ECTI Transactions on Computer and Information Technology
Journal homepage: https://ph01.tci-thaijo.org/index.php/ecticit/
Published by the ECTI Association, Thailand, ISSN: 2286-9131

An Efficient Electrocardiography Data Compression

Passakorn Luanloet[1], Watcharapan Suwansantisuk[2] and Pinit Kumhom[3]

ABSTRACT

In healthcare, electrocardiography (ECG) sensors generate a large amount of heart electrical signal that must be efficiently compressed to enable fast data transfer and reduce storage costs. Existing methods for ECG data compression do not fully exploit the characteristics of ECG signals, leading to suboptimal compression. This study proposes a data compression technique for ECG data by exploiting the known characteristics of ECG signals. Our approach combines Savitzky-Golay filtering, detrending, discrete cosine transform, scalar quantization, run-length encoding, and Huffman coding for the effective compression. To optimize the compression performance, we generated quantization intervals tailored to the ECG data characteristics. The proposed method experimentally produces a high compression ratio of 127.61 for a design parameter $K = 8$, a minimum percentage root mean square difference of 1.03% for $K = 128$, and a maximum quality score (QS) of 39.78, where K is the number of quantization intervals. Moreover, we compared the proposed method to state-of-the-art methods on a widely used ECG benchmark dataset. We found that the proposed method outperforms the others in terms of the QS, which measures the overall compression-decompression ability. By enabling more storage and faster data transfer, the proposed method can facilitate the widespread use and analysis of large volumes of ECG data, thereby contributing to advances in healthcare.

Article information:

Keywords: Data Management, ECG, Data Compression in Healthcare, Signal Processing

Article history:
Received: July 26, 2023
Revised: July 30, 2023
Accepted: July 30, 2023
Published: September 2, 2023 (Online)

DOI: 10.37936/ecti-cit.2023173.253629

1. INTRODUCTION

The rapid growth of healthcare technology in recent years has resulted in the development of numerous devices and sensors, significantly increasing the amount of generated data, particularly those used in (ECG) [1]. This massive influx of ECG data requires efficient management, storage, and analysis, which pose significant challenges for healthcare providers [2]. One effective solution to these challenges is data compression, which minimizes the size of the data, making storage and transmission more economical and efficient [3]. However, finding the right balance between compression efficiency, computational complexity, and signal fidelity is crucial, as this balance directly impacts the usability and reliability of the compressed ECG data in clinical settings [4]. Understanding the various techniques for ECG data compression is vital [5,6].

Data compression techniques are classified into two categories: lossless and lossy compression. Lossless compression techniques ensure perfect recovery of the original data from the compressed data, thereby preserving the quality of all the signals. Notable examples of lossless compression include run-length encoding (RLE) [6], Huffman coding [7], and Lempel-Ziv-Welch (LZW) compression [8], among others [9-12]. On the other hand, lossy compression techniques lead to some information loss but offer higher compression ratios (CRs). Lossy compression is proper when slight distortion in the decompressed signal is acceptable. Examples of lossy compression techniques include the discrete cosine transform (DCT) [13], wavelet-based methods [14], scalar quantization (SQ) [15], and subband coding [16]. The decision between lossless and lossy compression depends on the applications and factors such as CR, signal quality, computational

[1,2,3] The authors are with the Department of Electronic and Telecommunication Engineering, King Mongkut's University of Technology Thonburi, Bangkok, 10140, Thailand, Email: passakorn.l@mail.kmutt.ac.th, watcharapan.suw@mail.kmutt.ac.th and pinit.kumhom@mail.kmutt.ac.th.
[2] The corresponding author: watcharapan.suw@mail.kmutt.ac.th

complexity, and real-time performance [17]. For ECG data, lossy compression is more appropriate because physicians would accept an imperfect signal reconstruction that maintains vital features of the heart's electrical signal.

Several studies have presented techniques for compressing ECG data. For instance, a novel technique involving wavelet transform, dead-zone quantization, and modified run-length encoding (RLE) was presented in [18]. This approach boasts superior compression performance, particularly for applications such as Holter monitoring, where the data size can be a bottleneck. However, this technique was only evaluated using ECG signals from a limited dataset. The study [19] discussed the advantages and disadvantages of various ECG data compression techniques, with wavelet-based methods emerging as potential leaders owing to their excellent time-frequency localization and energy compaction properties. However, the performance of these methods depends on various factors, including the selection of the mother wavelet, decomposition level, and quantization strategy. Moreover, standard distortion measures do not always accurately reflect the clinical integrity of ECG signals [19], indicating a substantial need for more reliable and efficient validation methods. Another work [14] studied wavelet-based techniques for ECG data compression, including Huffman encoding and the Lempel-Ziv-Welch (LZW) algorithm. The study showed that combining the wavelet transform and LZW improved the compression ratio (CR) and peak root mean square difference. However, the generalization of these techniques to different types of ECG signals and clinical situations requires further investigation. A novel ECG data compression algorithm was proposed in [20], which uses a DCT-based discrete orthogonal Stockwell transform, dead-zone quantization, and run-length coding. This method exhibited a competitive performance when tested on the MIT-BIH Arrhythmia dataset. A semi-blind mean-opinion-score test indicated that the diagnostic qualities of the reconstructed ECG signals were medically acceptable. However, the performance of this method has only been tested on a specific dataset, which may limit its application to different scenarios. The study [10] proposed an ECG signal compression algorithm using the Blaschke unwinding adaptive Fourier decomposition. This method demonstrated superior performance and effectively preserved the R-peak information of the ECG signal. However, the computational complexity of this method at $O(N^2)$ is a limitation, where N is the number of samples. Nevertheless, the method shows promise for applications in telemedicine and ECG signal storage in e-health scenarios.

While much work has been done toward ECG compression, the existing work is fundamentally limited.

