indicates the number of sampling points in the recording, and R indicates the resolution. For ARRDB, $N_p = 650,000$ and $R = 11$, So $B_o = 7,150,000$.

B_c is the total bit number after compressing. In this paper, B_c is calculated according to

$$B_c = B_{init} + N_{hb} \times \left(B_{qrs} + B_{idx}\right) + B_{code} \tag{15}$$

where B_{init} is the bit number for initialization, N_{hb} is the heartbeat number, B_{qrs} is the R-peak indication code length, and B_{code} is the overall bit length for modified Golomb-Rice code. For ARRDB, $B_{init} = 33$, $B_{qrs} = 9$, and B_{idx} is calculated according to (6). The final CR is obtained by calculating the average CR of all the recordings in the database.

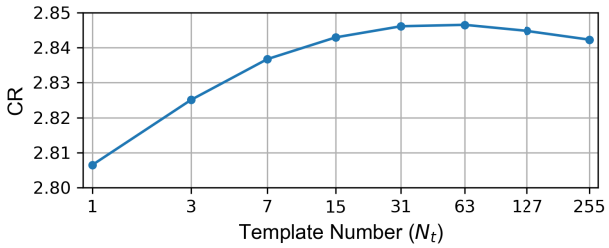


FIGURE 8. Different N_t values versus CRs on the ARRDB.

B. N_t, W_c SELECTION AND COMPRESSION RESULTS

To select N_t, we remove the context-based error modeling module and set N_t to 1, 3, 7, 15, 31, 63, 127, and 255 for alternatives. At this time, the total number of the predictors for the QRS region is exactly an integer power of 2, thus maximizing the utilization of the index code. Figure 8 shows the CRs obtained by using different N_t. It can be seen that the CR improves significantly as N_t increases from 1 to 7, and achieves the maximum value at 63. When N_t continues to increase, CR begins to fall. It is because 1) when the number of templates reaches a certain level, the templates contain enough QRS morphologies that are currently required, 2) using more templates increases B_{idx} value, thus decreasing the CR. Since the low-power ASICs or embedded systems are sensitive to the memory requirement, we select $N_t = 7$ for low-power design. For the cloud ECG compression scenario, we select $N_t = 63$.

To select W_c, we set W_c from 1 to 15 for alternatives and do experiments under the condition of $N_t = 7$ or $N_t = 63$. Figure 9 shows the results. It can be seen that the CR improves significantly as W_c increases from 1 to 6, and achieves the maximum value at 12. As W_c continues to increase, CR begins to fall. It is because that the correction values need to be updated during bias cancellation to converge to the statistical expectations. With the increase of the contexts, the points of each context become insufficient for converging,

TABLE 2. An example of using the proposed packaging format.

Cycle	n	$\epsilon[n]$	$M[n]$	t	k	Code	Description
Initialize	0	-	-	-	-	01111100011	Original binary codes of $x[0]$, $x[1]$, $x[2]$, which are 995, 1000, 997 respectively.
	1	-	-	-	-	01111101000	
	2	-	-	-	-	01111100101	
Cycle 0	3	-2	3	64	4	0 0011	Modified Golomb-Rice codes of the non-QRS region.
	4	0	0	51	3	0 000	
	5	-2	3	38	3	0 011	
	6	1	2	31	2	0 10	
				...			
	49	4	8	15	1	1111 0 0	
	50	-2	3	19	2	11 0 1	
	-	-	-	-	-	11111111 0	R-peak indication code.
	-	-	-	-	-	111	Prediction way index for the QRS region. "111" means 3rd order linear predictor.
	51	6	12	17	2	111 0 00	Modified Golomb-Rice codes of the QRS region.
	52	-1	1	24	2	0 01	
	53	3	6	19	2	1 0 10	
				...			
	85	3	6	47	3	0 110	
	86	-2	3	41	3	0 011	
Cycle 1	87	-3	5	33	3	0 101	Modified Golomb-Rice codes of the non-QRS region.
	88	-3	5	29	2	1 0 01	
	89	-4	7	26	2	1 0 11	
				...			

