Computers in Biology and Medicine 71 (2016) 24–34

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/cbm

# Phonocardiogram signal compression using sound repetition and vector quantization

Hong Tang [a,*], Jinhui Zhang [a], Jian Sun [a], Tianshuang Qiu [a], Yongwan Park [b]

[a] Department of Biomedical Engineering, Dalian University of Technology, Dalian, PR China
[b] Department of Information and Communication Engineering, Yeungnam University, Dae-dong, Gyeongsan-si, Gyeongsangbuk-do 712749, Republic of Korea

ARTICLE INFO

*Article history:*
Received 23 July 2015
Accepted 14 January 2016

*Keywords:*
Phonocardiogram signal
Signal compression
Sound repetition
Time–frequency decomposition
Vector quantization

ABSTRACT

*Background:* A phonocardiogram (PCG) signal can be recorded for long-term heart monitoring. A huge amount of data is produced if the time of a recording is as long as days or weeks. It is necessary to compress the PCG signal to reduce storage space in a record and play system. In another situation, the PCG signal is transmitted to a remote health care center for automatic analysis in telemedicine. Compression of the PCG signal in that situation is necessary as a means for reducing the amount of data to be transmitted. Since heart beats are of a cyclical nature, compression can make use of the similarities in adjacent cycles by eliminating repetitive elements as redundant. This study proposes a new compression method that takes advantage of these repetitions.
*Methods:* Data compression proceeds in two stages, a training stage followed by the compression as such. In the training stage, a section of the PCG signal is selected and its sounds and murmurs (if any) decomposed into time–frequency components. Basic components are extracted from these by clustering and collected to form a dictionary that allows the generative reconstruction and retrieval of any heart sound or murmur. In the compression stage, the heart sounds and murmurs are reconstructed from the basic components stored in the dictionary. Compression is made possible because only the times of occurrence and the dictionary indices of the basic components need to be stored, which greatly reduces the number of bits required to represent heart sounds and murmurs. The residual that cannot be reconstructed in this manner appears as a random sequence and is further compressed by vector quantization. What we propose are quick search parameters for this vector quantization.
*Results:* For normal PCG signals the compression ratio ranges from 20 to 149, for signals with median murmurs it ranges from 14 to 35, and for those with heavy murmurs, from 8 to 20, subject to a degree of distortion of ~5% (in percent root-mean-square difference) and a sampling frequency of 4 kHz.
*Discussion:* We discuss the selection of the training signal and the contribution of vector quantization. Performance comparisons between the method proposed in this study and existing methods are conducted by computer simulations.
*Conclusions:* When recording and compressing cyclical sounds, any repetitive components can be removed as redundant. The redundancies in the residual can be reduced by vector quantization. The method proposed in this study achieves a better performance than existing methods.


## 1. Background

Heart sounds are generated by the interactions between heart chambers, valves and great vessels and the blood flowing through them. Mechanical vibrations reflect the turbulence that occurs when heart valves close. Traditionally, a stethoscope is used in cardiac auscultation to listen to these sounds that provide important acoustic information regarding the condition of the heart. In home health monitoring applications, a phonocardiogram (PCG) signal may be continuously recorded for hours, days or even weeks to catch events related to heart hemodynamics. If a single channel PCG signal is collected for one day with a sampling frequency of 4 kHz and a digitization depth of 16 bits, the data storage space required is 24*60*60*4000*16 bits ≈ 0.69 GBytes. More storage space is needed if multi-channel signals are collected over several days. To save storage space in a record and play system, it is necessary to compress the data. Furthermore, the rapidly developing field of telemedicine has made it possible to transmit a PCG record to a health center for automatic remote analysis. Let us consider a scenario, as shown in Fig. 1. A person with a smart

* Corresponding author. Tel.: +86 411 84706009x3013.
*E-mail address:* tanghong@dlut.edu.cn (H. Tang).

http://dx.doi.org/10.1016/j.compbiomed.2016.01.017
0010-4825/

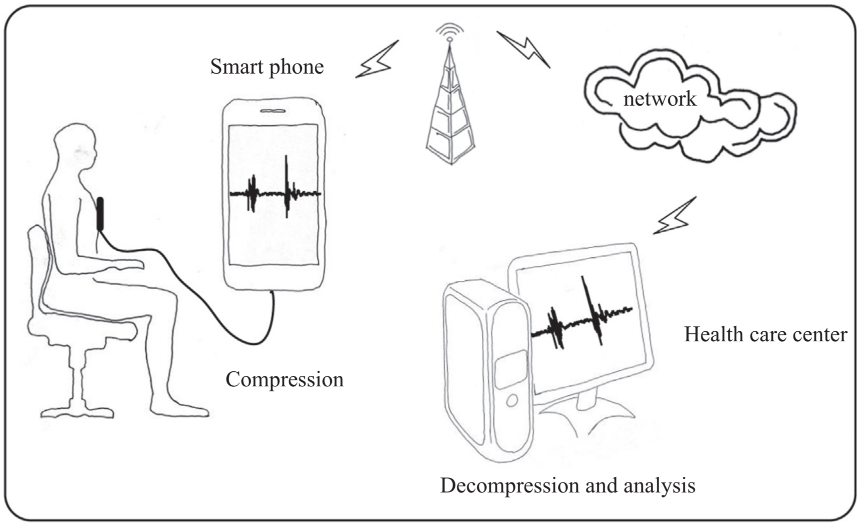


**Fig. 1.** Illustration of the remote analysis of a PCG signal in a mobile medicine application.

phone can access a remote health care service anywhere and anytime. The PCG signal (or other physiological signal) can be continuously recorded by attaching a PCG sensor at the required position. The data are compressed by the embedded algorithm in the smart phone. Remote analysis benefits from improved transmission efficiency via the compression technique. The PCG signal is decompressed at the health care center without any loss of physiological information.

To achieve the highest possible compression ratio with the lowest possible distortion, many researchers have developed compression methods for PCG data. The first reported method to compress a PCG signal used a wavelet transform (WT) [1]. The PCG signal was divided into non-overlapping blocks, and the wavelet transform was applied to each block independently. A hard threshold was applied to select the prominent wavelet coefficients, and the PCG signal was reconstructed from the reduced number of coefficients and their positions in the wavelet domain. To obtain the optimal wavelet, block length, and WT decomposition level for the compression, the authors used a genetic algorithm [2]. The compression was later implemented in an FPGA-based system [3]. It can be concluded that the smaller the number of selected coefficients, the higher the compression ratio of the method will be. However, the wavelet coefficients are not sparse enough, and to achieve an acceptably low distortion, a great number of coefficients must be selected. The computer experiments showed that the compression ratio was roughly 6 for a normal PCG signal at a sampling frequency of 8 kHz subject to a distortion of 6.7% in terms of the percent root-mean-square difference. Based on this reasoning, a transform was desired that would decompose a PCG signal into sparse components. Modified WT-based methods appeared in the following years and exhibited a similar performance [4–9]. Another hard threshold method based on adaptive Fourier decomposition was proposed in [10]. In addition to the lossy compression methods mentioned above, lossless compression in [11] using the Lempel–ziv–storer–szymanski technique was applied in a remote heart sound monitoring system. However, the compression ratios of lossless methods are generally lower than those of lossy methods. Consequently, more data needs to be transmitted over the network, while at the same time it is necessary to reduce the amount of data transmitted without the loss of physiological information.