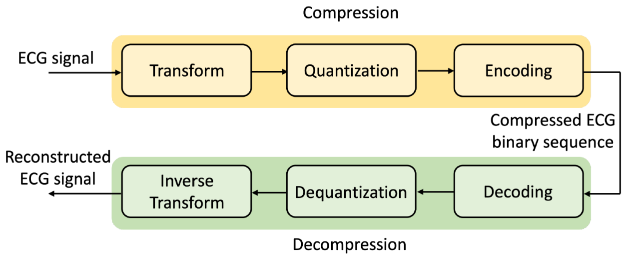


Fig.1: *This study developed methods of compression and decompression for ECG signals.*

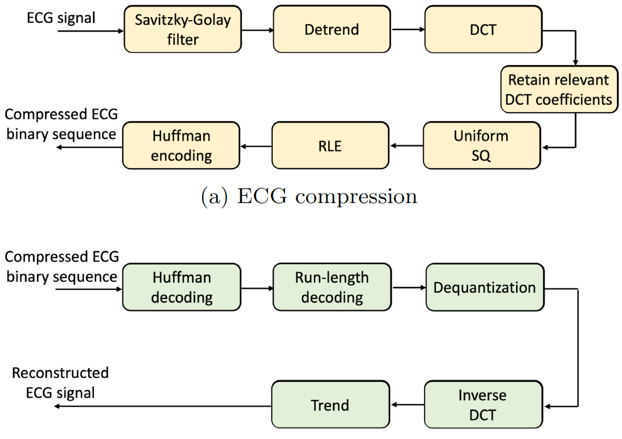
The prevalent issue in these approaches is the potential for further compression in techniques such as RLE and inconsistency in the selection of mother wavelets in wavelet transform-based methods. Moreover, some methods have substantial computational complexity, making them resource-intensive. These limitations indicate a compelling need for a new ECG compression method that can efficiently address the limitations of the state-of-the-art methods. The new ECG compression technique will offer the advantage of a high CR while ensuring signal fidelity.

In this study, we developed a compression technique for ECG data and evaluated its performance. The key idea behind our method is to employ a strategic sequence of operations, namely Savitzky-Golay filtering, detrending, DCT, uniform SQ, RLE, and Huffman coding. This series of operations enabled us to enhance CRs while maintaining low distortion levels, measured as the percentage root mean square difference (PRD). The main contributions of this paper are as follows:

- Design and implementation of a novel ECG signal compression approach that effectively incorporates a variety of signal processing and data compression techniques, resulting in high CRs with minimal distortion.
- Extensive performance evaluations using real-world ECG data, demonstrating that the proposed method is superior to existing state-of-the-art techniques in terms of CR and signal fidelity.

By enabling the efficient compression of ECG signals, our approach contributes to the optimized transmission of high-quality ECG data over networks, thereby promoting timely and accurate remote diagnosis, patient monitoring, and benefiting applications in telemedicine and remote health monitoring.

The remainder of this paper is organized as follows. Section 2 describes the system model. The proposed ECG compression methodology, including its key components, is explained in Section 3. Section 4 is dedicated to the performance evaluation, highlighting the superior performance of the proposed method. Finally, in Section 5, we conclude the paper, summarize the important findings, and suggest future research directions.



(a) ECG compression

(b) ECG decompression

Fig.2: *The proposed (a) compression and (b) decompression methods consist of several steps.*

2. SYSTEM MODEL

Figure 1 presents a schematic diagram of the processes involved in a compression-decompression system for ECG signals. The entry point of the system was an ECG signal. The focus of this study is on compression and decompression. Hence, the input ECG signal is taken to be the analog sequence $\{x(t_0), x(t_1), x(t_2), \ldots, x(t_{n-1})\}$ without loss of generality, where $x(t)$ is the continuous-time ECG signal, n is the sample size, and $t_0 < t_1 < t_2 < \cdots < t_{n-1}$ are the sampling times. The continuous-time ECG signal has been sampled to produce a discrete-time ECG signal at the entry point.

The compression stage is divided into three key subprocesses: transform, quantization, and encoding. The transform process converts the time-domain ECG signal into the frequency domain. This converted signal is then quantized. Quantization is a process that discretizes the signal values and reduces the number of bits required to represent each sample, thereby achieving data reduction. The encoding stage further compresses the signal by replacing the frequently occurring symbols with short codewords. The compressed signal is a significantly minimized version of the original signal, retaining the most critical information from the ECG data with fewer bits.

The decompression stage mirrors the compression stage in reverse order. During decompression, the compressed signal undergoes decoding, which converts codewords back into their original symbols. The signal then passes through dequantization, which converts each discrete signal value to an approximation of the original pre-quantized signal value. Finally, an inverse transform is applied to convert the frequency-domain signal back into the time-domain signal. The reconstruction of the ECG signal was then completed.

In this study, we aim to develop a method that compresses ECG data effectively and reconstructs the approximate signal from the compressed data without losing important diagnostic features. A desired compression-decompression system not only minimizes the size of the compressed signal but also retains the integrity of the critical clinical information in the reconstructed ECG signal.

3. PROPOSED METHOD

In this section, we present a novel ECG signal compression method designed to improve the CR while maintaining low distortion, as measured by the PRD. The proposed method leverages a combination of signal preprocessing, DCT, scalar quantization (SQ), RLE, and Huffman coding. This combination allows for the effective compression of ECG signals.

As an overview, the proposed ECG signal compression and decompression methods are depicted in Figures 2a-2b. In Figure 2a, the compression process commences with Savitzky-Golay filtering, which reduces noise and preserves important characteristics of the ECG signals. The Savitzky-Golay filter is followed by the detrending of the signal to remove the direct-current (DC) offset. After the detrending, a DCT was employed on the signal, enabling us to retain only the components carrying significant energy while nullifying the others. The subsequent steps of our compression strategy involve uniform SQ and RLE, which further reduce the number of bits to represent data. The last step of the process is Huffman coding, which further compresses the data without any loss of information. The output from the compressor was a highly compressed ECG signal represented by a binary sequence and retained a high degree of fidelity to the original signal.

In Figure 2b, the decompression method takes the compressed ECG binary sequence as the input and reverses the operation of the compression method. Steps in the decompression include Huffman decoding, run-length decoding, dequantization, inverse DCT, and trending. Dequantization performs a table lookup and maps each quantization level into the corresponding signal value. The trending adds back the DC offset to the signal. Savitzky-Golay filtering and nullification of the DCT coefficients were employed by the compressor, but their inverses are challenging to obtain or do not exist. Hence, the inverse operations of Savitzky-Golay filtering and nullification did not appear in the decompression process. The compression process involves major signal-processing operations, while the decompression process employs the inverses. For brevity, we describe only the compression operations in detail.