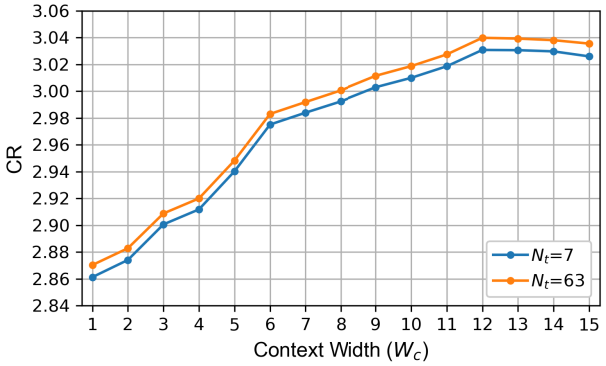


FIGURE 9. Different W_c values versus CR on the ARRDB when N_t is set to 7 or 63.

thus reducing the performance of bias cancellation. On both considering the memory requirement and the CR, we select $W_c = 6$ for low-power design. For the cloud ECG compression scenario, we select $W_c = 12$.

For convenience, we use "S" and "L" to represent the proposed compressing system for low-power design (which has a small number of variables) and the cloud (which has a large number of variables). Table. 3 shows the specific variable numbers and CRs on the ARRDB by using the proposed S system or L system.

C. COMPARISON WITH OTHER METHODS

Table. 4 compares the CR of the proposed system with several other existing methods. The proposed method achieves the CR second only to Miaou's method [10]. However, Miaou's method contains a codebook with a size of 1024×64, meaning that there are as many as 65536 variables in the codebook. Moreover, Miaou's method requires recursive operations, so that the computational complexity and the variable number of Miaou's method are higher than that of the proposed method. We have reproduced Miaou's method and modified the size of the codebook to explore the relationship between the CR and the variable number. Figure 10 shows the comparison of the proposed method and Miaou's method with different variable numbers. It can be seen that the CR of the proposed method is higher than Miaou's method when the variable numbers are similar and less than 16384. References [11]–[14] also achieve relatively high CR. However, the K-means cluster of Zhou's method [11] needs to square each point when matching templates, and the Huffman codebook in Zhou's method contains 2048 items the compress the ECG signal of the ARRDB (which has 11-bit resolution), more than the variable number of the proposed S system; the Takagi-Sugeno Fuzzy Neural Network in [12] contains many multiply-accumulate operations when predicting each point; and both [13] and [14] need integer division operations. So the computational complexity of [11]–[14] are higher than that of the proposed method. References [3], [5]–[9] used simple prediction and entropy encoding algorithms, but the proposed S system achieves significantly higher CR than these methods so that this method can save more transmission power or storage space.

V. DISCUSSION

A. EFFECT OF DIFFERENT R-PEAK DETECTION ALGORITHMS ON CR

In the proposed system, the R-peak detection algorithm used for the QRS region location can be replaced by other algorithms. We select two R-peak detection algorithms, i.e. Pan-Tompkins (PT) algorithm [26] and quadratic spline wavelet transform (QSWT) algorithm [27] as examples to explore the CRs of the proposed system when combined with

TABLE 3. Variable numbers and CRs of the proposed two representative systems on the ARRDB.

System	Nt	Wc	Template variables	Context variables	Total variables	channel1 CR	channel2 CR	average CR
S	7	6	252	192	444	2.955	2.995	2.975
L	63	12	2268	12288	14556	3.016	3.063	3.040

TABLE 4. Lossless compression result comparison with several other methods on the ARRDB.