The objective of this study was to achieve a higher compression ratio and lower degree of distortion when storing or transmitting heart sound signals. This was accomplished by using a previously undocumented approach whereby those sounds or murmurs are discarded as redundant that recur from one cardiac cycle to another. The heart sounds or murmurs are decomposed into their time–frequency components and the basic components are then extracted by clustering to produce a dictionary. The heart sounds or murmurs can then be reconstructed from their basic components, where for each component only the dictionary index and the time of occurrence need to be recorded. This strategy greatly reduces the number of bits required to represent heart sounds or murmurs. The residual, that part of the signal that cannot be reconstructed by the basic components, is further compressed by vector quantization, which results in a better fidelity.

## 2. Methods

### *2.1. Dictionary formulations*

#### *2.1.1. Time–frequency decomposition*

Heart sounds and/or murmurs are fast, time-varying waveforms that can be decomposed into components in a joint time–frequency domain. Decomposition can be achieved through various methods, such as the discrete wavelet-based method [1–9], the matching pursuit method [12,13], or adaptive time–frequency decomposition [14,15]. The discrete wavelet-based method and the matching pursuit method both produce non-sparse components, typically many more than are produced by adaptive time–frequency decomposition. So in this study we adopted the adaptive time–frequency decomposition since it produces a very sparse representation of heart sounds and murmurs. It is written as

$$x(t) = \sum_{i=1}^{M} a_i e^{(t-t_i)^2/(2\sigma_i^2)} \cos(2\pi f_i t + \beta_i), \tag{1}$$

where $x(t)$ is a heart sound signal, which is considered to be the sum of $M$ components. Each component is characterized by five parameters, namely amplitude, $a_i$; occurrence time, $t_i$; frequency, $f_i$; time-support, $\sigma_i$; and phase, $\beta_i$. The $i$-th component can be determined by a parameter vector $[a_i, t_i, f_i, \sigma_i, \beta_i]$. The presentation of (1) is similar to the short time Fourier transform (STFT)

$$H(t,f) = \int h(\tau) w(t-\tau) e^{-2\pi f \tau} d\tau, \tag{2}$$

where $h(t)$ is the signal to be analyzed. The sliding Gaussian window, $w(t)$, is centered at $t_i$ and covers the heart sounds. Its width should thus be equal to or greater than the span of heart sounds, commonly less than 0.2 s. So for the purposes of this study, the width of the window is set as 0.2 s. Eq. (2) shows that the parameters can be estimated based on the STFT. The $i$-th component to be separated from the signal is the most prominent component with the maximum magnitude; it is obtained by searching the peaks of the time–frequency spectrum. Occurrence time, $t_i$; frequency, $f_i$; amplitude, $a_i$; and phase, $\beta_i$ can be read directly at the found peak location. The frequency fineness is related to the FFT size used in the operation. The time resolution of the occurrence time $t_i$ is related to the frame interval of the sliding window. In this study, the FFT size selected is 2̂10, which results in a frequency fineness close to 4 Hz. And the frame interval is the sample period, i.e. 0.25 ms (the sample frequency in this study is 4 KHz).The $i$-th component with an undetermined time-support, $\sigma_i$, is

$$s_i(t,\sigma_i) = a_i e^{(t-t_i)^2/(2\sigma_i^2)} \cos(2\pi f_i t + \beta_i). \tag{3}$$

The time-support parameter, $\sigma_i$, can be determined by the following procedure. The residual after the $i$-th separation is

$$h_i(t,\sigma_i) = h_{i-1}(t) - s_i(t,\sigma_i), \tag{4}$$

where $h_0(t)$ is the original PCG signal. The energy of the $i$-th residual is

$$\rho_i(\sigma_i) = \int h_i(t,\sigma_i)^2 dt. \tag{5}$$

The time-support, $\sigma_i$, is chosen so as to minimize the residual energy. This search is similar to grid searching in a predefined range. The time-support value depends on the damping rate of a component. The faster the component damps, the smaller the time-support should be, and a slower damping rate goes with longer time-support. Each heart sound consists of many components, the duration of none of which can thus exceed the whole time span of heart sounds, which is smaller than or equal to 0.2 s. In this study, the range considered for the time-support search is in [0 0.2] s. That range is wide enough for all possible time-supports. The value of $M$ is determined by the number of iterations needed for the separation procedure to leave a sufficiently small final residual energy. In this study, the normalized residual energy threshold used to determine $M$ is 0.01.

For example, two PCG signals are decomposed into their components, as shown in Figs. 2 and 3. The PCG signal in Fig. 2 was collected from a normal subject in the authors ' lab without any sound-proofing. Each component separated from the record is represented by a circle and then plotted in its own planes: the joint time–amplitude plane, joint time–frequency plane, joint time–phase plane and joint time–time-support plane, as shown in Fig. 2(b)–(e). The PCG signal shown in Fig. 3 contains aortic murmurs. The separated components are plotted in their joint planes, as shown in Fig. 3(b)–(e). It can be seen that the separated components are highly repetitive from cycle to cycle, and this repetitive nature of the components is the basis for the PCG signal compression described in this study.

### 2.1.2. Dictionary formulation by SOM clustering

A portion of the heart sound signal to be compressed is selected as a training signal; that portion containing dozens of continuous cardiac cycles is decomposed into components by the above STFT technique. The components are characterized by their parameter