3.1 Signal Preprocessing

The first step in our proposed method involves preprocessing the ECG signal to enhance signal quality, leading to better compression performance. This section describes the preprocessing steps, which include

Savitzky-Golay filtering and detrending of the ECG signal.

The ECG signal is initially filtered using the Savitzky-Golay filter, a smoothing filter that preserves the important features, such as peak amplitudes and widths of the signal. The filter works by fitting a low-degree polynomial to the signal within a fixed-length moving window. In our implementation, we used a third-degree polynomial and a window length of $L = 11$ samples. Let $x_0, x_1, x_2, \ldots, x_{n-1}$ denote the input ECG signal. Let $c_0, c_1, \ldots, c_{L-1}$ denote the Savitzky-Golay coefficients. The output from the Savitzky-Golay filter is denoted by $\{u_0, u_1, u_2, \ldots, u_{n-1}\}$ and is the convolution between the input signal $\{x_0, x_1, x_2, \ldots, x_{n-1}\}$ and the filter coefficients $\{c_0, c_1, \ldots, c_{L-1}\}$. For simplicity, we took the output from the central part of the convolution and took the output to be the same size as the input. In the proposed method, the output from the Savitzky-Golay filter has the size of n.

The Savitzky-Golay filter is a suitable choice for preprocessing because it reduces noise and preserves key characteristics of the signal [21]. Unlike moving average or Gaussian filters, the Savitzky-Golay filter can effectively preserve the signal's higher moments, such as peak heights, widths, and areas, which are essential for accurate medical analysis and diagnosis. Additionally, the Savitzky-Golay filter is less sensitive to sudden changes in the signal, making it suitable for handling ECG signals with varying morphologies [2].

After filtering, the ECG signal may still contain a DC offset, which affects the compression performance. To alleviate this problem, we applied a detrending operation to the filtered signal. Detrending is the process of removing any constant or linear trend from data, resulting in a zero-mean signal.

Detrending is important in ECG signal processing for several reasons. Firstly, a DC offset can introduce artifacts or distortions in the signal, which may affect the accuracy of the subsequent analyses and diagnoses. Moreover, the presence of a DC offset can hinder the performance of compression algorithms because it may cause an increase in the overall signal's energy, resulting in a lower CR. By removing the DC offset, the signal's energy was concentrated around its relevant features, allowing for more efficient compression.

The detrending operation can be achieved by fitting a linear regression model to the signal and subtracting the linear regression line from the original signal. However, for the ECG signal, the slope of the linear regression line will be approximately zero due to the heart signal being approximately periodic. To simplify the detrending operation, we removed only the constant or the DC offset of the input signal. Let $\bar{u}$ denote the sample mean of the sample values $\{u_0, u_1, u_2, \ldots, u_{n-1}\}$, which are the input to the detrending operation. The detrended signal v_i can be obtained by subtracting the DC offset from the input signal: $v_i = u_i - \bar{u}$, for $0 \le i \le n-1$.

3.2 DCT

In the proposed method, the filtered and detrended ECG signals are transformed using DCT, which efficiently encapsulates the signal in the frequency domain. DCT is preferred because of its exceptional energy compaction properties, which enable it to represent the signal's information with a reduced number of coefficients compared to alternative transformations such as the discrete Fourier transform (DFT) or discrete wavelet transform (DWT) [14]. In addition, DCT is a real-valued transform, which leads to simplified computational and storage requirements compared to complex-valued transforms, such as DFT [10]. This aspect makes the DCT particularly appropriate for ECG applications, as it decreases the computational complexity and memory demands for encoding and decoding the compressed signal. Finally, DCT is robust against noise and artifacts within the signal [14], an attribute that is valuable for compression. Given that ECG signals can be tainted by several types of noise, such as muscle contractions, electrode movements, and power line interference, the DCT ensures that such noise sources have a minimal impact on the compressed signal, thereby preserving the critical characteristics of the ECG signal for precise medical analysis and diagnosis.

DCT concentrates the energy of the signal into a limited number of coefficients. The DCT of input signal $\{v_0, v_1, v_2, \ldots, v_{n-1}\}$ is given by the DCT coefficients $\{V_0, V_1, \ldots, V_{n-1}\}$, where

$$V_j = \sum_{i=0}^{n-1} v_i \cos\left[\frac{\pi}{n}\left(i + \frac{1}{2}\right) j\right], \tag{1}$$

and j is the frequency index, which varies from 0 to $n-1$. The DCT coefficients were sorted by their absolute values in descending order, and the number m of coefficients retained for further compression was adaptively determined to capture at least 98.01% of the total energy in the sequence. Here, the total energy E_{tot} of the sequence is defined to be the sum of squared absolute values: $E_{\text{tot}} = \sum_{j=0}^{n-1} |V_j|^2$. The threshold of 98.01% in the energy comes from the threshold of 99% in the norm because $0.9801 = 0.99^2$. In other words, if $V'_0, V'_1, V'_2, \ldots, V'_{n-1}$ denote the permutation of the DCT coefficients in the order of decreasing absolute values, that is $|V'_0| \ge |V'_1| \ge |V'_2| \ge \cdots \ge |V'_{n-1}|$, then the number of retaining DCT coefficient is the smallest integer m such that $\sqrt{\sum_{j=0}^{m-1} |V'_j|^2} \ge 0.99\sqrt{E_{\text{tot}}}$. This adaptive energy-based selection facilitates a balance between data compression and the preservation of signal integrity.