Ref	Transformation/Prediction	Entropy encoding	CR
Miaou and Chao [10]	Distortion-constrained codebook replenishment	Set partitioning in hierarchical tree + Bit-plane encoding	3.068
Zhou [11]	K-means cluster	Huffman encoding	2.93[ab]
Chua and Fang [5]	Discrete pulse code modulation + Error modeling	Golomb-Rice encoding	2.38
Chen and Wang [6]	Adaptive linear predictor	Two-stage Huffman encoding	2.43
Luo *et al.* [7]	Adaptive linear predictor	Two-stage Huffman encoding	2.53
Li *et al.* [8]	Adaptive linear predictor	Modified variable-length encoding	2.67
Deepu and Lian [3]	Short term linear predictor	Fixed-length encoding	2.28
Tseng *et al.* [12]	Takagi-Sugeno fuzzy neural network	Arithmetic encoding	2.96[a]
Tsai and Kuo [9]	Adaptive linear predictor	Golomb-Rice encoding	2.835
Tsai and Tsai [13]	Adaptive linear predictor	Golomb-Rice encoding	2.89
Rzepka [14]	Selective and multichannel linear predictor	Asymmetric numeral systems encoding	2.92
Proposed	Linear predictors + QRS template predictors + Error modeling	Modified Golomb-Rice encoding	(S) 2.975 (L) 3.040

[a] These two references used 12-bit as the resolution of the ARRDB. We set the resolution to 11-bit and recalculated the CR for comparison.
[b] This reference only used the 1st channel of the ARRDB for evaluation.

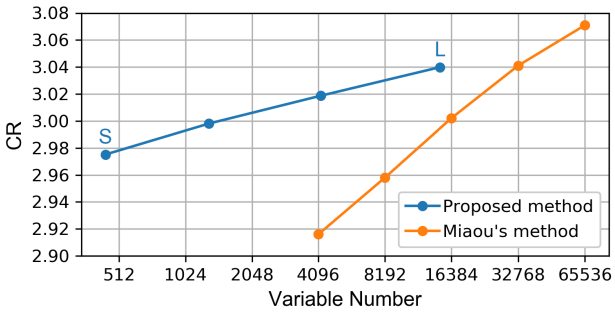


FIGURE 10. Comparison between the proposed method and Miaou's method with different variable numbers.

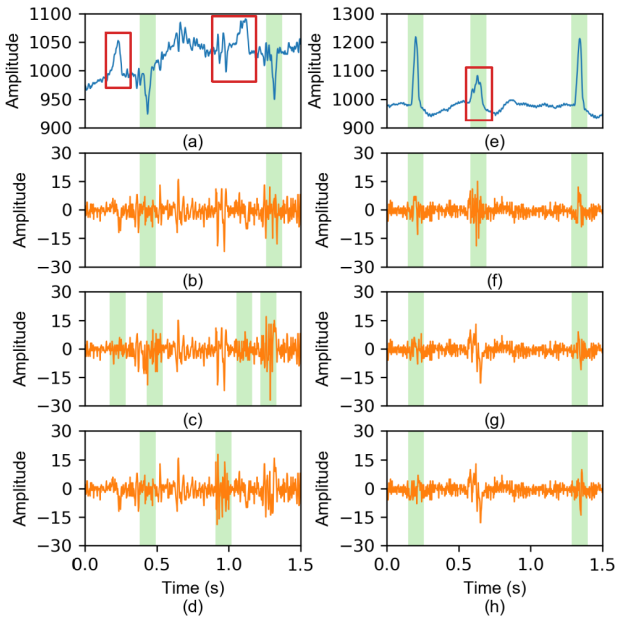


FIGURE 11. (a) A piece of the ECG from the ARRDB recording 108. It contains some sharp noises enclosed in red boxes. (b) Prediction errors of (a) by using the annotations. (c) Prediction errors of (a) by using the PT algorithm. (d) Prediction errors of (a) by using the QSWT algorithm. (e) A piece of the ECG from the ARRDB recording 210. There is an atypical R-peak morphology enclosed in a red box. (f) Prediction errors of (e) by using the annotations. (g) Prediction errors of (e) by using the PT algorithm. (h) Prediction errors of (e) by using the QSWT algorithm. The green backgrounds in (a),(b),(e),(f) indicate the QRS region obtained by the annotation, and the green backgrounds in (c),(d),(g),(h) indicate the QRS region obtained by the respective R-peak detection methods.