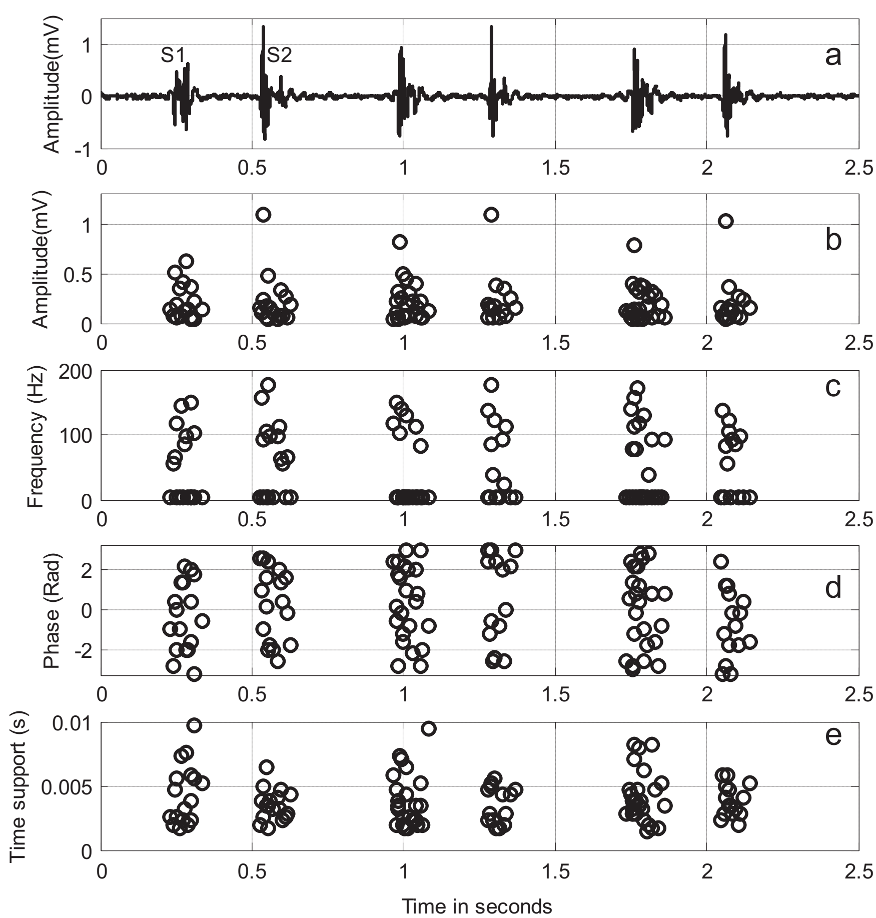


**Fig. 2.** Decomposition of a normal PCG signal. (a) A normal PCG signal. (b)–(e) The components are plotted in their joint time–amplitude, time–frequency, time–phase and time–time-support planes.

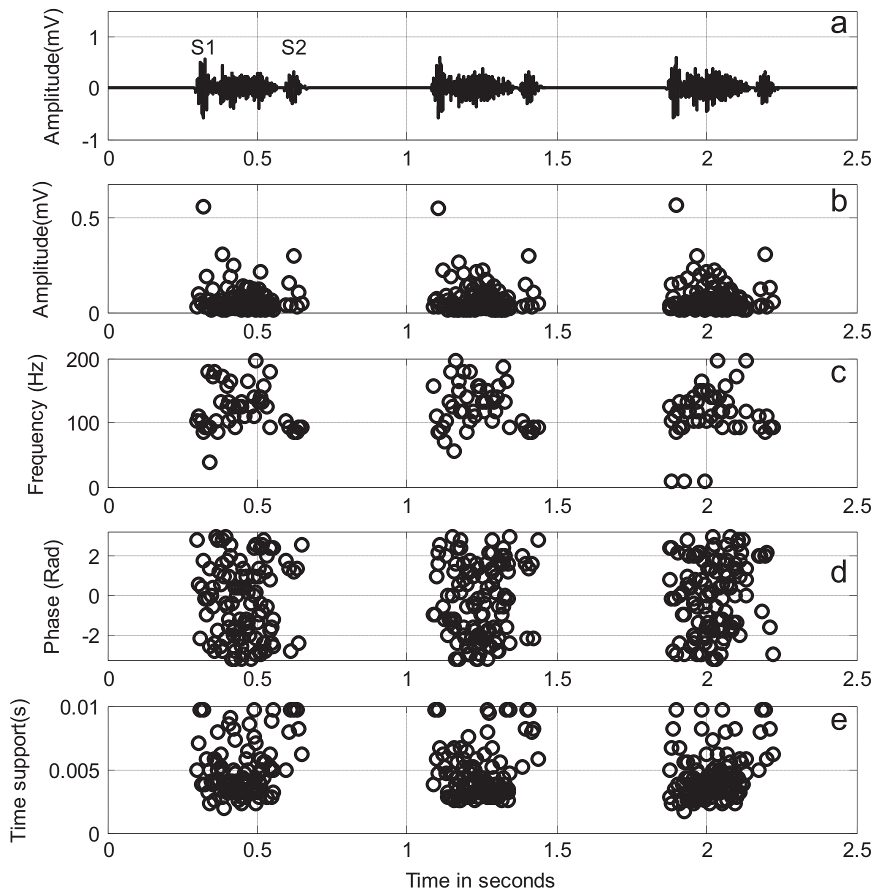


**Fig. 3.** Decomposition of an aortic stenosis PCG signal. (a) An aortic stenosis PCG signal. (b)–(e) The components are plotted in their joint time–amplitude, time–frequency, time–phase and time–time-support planes.

vectors $\{[a_i, t_i, f_i, \sigma_i, \beta_i], 1 \le i \le I\}$, where $I$ is the number of components decomposed from the PCG training signal. The value of $I$ is determined by the number of repetitions needed for the separation procedure to result in a residual energy of a predefined threshold value. The components tend to be repetitive in adjacent cardiac cycles, which appears as cyclical redundancy except for the occurrence times. Let us assume that each parameter vector $[a_i, f_i, \sigma_i, \beta_i]$ is represented as a point in four-dimensional space where all these points $\{[a_i, f_i, \sigma_i, \beta_i], 1 \le i \le I\}$ will congregate and form clusters, thereby identifying cyclical repetitions. Clustering permits the reduction of redundant storage and transmission which otherwise would be needed to represent sound or murmur repetitions. In this study, a self-organizing map (SOM) neural network is used for clustering in four-dimensional space [16]. The SOM has the authors' expected advantages. It is trained through unsupervised learning to produce a discretized representation of the input space of the training samples. The output of the cluster centers from the network preserves the topological properties of the input space. The components characterized by the cluster centers are those recurring in each cardiac cycle. They are the basic components allowing the reconstruction of any heart sound or murmur, and together they constitute a dictionary

$$\left\{[a_j^c, f_j^c, \sigma_j^c, \beta_j^c], 1 \le j \le J\right\}, \tag{6}$$

where $[a_j^c, f_j^c, \sigma_j^c, \beta_j^c]$ is the center point of the $j$-th cluster and $J$ is the number of cluster centers. The only parameter needed to be set in advance for SOM implementation is the expected number of cluster centers. That number is related to the degree of component repetition. The higher the proportion of repetitive components, the lower the number of required centers. So in order to capture a variety of heart sound signals containing murmurs, a relatively large number of components is desired. For the purposes of this study the number was set as 256, as determined through computer simulations. A number that high can accommodate even a signal containing heavy murmurs.

### 2.1.3. Reconstruction based on the dictionary

The PCG signal to be compressed is denoted as $y(t)$. The heart sounds or murmurs in the PCG signal to be compressed are decomposed into components by the above STFT technique. Therefore, any one of the components has a counterpart in the dictionary, i.e., the nearest one in name of Euler distance. Therefore, the heart sounds or murmurs in $y(t)$ can be reconstructed from the dictionary counterparts and the associated occurrence times

$$y(t) = \sum_{j=1}^{K} s(t_j, q_j), \tag{7}$$

where $q_j$ is the index of the $j$-th counterpart component in the dictionary, $t_j$ the occurrence time of the $j$-th component. $K$ is the number of counterpart components selected from the dictionary. The parameter vector mapped to index $q_j$ is $[a_j^c, f_j^c, \sigma_j^c, \beta_j^c]$. The reconstructed version of the heart sounds or murmurs is

$$y(t) = \sum_{j=1}^{K} a_j^c e^{(t-t_j)^2/(2(\sigma_j^c)^2)} \cos(2\pi f_j^c t + \beta_j^c), \tag{8}$$

where the occurrence time $t_j$ appears in each cardiac cycle. The occurrence times are stored in an array as one-dimensional sequence that needs to be compressed using lossless Huffman coding because of their non-uniform occurrence frequency, as Huffman coding produces minimum redundant codes and uses fewer bits to represent more frequent occurrences, more bits to represent less frequent occurrences. Huffman coding thus produces a very low average code length for each sequence.