Algorithm 1 Run Length Encoding

```
procedure RunLengthEncoding(vector inputData)
    outputData ← ∅
    count ← 1
    for i = 1 to length(inputData) − 1 do
        if inputData[i] = inputData[i + 1] then
            count ← count + 1
        else
            append (inputData[i], count) to outputData
            count ← 1
        end if
    end for
    append (inputData[end], count) to outputData
    return outputData
end procedure
```

Algorithm 2 Huffman Encoding

```
procedure HuffmanEncoding(inputData)
    frequency ← calculateFrequency(inputData)
    priorityQueue ← createPriorityQueue(frequency)
    while priorityQueue.size > 1 do
        left ← priorityQueue.extractMin()
        right ← priorityQueue.extractMin()
        newNode ← createInternalNode(left, right)
        priorityQueue.insert(newNode)
    end while
    huffmanTree ← priorityQueue.extractMin()
    codeTable ← generateCodeTable(huffmanTree)
    encodedData ← encodeData(inputData, codeTable)
    return encodedData, codeTable
end procedure
```

3.3 SQ

In addition to DCT, SQ was employed in the proposed method to compress the ECG signal further. SQ is a widely used lossy compression technique that represents a set of data points using a limited number of representative points. A uniform quantization approach was used in the proposed method. Uniform SQ partitions the range of possible input values into a set of intervals or bins. Each bin was assigned a representative value, and every input value falling within that bin was quantized to the bin's representative value. Let K denote the number of quantization intervals. Parameter K influences both CR and PRD. By adjusting K, we can control the trade-off between CR and PRD to achieve the desired balance between compression efficiency and signal fidelity.

In the proposed method, SQ is implemented as follows. Let $C = \{V'_0, V'_1, \ldots, V'_{m-1}, 0\}$ denote the set of retaining DCT coefficients and zero, which is the nullified coefficient. Elements of C are the possible input values to the SQ. Let $V_{\max}$ and $V_{\min}$ denote the maximum and minimum elements, respectively, of set C. Let Δ denote the quantization step size: $\Delta = (V_{\max} - V_{\min})/K$. If $V \in C$ is the input to the quantizer, then the proposed quantization level is given by

$$q(V) = \left(\frac{p(V)}{K} \cdot (K-1) \right), \tag{2}$$

where $p(V) = \left(\frac{V - V_{\min}}{\Delta}\right)$. The value of $p(V)$ is either $0, 1, 2, \ldots,$ or K. The expression for $q(V)$ scales the value of $p(V)$ to be in the set $\{0, 1, 2, \ldots, K-1\}$. Function $q(\cdot)$ produces K quantization levels as needed. Overall, SQ reduces the number of bits required to represent the DCT coefficients. The combination of DCT and SQ allows robust compression while preserving the critical features of the ECG signal.

3.4 RLE and Huffman Encoding

Following the SQ process, the next step is RLE and Huffman encoding, which reduce the data size. RLE is a simple lossless compression technique that is particularly effective for ECG data containing a long sequence of repeating values. In Algorithm 1, the command *outputData* $\leftarrow \emptyset$ is used to initialize an empty set for the output. RLE encodes a sequence of repeated values as a single value and a count of its consecutive occurrences, effectively reducing the amount of data required for representation. RLE was applied to the SQ levels obtained in the previous step. By exploiting any potential repetition in the SQ levels, RLE can further compress the data before passing them to Huffman encoding.

Huffman coding is an optimal prefix-free coding algorithm that assigns variable-length binary codes to the symbols in the input data based on their probabilities of occurrence. Symbols with higher probabilities are assigned shorter codewords, whereas those with lower probabilities are assigned longer codewords. Huffman coding ensures that the average codeword length is minimized, resulting in efficient compression.

In Algorithm 2, Huffman encoding proceeds as follows. A binary tree known as a Huffman tree was constructed, with the leaf nodes signifying the symbols present in the input data and their respective probabilities. The probabilities are computed from the relative frequencies of input symbols. Initially, unique symbols within the data were identified and converted to double-precision. Subsequently, the occurrence of each unique symbol in the data is counted to create a histogram. These counts are then divided by the total number of data elements to yield the relative frequencies that approximate the probabilities. The tree is constructed using a bottom-up approach, consistently merging the two nodes with the lowest probabilities until only one node remains, serving as the root node of the tree. The binary codes for the symbols were then generated by traversing the tree from the root to the leaf nodes, with bit 0 designated for the left branch and 1 for the right branch. The final binary codewords are then transmitted or stored alongside the codebook, Huffman tree, and RLE encoding information, which are necessary for the reconstruction process. Overall, the combination of DCT, SQ, RLE, and Huffman encoding in our proposed method allows for an efficient and compact representation of the ECG signal while preserving the essential features required for accurate analysis and diagnosis.

4. PERFORMANCE EVALUATION

In this section, we discuss the performance metrics, provide a canonical example of the signal output at each stage of the compressor, and compare the proposed method with state-of-the-art methods on a standard ECG dataset.

4.1 Performance Metrics

We used standard performance metrics, which are the CR, PRD, and QS, to evaluate the compression and reconstruction performance. CR quantifies the degree of compression and is the ratio of the original ECG signal size to the compressed signal size:

$$\mathrm{CR} = \frac{N_{\mathrm{orig}}}{N_{\mathrm{comp}}}, \tag{3}$$

where N_{orig} is the number of bits required to store the original ECG signal, and N_{comp} is the number of bits required to store the compressed signal. A higher CR indicates better compression performance, representing a higher degree of reduction in the amount of data needed to represent the signal. The PRD measures the distortion introduced by the compression process, and is the ratio of the root mean square difference between the original and reconstructed ECG signals to the root mean square amplitude of the original signal:

$$\mathrm{PRD} = \frac{100\%}{\sqrt{n} \times x_{\max}} \sqrt{\sum_{i=0}^{n-1} (x_i - \hat{x}_i)^2}, \tag{4}$$

where $\hat{x}_i$ is the ith sample of the reconstructed ECG signal, and $x_{\max}$ is the maximum absolute value of the ECG signal: $x_{\max} = \max_{0 \le i \le n-1} |x_i|$. A lower PRD indicates better compression performance because it represents a lower degree of distortion introduced by the compression process. QS is the ratio of CR to PRD:

$$\mathrm{QS} = \frac{\mathrm{CR}}{\mathrm{PRD}}. \tag{5}$$

The QS metric provides an overall measure of the effectiveness of the compression method, considering both the CR and fidelity of the reconstructed signal. A higher QS value indicates a better compression performance.

In addition to the metrics mentioned above, computational complexity was employed as a performance metric. This metric evaluates the computational resources in terms of running time required by the compression and decompression algorithms. A method with lower computational complexity is preferred, particularly for real-time applications, as it ensures faster processing. We measured the performance of the compression-decompression methods by the CR, PRD, QS, and computational complexity.