other R-peak detection algorithms. Among them, the Pan-Tompkins algorithm is a popular R-peak detection algorithm proposed in 1985, whose detection accuracy is a bit lower than Ieong's algorithm. The QSWT algorithm has a bit higher detection accuracy than Ieong's algorithm. Besides, we also combine the proposed method with the annotations from the ARRDB to obtain the reference CRs. The results are shown in Table. 5. It can be seen that although the CR using the PT algorithm is the lowest, the performance decrement is not significant. The CR of the PT algorithm is at most 0.010 lower than the highest CR, which is obtained by using the annotation. The CRs of the QSWT algorithm and Ieong's algorithm only decrease 0.003 and 0.004 at most. It is because the false detected or false missed R-peaks are generally noisy or have atypical morphologies, which have similar prediction performance by using the QRS region predictor or non-QRS region predictor, as shown in Fig. 11. At present, many R-peak detection algorithms or hardware implementations with higher detection performance than PT algorithm or QSWT algorithm have been published, such as [3], [28]–[31]. It can be inferred that using the proposed compression system combined with these R-peak detection methods can also achieve similar CRs.

B. LIMITATIONS

Though the proposed ECG compression system achieves high CR based on simple prediction and encoding algorithms, its

TABLE 5. CR of each channel on the ARRDB by using the proposed system combined with the R-peak annotations, PT algorithm and QSWT algorithm.

System	Annotations			PT algorithm			QSWT algorithm		
	channel1	channel2	average	channel1	channel2	average	channel1	channel2	average
S	2.958	2.999	2.979	2.951	2.988	2.969	2.956	2.996	2.976
L	3.021	3.066	3.044	3.016	3.06	3.038	3.018	3.064	3.041

limitations should be recognized. First, this method needs to cooperate with an R-peak detection algorithm. Second, this method contains two arrays, QRS templates, and contexts, which occupy some storage space when compressing.

VI. CONCLUSION

In this paper, we have proposed a lossless ECG compression system based on dual-mode prediction and error modeling. The ECG signal is firstly divided into the QRS regions and the non-QRS regions. The 1st order linear predictor is applied to predict the non-QRS region, and the combination of the 3rd order linear predictor and QRS templates predictor is applied to predict the QRS region. After prediction, a context-based error modeling module is added to cancel the statistical bias of the prediction errors. By packaging the modified Golomb-Rice codes of prediction errors, R-peak indication codes, and QRS prediction indexes, the final compressed bitstream is obtained. The proposed system is evaluated on the ARRDB. Experiment results show that the proposed system achieves a CR from 2.975 to 3.040 with memory requirements as low as 444 to 14556 variables. The proposed ECG compression method is highly applicable to both the low-power ECG monitor and the cloud.

REFERENCES

[1] Z. Lu, D. Youn Kim, and W. A. Pearlman, "Wavelet compression of ECG signals by the set partitioning in hierarchical trees algorithm," *IEEE Trans. Biomed. Eng.*, vol. 47, no. 7, pp. 849–856, Jul. 2000.

[2] C.-I. Ieong, P.-I. Mak, C.-P. Lam, C. Dong, M.-I. Vai, P.-U. Mak, S.-H. Pun, F. Wan, and R. P. Martins, "A 0.83-μW QRS detection processor using quadratic spline wavelet transform for wireless ECG acquisition in 0.35-μm CMOS," *IEEE Trans. Biomed. Circuits Syst.*, vol. 6, no. 6, pp. 586–595, Dec. 2012.

[3] C. J. Deepu and Y. Lian, "A joint QRS detection and data compression scheme for wearable sensors," *IEEE Trans. Biomed. Eng.*, vol. 62, no. 1, pp. 165–175, Jan. 2015.

[4] A. Koski, "Lossless ECG encoding," *Comput. Methods Programs Biomed.*, vol. 52, no. 1, pp. 23–33, Jan. 1997.

[5] E. Chua and W.-C. Fang, "Mixed bio-signal lossless data compressor for portable brain-heart monitoring systems," *IEEE Trans. Consum. Electron.*, vol. 57, no. 1, pp. 267–273, Feb. 2011.