In practical terms, the PCG signal consists not only of heart sounds and/or murmurs, which can be modeled by (1), but also of random features (such as ambient acoustic noise, instrument noise, artifacts induced by movement, etc., and the potential combined effects of these on the signal), which are not described by the basic components. A residual, $r(t)$, is used to represent the random part, $y(t)$ is taken as the sum of two parts according to

$$y(t) = \sum_{j=1}^{K} a_j^c e^{(t\ t_j)^2/(2(\sigma_j^c)^2)} \cos(2\pi f_j^c t + \beta_j^c) + r(t). \tag{9}$$

The PCG signal is divided into two parts. The first partial signal allows reconstruction of the heart sounds including most murmurs as modeled by the sum of time–frequency components, according to (1), see Figs. 2(a) and 3(a). Basic components used for the reconstruction are selected from the associated dictionaries, as shown in Figs. 4(b) and 5(b). The second part of the signal contains the residual, $r(t)$, which appears as a random sequence and cannot be modeled by using the basic components; see Figs. 4(c) and 5(c). However, that part, which may contain important physiological information, needs to be recovered in the decompressed version to achieve the lowest possible distortion degree through further compression using vector quantization (VQ), as described in the next section.

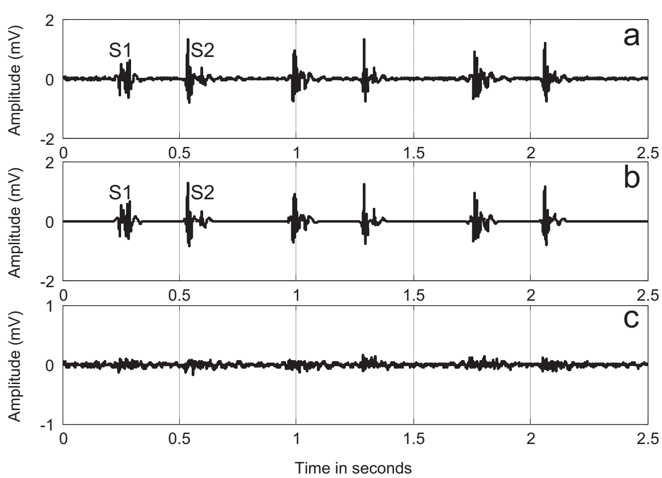


**Fig. 4.** A normal PCG signal reconstruction using the dictionary. (a) A normal PCG signal. (b) Reconstructed signal. (c) Residual.

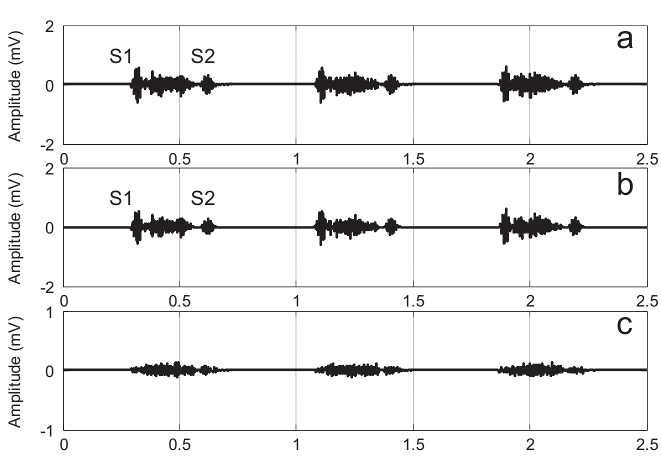


**Fig. 5.** Aortic stenosis PCG signal reconstruction using the dictionary. (a) An aortic stenosis signal. (b) Reconstructed signal. (c) Residual.

### 2.2. Residual compression by vector quantization

The residuals shown in Fig. 4(c) and Fig. 5(c) are random, noise-like features representing the parts that cannot be reconstructed using the associated dictionary. These residuals are further compressed by vector quantization (VQ), an effective approach for adapting random sequences [17,18]. There has to be an embedded structure in the residual sequence even if it is a random noise-like sequence. Vector quantization is used to extract the structure of the residual sequence, typically divided into non-overlapping segments that are called vectors. Each segment includes $k$ consecutive samples of the sequence. A segment is considered to be a $k$-dimensional vector. Thus, the residual is grouped into many vectors. The vectors, by virtue of their embedded structure, form clusters in the $k$-dimensional space. A clustering operation, such as the well-known k-means clustering, can be performed using an algorithm where each cluster is represented by its centroid. All the centroids together form the codebook, and their total number constitutes the codebook size. The codebook, which is built in the training stage, can represent any input vector and is defined as

$$\{\mathbf{Y}_i, i = 1, \cdots, N\}, \tag{10}$$

where $N$ is the number of centroids and $\boldsymbol{Y}_i$ is a code-vector in the codebook. At the encoder, the input vector $\boldsymbol{X}$ is compared with each code-vector $\boldsymbol{Y}_i$ in the codebook. Using the mean-squared error to measure the distance, the nearest code-vector $\boldsymbol{Y}_j$ can be found such that

$$\|\boldsymbol{X} - \boldsymbol{Y}_j\|^2 \le \|\boldsymbol{X} - \boldsymbol{Y}_i\|^2, \quad i = 1, 2, \cdots N \tag{11}$$

Index and codebook serve to construct a lookup table. Each input vector $\boldsymbol{X}$ has its counterpart in the code-vector $\boldsymbol{Y}_j$ so that for the purposes of data compression only the index $j$ needs to be stored on the side of the encoder, which greatly reduces the space required for input vector storage. An optimal codebook formulation, since it largely determines compression performance, must reflect the residual's embedded structure. This will be discussed in the next section.

### 2.3. Formulation of the Codebook

#### 2.3.1. Performance indicators

There are two performance indicators for evaluating data compression: the compression ratio (CR) and the percent root-mean-square difference (PRD). The general definition of the CR is

$$\mathrm{CR} = \frac{\text{bits of signal before compression}}{\text{bits after compression}}. \tag{12}$$

For the purposes of this study, the CR is calculated according to

$$\mathrm{CR} = \frac{b^* L_y}{C_t + C_q + C_j}, \tag{13}$$

where $b$ is the number of A/D bits for digitizing, $L_y$ is the number of samples of the PCG signal to be compressed and $C_t$, $C_q$ and $C_j$ are the numbers of bits used to store the occurrence times, component indices in the dictionary, and vector indices in the codebook, respectively.

The PRD used to evaluate the distortion between the original signal and the decompressed signal is

$$\mathrm{PRD} = \sqrt{\sum_i (x(i) - x^d(i))^2 \Big/ \sum_i x^2(i)} \times 100\%, \tag{14}$$

where $x$ is the PCG signal to be compressed and $x^d$ is the decompressed version. Note that the PRD is defined as the percent root-mean-square difference between the original and the decompressed version. Computer simulations show that it is a reliable indicator. The difference between original and decompressed signals is so small that it cannot be detected by a visual check in a normal printed time-scale (4 cm/s), even if the PRD is near 15%.