Table 1: *The DCT is the most complex step of the proposed compression.*

Step	Complexity
Savitzky-Golay filter	$O(n)$
Detrending	$O(n)$
DCT	$O(n \log n)$
SQ	$O(n)$
RLE	$O(n)$
Huffman encoding	$O(n \log K)$

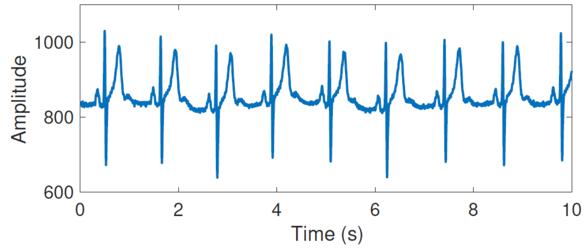
Table 2: *The inverse DCT is the most complex step of the proposed decompression.*

Step	Complexity
Huffman decoding	$O(n \log K)$
RLE	$O(n)$
Dequantization	$O(n)$
Inverse DCT	$O(n \log n)$

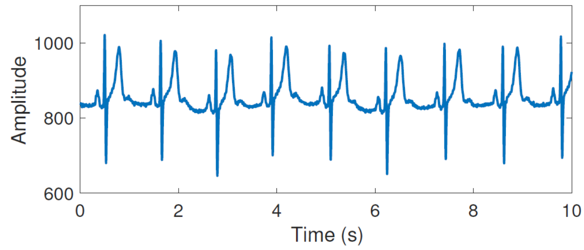
4.2 Computational Complexity

Tables 1 and 2 show the complexity for every step in the ECG compression and decompression. As shown in Table 1, the compression incorporates various steps ranging from the Savitzky-Golay filter, detrending, and the DCT to uniform SQ, RLE, and Huffman encoding. The window size $L = 11$ is a constant, implying a linear time $O(Ln) = O(n)$ for Savitzky-Golay filtering. Except for the DCT and Huffman encoding, which introduce logarithmic factors, every other step operates in linear time, making them efficient. Similarly, the decompression phase outlined in Table 2 remains mostly linear in complexity except for the inverse DCT and Huffman decoding.

According to Tables 1 and 2, the bottleneck steps are DCT and inverse DCT, which have a running time

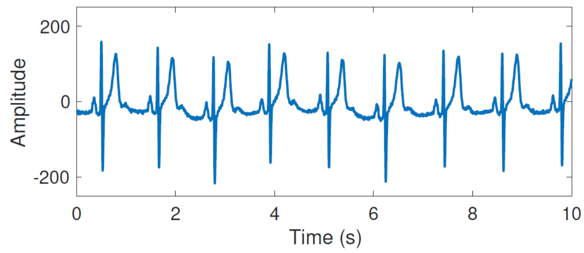


(a) An original ECG signal

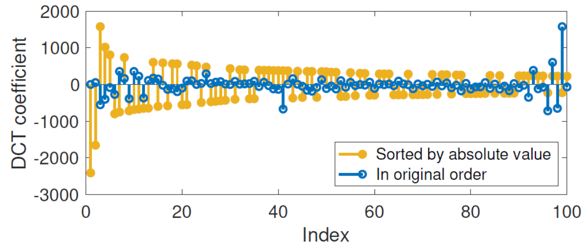


(b) After Savitzky-Golay filtering

Fig.3: *The Savitzky-Golay filter effectively preserves the essential features of the signal. The original ECG signal in (a) is visually similar to the filtered signal in (b).*



(a) After the detrending



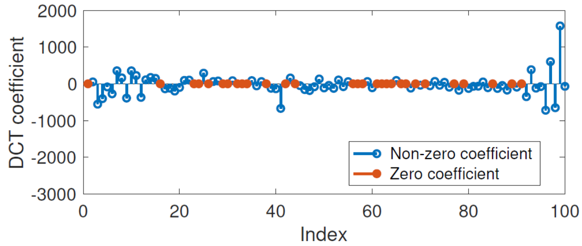
(b) After DCT

Fig.4: *The signal in (a) is an output of the detrending, which removes the DC offset. The DCT coefficients of the signal are shown in (b).*

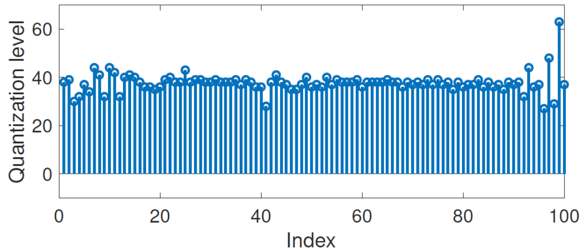
complexity of $O(n \log n)$. Huffman coding and decoding are less complex than the DCT and inverse DCT because $K < n$. Summing the running time complexity of all steps, the overall running time complexity of compression and decompression is $O(n \log n)$.

4.3 A Canonical Example

To validate the proposed method, we examined the output of each key component using an actual ECG signal from the PhysioNet MIT-BIH Arrhythmia dataset as the input. In Figure 3, we present two stages of ECG signal processing: (a) the original ECG signal, which serves as a reference for evaluating the compression quality at each stage of the proposed method, and (b) the ECG signal after applying the Savitzky-Golay filter. The filtered signal effectively reduced the noise while maintaining the essential features of the original ECG. The Savitzky-Golay filter smoothed the signal and preserved the high-frequency

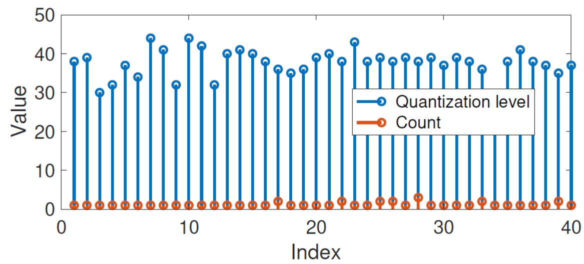


(a) After non-significant DCT coefficients were zeroed

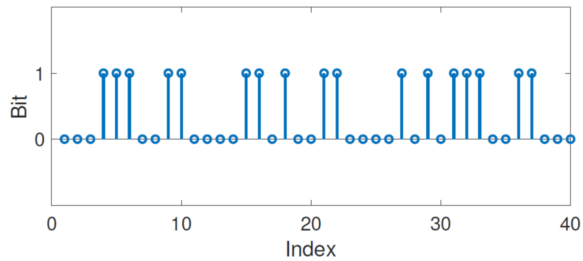


(b) After SQ

Fig.5: *In the lossy compression method, (a) shows the result after discarding non-significant DCT coefficients, while (b) displays the outcome after quantizing the coefficient values.*