[6] S.-L. Chen and J.-G. Wang, "VLSI implementation of low-power cost-efficient lossless ECG encoder design for wireless healthcare monitoring application," *Electron. Lett.*, vol. 49, no. 2, pp. 91–93, Jan. 2013.

[7] G.-A. Luo, S.-L. Chen, and T.-L. Lin, "VLSI implementation of a lossless ECG encoder design with fuzzy decision and two-stage Huffman coding for wireless body sensor network," in *Proc. 9th Int. Conf. Inf., Commun. Signal Process.*, Dec. 2013, pp. 1–4.

[8] K. Li, F. Chen, Y. Pan, R. Huan, and K.-T. Cheng, "Real-time lossless ECG compression for low-power wearable medical devices based on adaptive region prediction," *Electron. Lett.*, vol. 50, no. 25, pp. 1904–1906, Dec. 2014.

[9] T.-H. Tsai and W.-T. Kuo, "An efficient ECG lossless compression system for embedded platforms with telemedicine applications," *IEEE Access*, vol. 6, pp. 42207–42215, 2018.

[10] S.-G. Miaou and S.-N. Chao, "Wavelet-based lossy-to-lossless ECG compression in a unified vector quantization framework," *IEEE Trans. Biomed. Eng.*, vol. 52, no. 3, pp. 539–543, Mar. 2005.

[11] Q. Zhou, "Study on ECG data lossless compression algorithm based on K-means cluster," in *Proc. ETP Int. Conf. Future Comput. Commun.*, Jun. 2009, pp. 91–93.

[12] C.-K. Tseng, L.-J. Kau, and W.-Y. Cheng, "A Takagi-Sugeno fuzzy neural network-based predictive coding scheme for lossless compression of ECG signals," in *Proc. IEEE Region Conf. (TENCON)*, Nov. 2017, pp. 1646–1660.

[13] T.-H. Tsai and F.-L. Tsai, "Efficient lossless compression scheme for multi-channel ECG signal," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 1289–1292.

[14] D. Rzepka, "Low-complexity lossless multichannel ECG compression based on selective linear prediction," *Biomed. Signal Process. Control*, vol. 57, Mar. 2020, Art. no. 101705.

[15] 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, Jun. 2001.

[16] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, Jun. 2000.

[17] F. G. Yanowitz. (Jan. 2018). *Introduction to ECG Interpretation*. [Online]. Available: http://ecg.utah.edu/pdf/

[18] G. M. Friesen, T. C. Jannett, M. A. Jadallah, S. L. Yates, S. R. Quint, and H. T. Nagle, "A comparison of the noise sensitivity of nine QRS detection algorithms," *IEEE Trans. Biomed. Eng.*, vol. 37, no. 1, pp. 85–98, Jan. 1990.

[19] X. Wu and N. Memon, "CALIC—A context based adaptive lossless image codec," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. Conf.*, vol. 4, May 1996, pp. 1890–1893.

[20] X. Wu and N. Memon, "Context-based, adaptive, lossless image coding," *IEEE Trans. Commun.*, vol. 45, no. 4, pp. 437–444, Apr. 1997.

[21] X. Wu, "Lossless compression of continuous-tone images via context selection, quantization, and modeling," *IEEE Trans. Image Process.*, vol. 6, no. 5, pp. 656–664, May 1997.

[22] N. Memon, X. Kong, and J. Cinkler, "Context-based lossless and near-lossless compression of EEG signals," *IEEE Trans. Inf. Technol. Biomed.*, vol. 3, no. 3, pp. 231–238, Sep. 1999.

[23] N. Sriraam and C. Eswaran, "Context based error modeling for lossless compression of EEG signals using neural networks," *J. Med. Syst.*, vol. 30, no. 6, pp. 439–448, Nov. 2006.

[24] M. J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," *IEEE Trans. Image Process.*, vol. 9, no. 8, pp. 1309–1324, Aug. 2000.