### 2.3.2. Compression performance with respect to the codebook

In VQ, the proximity of the output vector to the input vector is determined by the codebook formulation, i.e., the proximity is related to $k$ and $N$. Computation simulations were conducted, where $k$ varied from 1 to 30, and $N$ was selected to be 256, 128, 64, 32 or 16. Fig. 6 shows the compression performance with respect to $k$ and $N$. For a fixed $k$, the PRD increases with decreasing $N$. For a fixed $N$, the PRD increases with increasing $k$. To achieve an acceptably low PRD, a codebook must have nearly optimal $k$ and $N$ values.

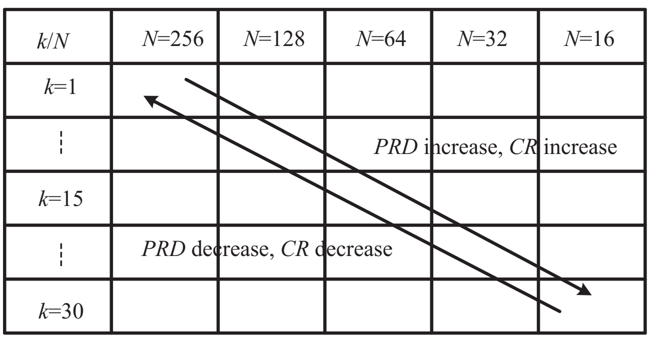


**Fig. 6.** Compression performance with respect to $k$ and $N$.

It is possible to find a nearly optimal $k$ and $N$ by searching through each grid in the table. However, that would be very time-consuming. A quick search algorithm is proposed in the next subsection to obtain a suboptimal $k$ and $N$.

### 2.3.3. A quick search algorithm for a suboptimal k and N

The algorithm is shown in Fig. 7. It consists of initialization, candidate search for codebook size, and local search.

Before the compression begins, a desired PRD is set, which is named $\mathrm{PRD_d}$ in this study. As the operator prepares to compress the signal, the goal of the compression is to obtain the highest possible CR subject to the $\mathrm{PRD_d}$. The initial selection of ($k$, $N$) can theoretically begin from any grid in the table. Usually, this selection starts from a middle grid, for example, $k=15$ and $N=64$. The corresponding $\mathrm{PRD}_{(k,\,N)}$ and $\mathrm{PRD}_{(k+1,\,N/2)}$ can be calculated for the initial selection.

The purpose of the candidate search is to find the best candidates for the codebook size. Once $\mathrm{PRD_d}$ meets the inequality $\mathrm{PRD}_{(k,\,N)} \le \mathrm{PRD}_d < \mathrm{PRD}_{(k+1,\,N/2)}$, the codebook size will be either $N$ or $N/2$.

The purpose of the local search is to find the optimal vector size to go with $N$ and $N/2$. That is, to find the $k$ value that will result in the PRD approaching but not exceeding the desired PRD value and then to examine the resulting ($k$, $N$) pairs so as to choose the ($k$, $N$) pair allowing the highest CR. The search will start in both the left

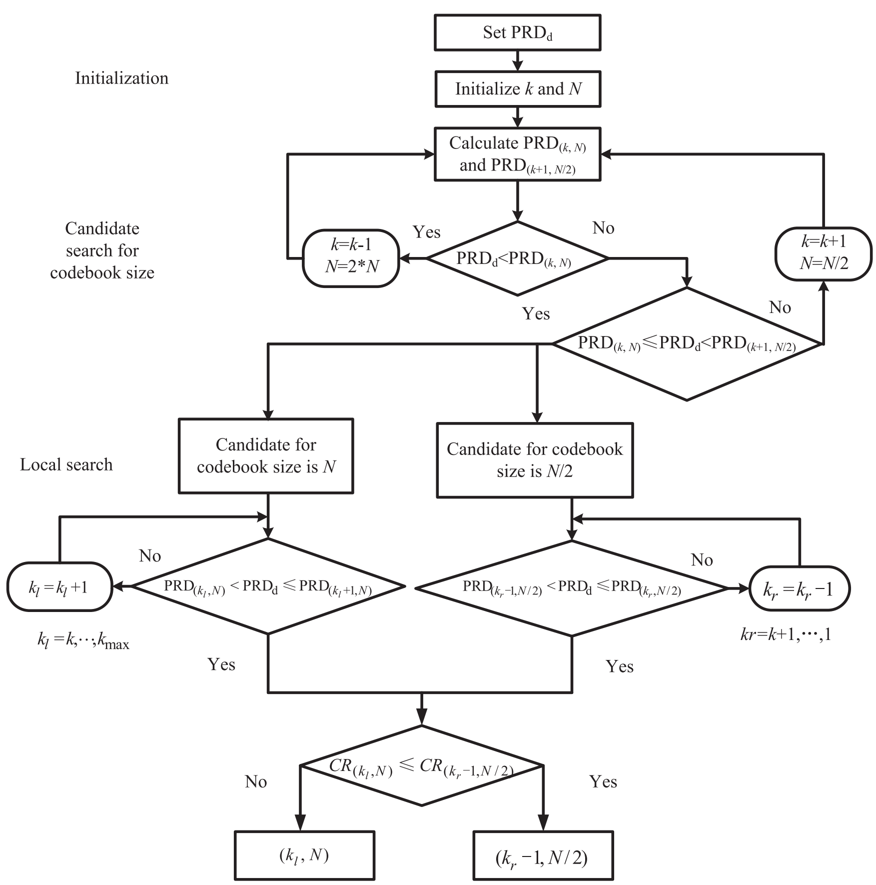


**Fig. 7.** The search process for determining the suboptimal vector dimensionality and codebook size.

and right branch. In the left branch, once the pair ($k_l$, $N$) meets $\mathrm{PRD}_{(k_l,N)} < \mathrm{PRD}_d \le \mathrm{PRD}_{(k_l+1,N)}$ as $k_l$ is increased from $k$ to $k_{max}$ in increase of one, the pair ($k_l$, $N$) must be the best one available to go with $N$. Similarly, in the right branch, once the pair ($k_r$, $N/2$) meets $\mathrm{PRD}_{(k_r-1,N/2)} < \mathrm{PRD}_d \le \mathrm{PRD}_{(k_r,N/2)}$ as $k_r$ is reduced from $k+1$ to 1 in decrements of one, $k_r-1$ will be assumed as the best codebook vector dimensionality for the codebook size of $N/2$. The inequality in the final diamond shape should read $\mathrm{CR}_{(k_l,N)} \le \mathrm{CR}_{(k_r-1,N/2)}$. The pair of ($k_l$, $N$) is chosen if the inequality is not satisfied and the pair ($k_r-1$, $N/2$) is chosen if the inequality is satisfied.

It must be stated here that the final ($k_l$, $N$) pair and the final ($k_r-1$, $N/2$) pair might not be globally optimal. That follows from the logical structure of the table (the PRD value increases with decreasing $N$ for a fixed $k$ and the PRD value increases with increasing $k$ with a fixed $N$). Multiple suboptimal pairs of ($k$, $N$) may be found for a given desired PRD if the initialization is different.

#### 2.3.4. Operational steps of the proposed compression method

The operational steps of the proposed PCG signal compression, which consists of a training stage and a compression stage, are shown in Fig. 8.