(a) After RLE



(b) After Huffman encoding

Fig.6: *For the lossless compression method, (a) shows the result of applying the RLE, while (b) illustrates the output from Huffman encoding.*

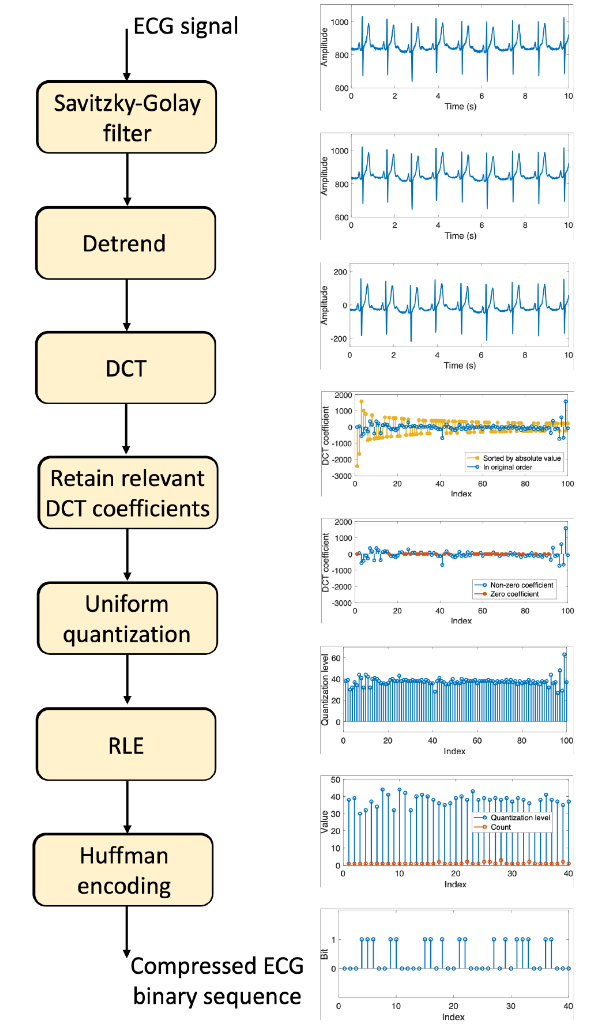


Fig.7: *The proposed ECG signal compression method involves a series of signal processing steps.*

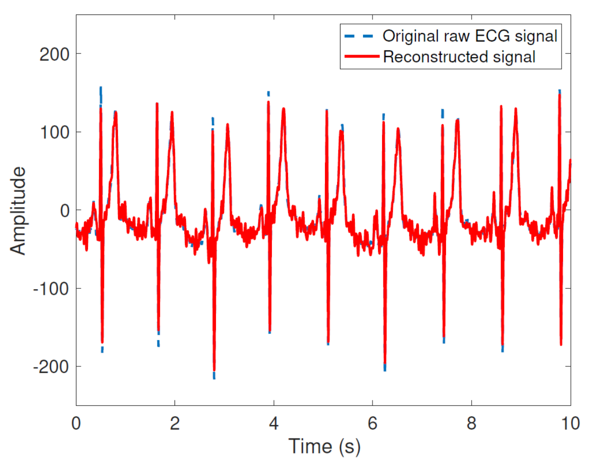


Fig.8: *The decompressed signal (solid red line) uses a parameter $K = 64$. It is visually similar to the original signal (dashed blue line), which is the 117th record of the MIT-BIH Arrhythmia dataset.*

content of the original ECG signal.

Figure 4 shows two more stages of the compression process: (a) the ECG signal after detrending, and (b) the ECG signal after the DCT has been applied. The DCT converts the time-domain ECG signal into a frequency domain and represents the signal with fewer coefficients. Figure 5 illustrates the subsequent steps of compression: (a) the signal after the DCT, which retains a specific number of DCT coefficients representing approximately 98% of the signal energy, and (b) the ECG signal after the SQ. Finally, Figure 6 presents the last two stages of the compression process: (a) the signal after being compressed with RLE, which encodes consecutive occurrences of the same value as a single value and its repetition count, and (b) the binary signal after being compressed with Huffman coding.

A visual comparison between the original ECG signal in Figure 7 demonstrates that the proposed compression method effectively preserves the key characteristics of the original ECG signal with minimal distortion and noise introduced throughout the compression process. The figure provides a detailed overview of each stage of compression, emphasizing the successful preservation of the essential features in the compressed ECG signal.

To further illustrate the effectiveness of the proposed compression method, a comparison between the original and decompressed ECG signals is shown in Figure 8. In Figure 8, the blue dashed waveform is the original ECG signal, which serves as our reference, whereas the red solid waveform shows the decompressed signal. A close visual inspection of these two waveforms reveals that the decompressed signal preserves the overall shape and essential characteristics of the original ECG signals. Despite the substantial reduction in the data size achieved through the compression process, the decompressed signal exhibited minimal distortion and noise, demonstrating the effectiveness of the proposed method in preserving the critical information necessary for accurate interpretation and diagnosis. This finding underscores the potential of the proposed ECG signal compression method as a robust tool for efficient ECG data storage and transmission for subsequent diagnostic use.