[25] A. Kiely, "Selecting the Golomb parameter in rice coding," *IPN Prog. Rep.*, vol. 42, p. 159, Nov. 2004.

[26] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," *IEEE Trans. Biomed. Eng.*, vols. BME–32, no. 3, pp. 230–236, Mar. 1985.

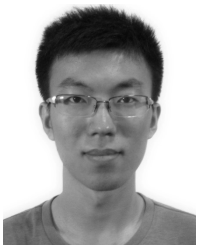
[27] C. Li, C. Zheng, and C. Tai, "Detection of ECG characteristic points using wavelet transforms," *IEEE Trans. Biomed. Eng.*, vol. 42, no. 1, pp. 21–28, Jan. 1995.

[28] C. Nayak, S. K. Saha, R. Kar, and D. Mandal, "Automated QRS complex detection using MFO-based DFOD," *IET Signal Process.*, vol. 12, no. 9, pp. 1172–1184, Dec. 2018.

[29] A. Burguera, "Fast QRS detection and ECG compression based on signal structural analysis," *IEEE J. Biomed. Health Informat.*, vol. 23, no. 1, pp. 123–131, Jan. 2019.

[30] Y. Zou, J. Han, S. Xuan, S. Huang, X. Weng, D. Fang, and X. Zeng, "An energy-efficient design for ECG recording and R-Peak detection based on wavelet transform," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 62, no. 2, pp. 119–123, Feb. 2015.

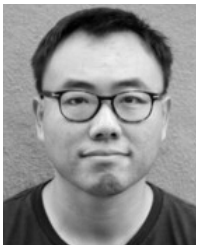
[31] X. Tang, Q. Hu, and W. Tang, "A real-time QRS detection system with PR/RT interval and ST segment measurements for wearable ECG sensors using parallel delta modulators," *IEEE Trans. Biomed. Circuits Syst.*, vol. 12, no. 4, pp. 751–761, Aug. 2018.



MENGHAN JIA received the B.S. degree in information engineering from Zhejiang University, Hangzhou, China, in 2015, where he is currently pursuing the Ph.D. degree with the Institute of VLSI Design. His current research interests include ECG signal analysis and ECG processor.

FEITENG LI received the B.S. degree in the Internet of Things engineering from the School of Information Science and Technology, Southwest Jiaotong University, Chengdu, China, in 2014. He is currently pursuing the Ph.D. degree with the College of Electrical Engineering, Zhejiang University, Hangzhou, China.

His research interests include physiological signals processing with machine learning and ultra-lower-power neural network accelerator.



YU PU (Member, IEEE) received the B.S. degree from Zhejiang University, Hangzhou, China, in 2004, and the Ph.D. degree in electrical engineering from the Eindhoven University of Technology, The Netherlands, in association with the NXP Research, in 2009. From 2009 to 2011, he was a Research Assistant Professor with Sakurai Laboratory, University of Tokyo, Japan. From 2011 to 2012, he was a Research Scientist with the Accelerator Team, IBM Research Zurich, Switzerland. From 2012 to 2013, he was a Principal Scientist with NXP Research, where he led Research and Development in ultra-low-power MCUs. From 2014 to 2019, he was with Qualcomm Research, San Diego, CA, USA, and led Research and Development in always-on Android wearable SoC from concept to mass production. Since 2019, he has been with Alibaba DAMO Computing Research Laboratory, Sunnyvale, CA, USA. He has authored or coauthored more than 40 scientific publications and held more than 20 U.S. patents. He was an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I. He is currently an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II. He is also an Adjunct Full Professor with Shanghai Jiaotong University and Zhejiang University, China.

ZHIJIAN CHEN received the B.S. and Ph.D. degrees from the College of Electrical Engineering, Zhejiang University, Hangzhou, China, in 2006 and 2011, respectivley.

From 2011 to 2013, he was a Postdoctoral Researcher with the College of Electrical Engineering, Zhejiang University, where he has been a Lecturer with the College of Information Science and Electronic Engineering, since 2013. His research interest is ultra-low-power physiological signal processor design.