##### 2.3.4.1. Training stage

Step 1: Select a portion of the PCG signal that needs to be compressed as the training signal.
Step 2: Decompose the heart sounds and/or murmurs in the training signal into components.
Step 3: Perform clustering on the components to create a dictionary.
Step 4: Reconstruct the heart sounds and/or murmurs based on the dictionary to obtain the residual.
Step 5: Set the desired PRD. Search for $k$ and $N$, then build a codebook using a clustering algorithm.

##### 2.3.4.2. Compression stage

Step 1: Decompose the heart sounds and/or murmurs in the PCG signal into components using the STFT technique shown in Section 2. Find the nearest vector from the dictionary for each component. Store the dictionary indices and the occurrence times.
Step 2: Compress the residuals by VQ using the codebook. Store the vector indices.
Step 3: Encode the indices of the components, the occurrence times and the vector indices by Huffman coding.

It is known that heart sounds from different subjects have different time–frequency characteristics. So, the dictionary and codebook trained for one subject does not generalize to another. The training stage must be operated separately for every subject.

## 3. Results

### 3.1. Data sources

The proposed compression scheme was evaluated using various PCG signals from two sources: low-noise level PCG signals downloaded from the cardiology center website database at Dundee University, United Kingdom [19], and PCG signals collected in the authors' laboratory that does not offer any particular soundproofing and is thus acoustically similar to most home health care environments, where noise control tends to be unavailable. All of the PCG records were converted to a 16-bit sampling depth and a sampling frequency of 4 kHz. The bandwidth of heart sounds and murmurs is typically less than 800 Hz. A sampling frequency of 4 kHz is adequate for the Nyquist sampling theorem. A higher frequency was not necessary and it might in fact have introduced redundancies.

### 3.2. Compression performance evaluation

The desired PRD for all of the PCG signals was set at 5%. 17 typical cases were used for performance evaluation. Table 1 shows the 9 cases downloaded from the website database. Table 2 shows 8 real signals collected in the authors' lab. In the operation of the proposed compression scheme, ten cardiac cycles were included in

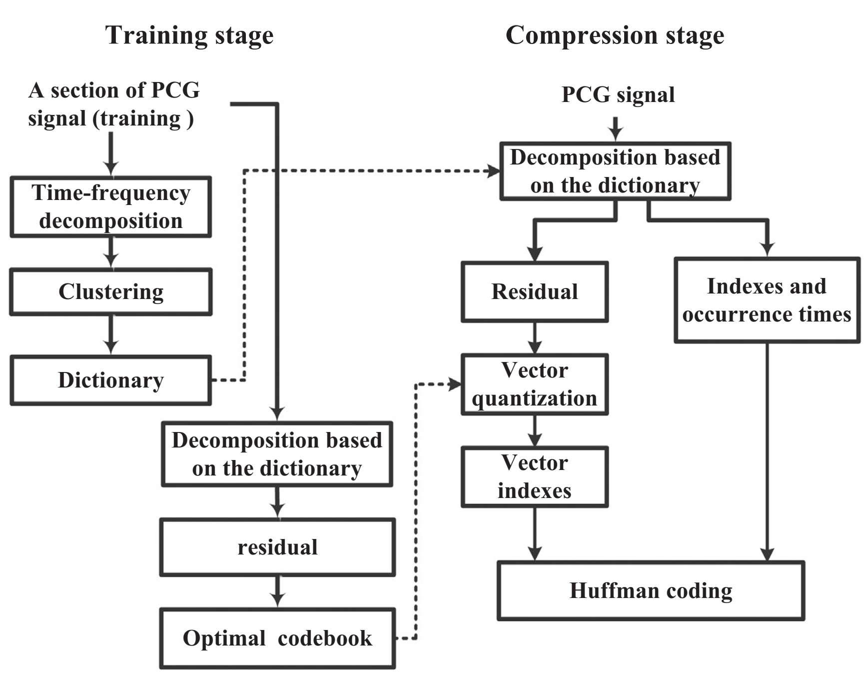


**Fig. 8.** Operational steps of the proposed PCG signal compression method.

the training signal for each experiment. In both tables, compression performance is indicated by PRD and CR.

The compression ratios of the first five signals in Table 1 were extremely high, greater than 90. These signals were almost free of noise and highly repetitive between adjacent cycles and thus met the assumptions of the proposed compression method very well. The heart sounds could be fully reconstructed by a few basic components from the dictionary. Their residuals varied at low magnitude but could still be represented by a small codebook size ($N$=16, 32) and high vector dimension ($k$=22, 24, 30). Therefore, extremely high compression ratios could be achieved for these five signals. That was not the case, however, for records 6, 7, 8 and 9.

**Table 1**
Performance evaluations for signals from the website database subject to a desired PRD of 5%.

| No. | PCG record[a] | Record length (s) | $k$ | $N$ | PRD (%) | CR |
|---|---|---|---|---|---|---|
| #1 | Normal heart sound | 96.5 | 30 | 16 | 4.98 | 149.2 |
| #2 | Third heart sound | 99.9 | 30 | 32 | 4.60 | 97.1 |
| #3 | Fourth heart sound | 100.1 | 22 | 16 | 4.82 | 95.3 |
| #4 | Summation gallop | 86.8 | 24 | 16 | 4.93 | 92.4 |
| #5 | Atrial fibrillation | 120.2 | 20 | 16 | 4.89 | 111.9 |
| #6 | Mitral stenosis | 96.4 | 20 | 16 | 4.96 | 34.1 |
| #7 | Aortic stenosis | 88.9 | 8 | 128 | 4.95 | 19.2 |
| #8 | Pulmonary stenosis | 99.4 | 4 | 32 | 4.83 | 21.3 |
| #9 | Mitral regurgitation | 68.8 | 14 | 256 | 4.86 | 16.9 |

[a] Data were obtained from the website database of the cardiology center at Dundee University, United Kingdom [19]. Records #1–#5 are very repetitive and almost noise-free.

**Table 2**
Performance evaluations for real case signals subject to a desired PRD of 5%.

| No. | PCG record | Record length (s) | $k$ | $N$ | PRD (%) | CR |
|---|---|---|---|---|---|---|
| #10 | Normal heart sound | 200 | 14 | 256 | 4.95 | 20.9 |
| #11 | Normal heart sound | 200 | 9 | 64 | 4.89 | 19.2 |
| #12 | Normal heart sound | 200 | 14 | 16 | 4.90 | 35.1 |
| #13 | Normal heart sound | 200 | 11 | 64 | 4.75 | 22.0 |
| #14 | Fourth heart sound | 200 | 13 | 256 | 4.81 | 12.1 |
| #15 | Mitral stenosis | 200 | 4 | 128 | 4.92 | 11.0 |
| #16 | Aortic stenosis | 200 | 6 | 256 | 4.83 | 8.6 |
| #17 | Aortic regurgitation | 200 | 4 | 64 | 4.88 | 10.3 |

Visual observation of the signals (Fig. 5) already showed that they included murmurs of a typically higher frequency than that of normal heart sounds. Their residuals were highly variable and the signals thus required a shorter vector dimension ($k$=20, 8, 4, 14) in the VQ operation. See records 6, 7, 8 and 9 in Table 1.