4.4 Compression Performance

The experiments were conducted using ECG data from the PhysioNet MIT-BIH Arrhythmia dataset, a publicly available dataset containing 48 half-hour, two-channel ambulatory ECG recordings. The ECG signals in this dataset were sampled at a frequency of 360 Hz with an 11-bit resolution over a 10 mV range. The proposed ECG compression method was applied to the ECG dataset. In these experiments, we varied the number K of quantization intervals for the SQ to 8, 16, 32, 64, and 128. For a given K,

Table 3: *With a suitable parameter K, the proposed method has the highest AQS.*

Method	ACR	APRD	AQS
Kumar *et al.* [22]	5.67	2.4	2.36
Huang *et al.* [23]	13	2.89	–
Jha *et al.* [19]	11.49	3.43	3.82
Lee *et al.* [24]	5.19	0.23	22.32
Jha *et al.* [20]	6.27	5.37	1.49
Lee *et al.* [25]	12	6.35	1.89
Pandey *et al.* [22]	18.59	1.06	17.57
Abo-Zahlad *et al.* [26]	14.23	3.84	3.69
Kolekar *et al.* [18]	17.18	3.92	4.37
Tan *et al.* [10]	35.53	1.47	32.58
Proposed ($K = 8$)	127.61	3.77	39.78
Proposed ($K = 16$)	46.85	2.48	21.11
Proposed ($K = 32$)	25.32	1.48	19.37
Proposed ($K = 64$)	18.82	1.10	19.37
Proposed ($K = 128$)	15.75	1.03	17.52

the proposed method generated the CR, PRD, and QS for each of the 48 recordings. The average compression ratio (ACR), average percentage root mean square difference (APRD), and average quality score (AQS) over the 48 recordings were reported in Table 3 for the proposed method. On the other hand, the ACR, APRD, and AQS for each existing method were taken from the original paper, which also used the PhysioNet MIT-BIH Arrhythmia dataset.

For $K = 8$ quantization intervals, the proposed method achieved an impressive ACR of 127.61, with an APRD of 3.77%, yielding an AQS of 39.78. As the number of quantization intervals were increased to 16, the ACR reduced to 46.85, but the APRD improved to 2.48%, resulting in an AQS of 21.11. As the number of quantization intervals was further increased to 32, the ACR decreased to 25.32, but the APRD further improved to 1.48%, yielding an AQS of 19.37. Increasing the number of quantization intervals to 64 resulted in a further reduction in ACR to 18.82, and an improvement in APRD to 1.10%, yielding an AQS identical to the case of $K = 32$ setting at 19.37. For the number of quantization intervals of 128, the ACR was reduced to 15.75, with the APRD of 1.03%, resulting in an AQS of 17.52.

From Table 3, the proposed method stands out for its high ACR, especially when $K = 8$, indicating that it can significantly reduce the size of ECG data. Even when we changed the value of K, our method performed well. It consistently provides accurate signal reconstructions as measured by the APRD, which takes a smallest value at $K = 128$. Lastly, the high AQS values across different K indicate that our method successfully balances between reducing the data size and maintaining the crucial details of the ECG signal. The proposed method can be a practical choice for ECG signal compression.

Overall, the proposed method delivered superior ACRs while maintaining low APRDs, even as the number of quantization intervals increased. AQS provides evidence of an effective balance between the two measures, demonstrating the overall efficiency and robustness of the proposed compression method. As the number of quantization intervals decreased from $K = 128$ to $K = 8$, a steady improvement in the AQS was observed, with the highest AQS achieved at $K = 8$. In addition, the energy stored in the chosen DCT coefficients preserves the essential features in the compressed ECG signals. Our approach retained approximately 98% of the signal energy in the selected DCT coefficients. This high-energy retention ensures that the compressed signals capture the information necessary for an accurate medical diagnosis.

5. CONCLUSIONS

In this study, we proposed a novel approach for ECG signal compression using a preprocessing step, the DCT, SQ, RLE, and Huffman coding. Our experiments were conducted using the MIT-BIH Arrhythmia dataset and demonstrated the effectiveness of our proposed method. In particular, we achieved a high CR while maintaining a low PRD across various numbers of quantization intervals, as indicated by consistently impressive QS results. The simplicity and efficiency of our method, mainly owing to the use of uniform quantization, suggest its suitability for real-time applications.

This study can be extended in several directions. Future research should investigate alternative strategies for quantization. In addition, future research may examine the role of design parameters, such as the percentage of total energy in the DCT coefficients, in the compression performance. This continuous refinement and optimization of our ECG compression methodology will ensure its robustness, efficiency, and reliability in providing accurate ECG data to healthcare providers.

ACKNOWLEDGMENT

This work was supported in part by the Petchra Pra Jom Klao Ph.D. Research Scholarship from King Mongkut's University of Technology Thonburi and by the Faculty of Engineering, King Mongkut's University of Technology Thonburi, under the Research Strengthening Project.

References

[1] U. Jayasankar, V. Thirumal, and D. Ponnurangam, "A survey on data compression techniques: From the perspective of data quality, coding schemes, data type and applications," *Journal of King Saud University-Computer and Information Sciences*, vol. 33, no. 2, pp. 119–140, 2021.

[2] M. Islam, G. Tangim, T. Ahammad, M. Khondokar *et al.*, "Study and analysis of ECG signal using Matlab & Labview as effective tools," *International Journal of Computer and Electrical Engineering*, vol. 4, no. 3, p. 404, Jun. 2012.

[3] A. Thorén, A. Rawshani, J. Herlitz, J. Engdahl, T. Kahan, L. Gustafsson, and T. Djärv, "ECG-monitoring of in-hospital cardiac arrest and factors associated with survival," *Resuscitation*, vol. 150, pp. 130–138, 2020.

[4] World Health Organization, "Cardiovascular diseases (CVDs)," `https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)`, 2022, retrieved April 20, 2023.

[5] L. Sörnmo and P. Laguna, "Electrocardiogram (ECG) signal processing," *Wiley Encyclopedia of Biomedical Engineering*, 2006.

[6] S. Hamdan, A. Awaian, and S. Almajali, "Compression techniques used in IoT: A comparitive study," in *International Conference on new Trends in Computing Sciences*, Amman, Jordan, Oct. 2019, pp. 1–5.

[7] K. Rana and S. Thakur, "Data compression algorithm for computer vision applications: A survey," in *International Conference on Computing, Communication and Automation*, Greater Noida, India, 2017, pp. 1214–1219.

[8] T. Li, T. Zhao, M. Nho, and X. Zhou, "A novel RLE & LZW for bit-stream compression," in *IEEE International Conference on Solid-State and Integrated Circuit Technology*, Hangzhou, China, Oct. 2016, pp. 1600–1602.

[9] K. Sharma and K. Gupta, "Lossless data compression techniques and their performance," in *International Conference on Computing, Communication and Automation*, Greater Noida, India, May 2017, pp. 256–261.