Table 2 shows the compression performance for 8 real cases (numbered from #10 to #17), 4 normal and 4 abnormal signals. The lack of soundproofing is reflected in a greater proportion of noise. Each set of data contained a sizeable residual, which made it necessary to use small vector dimensions in the VQ operations to meet the desired PRD. At the same time there was less repetition of sounds in these real cases, which somewhat lowered the CRs of the real case normal signals #10, #11, #12 and #13 to an average of 25.7. The CRs of the real case abnormal signals #14, #15, #16 and #17 were even lower: 12.1, 11, 8.6 and 10.3, respectively. To give readers a better understanding of the proposed compression of real case signals, Figs. 9 and 10 show visual plots of two examples. Although compression performance is somewhat less for the real case than for the noise-free signals, the method of compression proposed in this study is nevertheless attractive for practical applications because it performs better than any of the methods that are being used so far, as shown in Section 4.3.

## 4. Discussion

### 4.1. Selection of the training PCG signal

In the training stage a portion of the PCG signal that needs to be compressed is chosen for establishing a dictionary and codebook that are representative of the entire signal. The purpose of the dictionary and codebook is to reconstruct the heart sounds and murmurs and to extract the embedded structure in the residual of the particular PCG that is being recorded and compressed for storage or transmission. The longer the training signal the more likely it is to represent the whole PCG, and it would even be tempting to use the whole PCG as the training signal, except that such an undertaking would be very time consuming and cause an enormous computational load. From an engineering standpoint that is fortunately not necessary, since heart sounds are highly repetitive from cycle to cycle so that dictionary and codebook can

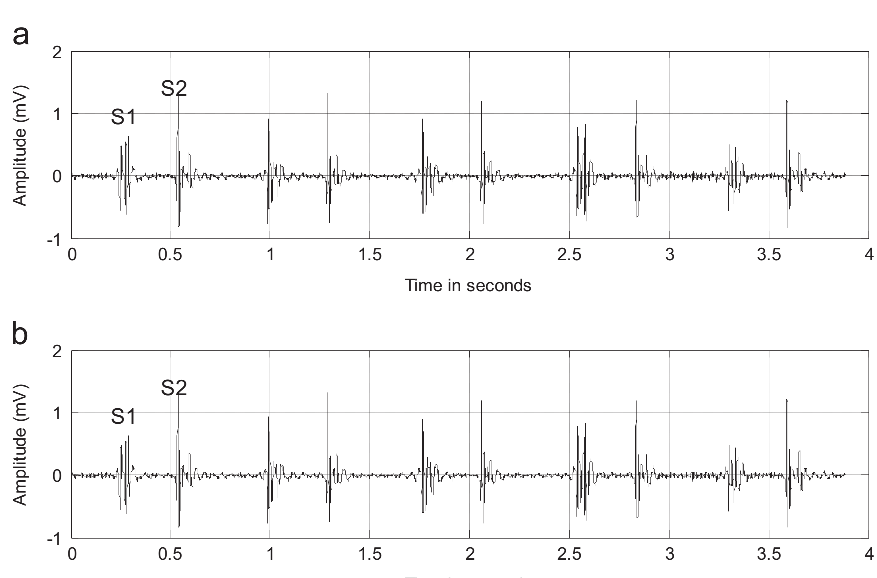


**Fig. 9.** Compression and decompression of a normal PCG signal collected in the authors' lab (PCG Record no. 13 in Table 2). (a) A portion of a normal PCG signal to be compressed. (b) The decompressed PCG signal subject to a compression ratio of 22 and a distortion degree of 4.75%.

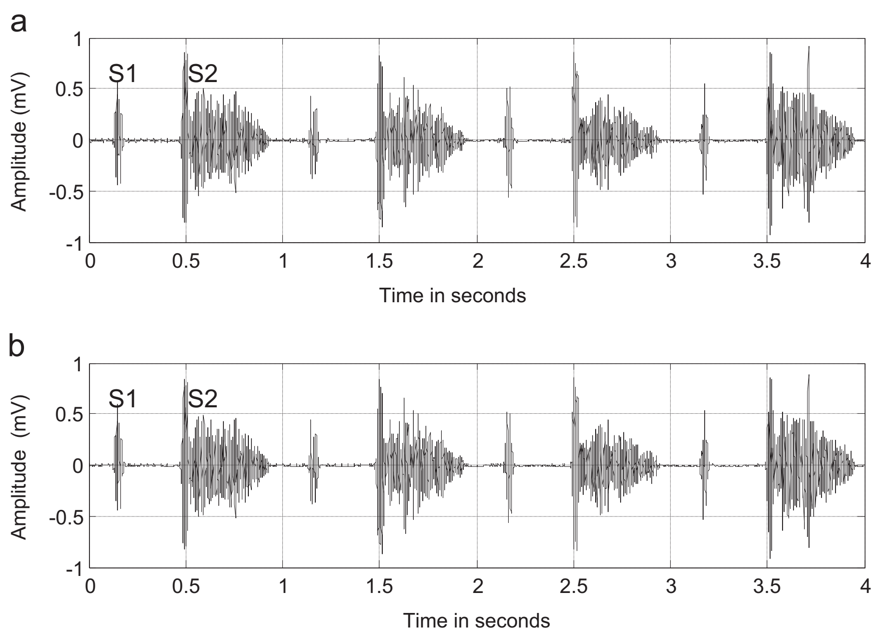


**Fig. 10.** Compression and decompression of a PCG signal with aortic regurgitation (PCG Record no. 17 in Table 2). (a) A portion of a PCG signal with aortic regurgitation to be compressed. (b) The decompressed PCG signal subject to a compression ratio of 10.3 and a distortion degree of 4.88%.

**Table 3**
The contribution of vector quantization for real case signals.

| No. | PCG record | With VQ | | | Without VQ | | | Without VQ | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | *J* | PRD(%) | CR | *J* | PRD(%) | CR | *J* | PRD(%) | CR |
| #10 | Normal heart sound | 256 | 4.95 | 20.9 | 256 | 13.8 | 36.4 | 600 | 8.1 | 18.2 |
| #11 | Normal heart sound | 256 | 4.89 | 19.2 | 256 | 10.1 | 34.0 | 600 | 7.0 | 17.5 |
| #12 | Normal heart sound | 256 | 4.90 | 35.1 | 256 | 13.6 | 49.1 | 500 | 10.2 | 28.1 |
| #13 | Normal heart sound | 256 | 4.75 | 22.0 | 256 | 18.3 | 43.0 | 700 | 9.3 | 19.8 |
| #14 | Fourth heart sound | 256 | 4.81 | 12.1 | 256 | 17.2 | 52.0 | 800 | 9.1 | 24.4 |
| #15 | Mitral stenosis | 256 | 4.92 | 11.0 | 256 | 15.9 | 34.5 | 1300 | 10.1 | 18.6 |
| #16 | Aortic stenosis | 256 | 4.83 | 8.6 | 256 | 22.5 | 17.3 | 1300 | 10.0 | 11.1 |
| #17 | Aortic regurgitation | 256 | 4.88 | 10.3 | 256 | 23.9 | 27.2 | 1300 | 10.2 | 12.4 |

#ce:italic>Note:#/ce:italic> *J* is the size of dictionary.

be reliably extracted from as few as ten cardiac cycles. Generally speaking, a selection of 10–20 contiguous cycles from the PCG signal as a training signal is sufficient to construct a qualified dictionary and codebook.