[10] C. Tan, L. Zhang, and H.-t. Wu, "A novel Blaschke unwinding adaptive-Fourier-decomposition-based signal compression algorithm with application on ECG signals," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 2, pp. 672–682, Mar. 2018.

[11] D. Birvinskas, V. Jusas, I. Martisius, and R. Damasevicius, "Fast DCT algorithms for EEG data compression in embedded systems," *Computer Science and Information Systems*, vol. 12, no. 1, pp. 49–62, Aug. 2015.

[12] R. B. Patil and K. Kulat, "Audio compression using dynamic Huffman and RLE coding," in *International Conference on Communication and Electronics Systems*, Coimbatore, India, Oct. 2017, pp. 160–162.

[13] R. J. Barsanti and A. Athanason, "Signal compression using the discrete wavelet transform and the discrete cosine transform," in *Proceedings of IEEE Southeastcon*, Jacksonville, FL, USA, Apr. 2013, pp. 1–5.

[14] S. Chandra, A. Sharma, and G. K. Singh, "A comparative analysis of performance of several wavelet based ECG data compression methodologies," *Innovation and Research in BioMedical Engineering*, vol. 42, no. 4, pp. 227–244, May 2021.

[15] S. Jancy and C. Jayakumar, "Various lossless compression techniques surveyed," in *International Conference on Science Technology Engineering & Management*, Chennai, India, Mar. 2017, pp. 65–68.

[16] A. Fathi and F. Faraji-kheirabadi, "ECG compression method based on adaptive quantization of main wavelet packet subbands," *Signal, Image and Video Processing*, vol. 10, pp. 1433–1440, Jul. 2016.

[17] C. Patauner, A. Marchioro, S. Bonacini, A. U. Rehman, and W. Pribyl, "A lossless data compression system for a real-time application in HEP data acquisition," *IEEE Transactions on Nuclear Science*, vol. 58, no. 4, pp. 1738–1744, Aug. 2011.

[18] M. Kolekar, C. Jha, and P. Kumar, "ECG data compression using modified run length encoding of wavelet coefficients for Holter monitoring," *Innovation and Research in BioMedical Engineering*, vol. 43, no. 5, pp. 325–332, 2022.

[19] C. K. Jha and M. Kolekar, "Electrocardiogram data compression techniques for cardiac healthcare systems: A methodological review," *Innovation and Research in BioMedical Engineering*, vol. 43, no. 3, pp. 217–228, 2022.

[20] C. K. Jha and M. H. Kolekar, "Electrocardiogram data compression using DCT based discrete orthogonal Stockwell transform," *Biomedical Signal Processing and Control*, vol. 46, pp. 174–181, Sep. 2018.

[21] N. Rastogi and R. Mehra, "Analysis of Savitzky-Golay filter for baseline wander cancellation in ECG using wavelets," *International Journal of Engineering Sciences & Emerging Technologies*, vol. 6, no. 1, pp. 2231–6604, Aug. 2013.

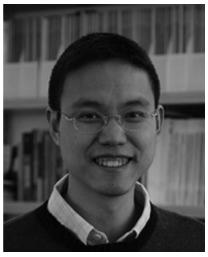
[22] R. Kumar, A. Kumar, and R. K. Pandey, "Beta wavelet based ECG signal compression using lossless encoding with modified thresholding," *Computers & Electrical Engineering*, vol. 39, no. 1, pp. 130–140, 2013.

[23] B. Huang, Y. Wang, and J. Chen, "ECG compression using the context modeling arithmetic coding with dynamic learning vector–scalar quantization," *Biomedical Signal Processing and Control*, vol. 8, no. 1, pp. 59–65, Jan. 2013.

[24] S. Lee, J. Kim, and M. Lee, "A real-time ECG data compression and transmission algorithm for an e-health device," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 9, pp. 2448–2455, Sep. 2011.

[25] H. Lee and K. M. Buckley, "ECG data compression using cut and align beats approach and 2-D transforms," *IEEE Transactions on Biomedical Engineering*, vol. 46, no. 5, pp. 556–564, May 1999.

[26] M. M. Abo-Zahhad, T. K. Abdel-Hamid, and A. M. Mohamed, "Compression of ECG signals based on DWT and exploiting the correlation between ECG signal samples," *International Journal of Communications, Network and System Sciences*, vol. 2014, Jan. 2014.



Watcharapan Suwansantisuk received B.S. degrees in electrical and computer engineering and in computer science from Carnegie Mellon University, Pennsylvania, in 2002, and M.S. and Ph.D. degrees in electrical engineering from the Massachusetts Institute of Technology in 2004 and 2012, respectively. He is currently an Assistant Professor at King Mongkut's University of Technology Thonburi (KMUTT), Thailand.

Before joining KMUTT, he spent summers at the University of Bologna, Italy, as a visiting research scholar, and at the Alcatel-Lucent Bells Laboratory, NJ, as a research intern. His main research interests include wireless communications, synchronization, and statistical signal processing.

Dr. Suwansantisuk serves on the technical program committees for various international conferences and served as the symposium Co-Chair for the IEEE Global Communications Conference in 2015. He received the Leonard G. Abraham Prize in the field of communications systems from the IEEE Communications Society in 2011, jointly with Prof. M. Chiani and Prof. M. Win, and the Best Paper Award from the IEEE RIVF International Conference on Computing and Communication Technologies in 2016, jointly with N. Chedoloh.

Passakorn Luanloet received his B.Eng. degree in electronic and telecommunication engineering from King Mongkut's University of Technology Thonburi (KMUTT) in 2014. He is currently pursuing a doctoral degree in electrical and information engineering technology at the Department of Electronic and Telecommunication Engineering, KMUTT. His research interests include data compression, compression techniques, data acquisition systems, and signal processing.

Pinit Kumhom received the B.Eng. degree in electrical engineering from the King Mongkut's Institute of Technology Thonburi, Thailand, in 1988, and the Ph.D. degree in electrical and computer engineering from Drexel University, Pennsylvania, in 2000. He is currently an Assistant Professor with the King Mongkut's University of Technology Thonburi. His research interests include the Internet of Things and its applications, digital system design and implementation, and signal and image processing.