In an ideal case, the PCG signal to be compressed is steady, where the time–frequency features of the sounds or murmurs and the embedded structure in the residual do not vary over time. The dictionary and codebook extracted from a short training signal are then sufficient for rebuilding the whole PCG signal. However, the signal to be compressed may result from hours or even days of recording, long enough that the time–frequency features and the embedded structure in the random residual may in fact be subject to some variation. To track any such fluctuations and to ensure an acceptable performance it is necessary to update the dictionary and codebook at certain intervals, every 10 min with typical applications.

### 4.2. *Contribution of vector quantization*

In this study the purpose of vector quantization is to compress the residuals. Fig. 4(c) and Fig. 5(c) show that the residual is the random part of the signal outside the time–frequency model. The residual may comprise such details as heart sounds or murmurs, ambient noise or interference. If the energy of the residual is small enough to be of little consequence, then it is reasonable to discard it and omit vector quantization. However, if the residual is significant, vector quantization plays an important role in keeping the PRD low. To illustrate the contribution of vector quantization, Table 3 shows compression performance with and without vector quantization for the 8 real-case signals used in this study. Each signal was compressed in three different ways, once with VQ, dictionary size of 256 and normalized energy of the residual at 0.05 (figures shown on the left), once without VQ but with both the dictionary size and normalized energy of the residual unchanged at 256 and 0.05 (figures shown in the middle), and once without VQ but a dictionary large enough to reach a CR that would approach the figure of compression with VQ (figures shown on the right). To achieve that CR, the dictionary size was ramped up to 300 and the in discrete increments of 100, which resulted in dictionary sizes from 500 to 1300. Comparison of the two compressions without VQ shows that while the PRD can be reduced by increasing the size of the dictionary, a point is soon reached where an ever larger dictionary does not result in an appreciable further reduction of the PRD. The reason for that, in the opinion of the authors, is that the residual consists of noise and thus cannot be recovered by a dictionary, no matter what size. But it can be compressed and recovered by VQ. For each signal shown in Table 3, the PRD achieved with VQ is considerably lower than without while maintaining an acceptable CR. Vector quantization has reduced the PRD by at least half, from 10.1% to 4.89% in the case of a normal heart sound, and from 23.9% to 4.88% in the case of aortic regurgitation. We conclude that VQ has more advantage to reduce the PRD than the creation of a larger dictionary and thus becomes a viable alternative in all situations where the operator needs or wants a high CR and can tolerate a PRD as high as 5%.

### 4.3. Comparison with existing methods

Compression based on wavelet transforms (WT-based compression) is a well-established method [1]. Another possible compression method is direct vector quantization (VQ), whereby a PCG record is compressed only by vector quantization. The WT-based method, the VQ method and our proposed method were each applied to all case signals (records #1–#9 in Table 1 and records #10–#17 in Table 2). The performance comparisons are listed in Table 4.

VQ performed the worst, with the highest PRD for a given CR; its associated codebook did not sufficiently represent fast vibrating sounds and murmurs. There was thus too large a discrepancy between output vector and the associated input vector. The WT-based compression method performed somewhat better. The PCG signal can be compressed by wavelet transformation when the wavelet coefficients are concentrated in a narrow range. More than 99% of the signal's energy can then be retained in the wavelet domain by thresholding the signal to be compressed. Thus, the WT-based method of compression performs best with low-noise signals. However, the performance may worsen significantly when the noise energy increases even slightly. This shows that the PRD for WT-based compressions worsens with even a slight increase of the retained energy. The compression method proposed in this study performs considerably better. Of the three methods compared here it achieved the highest CR with comparable PRD, the best trade-off between signal compression and reliability. We can thus conclude that the superior performance of the proposed method can be attributed to (1) reduced redundancy from sound repetition and (2) compression of the residual (the random part) by vector quantization. The proposed method is also applicable to other repetitive signals, e.g., pulse, ECG and cardiac hemodynamic signals.

**Table 4**
Performance comparison among existing methods.

| No. | PCG record | Proposed | | WTC | | VQ | |
|---|---|---|---|---|---|---|---|
| | | PRD(%) | CR | PRD(%) | CR | PRD(%) | CR |
| #1 | Normal heart sound | 4.98 | 149.2 | 4.94 | 99.6 | 4.97 | 42.1 |
| #2 | Third heart sound | 4.60 | 97.1 | 4.88 | 76.3 | 4.65 | 26.7 |
| #3 | Fourth heart sound | 4.82 | 95.3 | 5.39 | 76.5 | 6.16 | 25.7 |
| #4 | Summation gallop | 4.93 | 92.4 | 4.95 | 67.1 | 4.74 | 26.6 |
| #5 | Atrial fibrillation | 4.89 | 111.9 | 4.87 | 91.2 | 5.83 | 27.7 |
| #6 | Mitral stenosis | 4.96 | 34.1 | 5.04 | 29.3 | 6.09 | 21.2 |
| #7 | Aortic stenosis | 4.95 | 19.2 | 5.37 | 17.5 | 7.64 | 11.4 |
| #8 | Pulmonary stenosis | 4.83 | 21.3 | 5.65 | 18.1 | 5.28 | 11.5 |
| #9 | Mitral regurgitation | 4.86 | 16.9 | 5.33 | 14.5 | 6.07 | 10.9 |
| #10 | Normal heart sound | 4.95 | 20.9 | 5.26 | 15.8 | 6.25 | 9.1 |
| #11 | Normal heart sound | 4.89 | 19.2 | 4.92 | 16.8 | 6.34 | 18.1 |
| #12 | Normal heart sound | 4.90 | 35.1 | 5.03 | 22.0 | 5.08 | 18.3 |
| #13 | Normal heart sound | 4.75 | 22.0 | 5.05 | 16.5 | 6.20 | 9.4 |
| #14 | Fourth heart sound | 4.81 | 12.1 | 5.09 | 10.6 | 8.19 | 9.3 |
| #15 | Mitral stenosis | 4.92 | 11.0 | 5.22 | 9.5 | 9.53 | 9.1 |
| #16 | Aortic stenosis | 4.83 | 8.6 | 4.92 | 6.8 | 10.31 | 8.6 |
| #17 | Aortic regurgitation | 4.88 | 10.3 | 5.17 | 8.9 | 11.48 | 9.4 |

### 4.4. Computational complexity

The computational complexity of the proposed method is mainly an issue in STFT operations and time-support searching. It is assumed that the length of the sliding window is $L$. The parameters are estimated from the STFT with time resolution of one sampling period. Therefore, the computational complexity of the STFT operation is $0.5{*}L^2{*}\log_2 L$. A time-support search based on the minimum energy criteria needs $K{*}L$ operations where $K$ is the number of grids for the equal partitioning of the predefined range of the time-support. Thus, $M{*}K{*}L$ operations are needed for $M$ components, and the total computational complexity of the proposed compression is $0.5{*}L^2{*}\log_2 L+M{*}K{*}L$. It is to be noted that the second term $M{*}K{*}L$ is commonly much less than $0.5{*}L^2{*}\log_2 L$, insignificant enough to be ignored. Therefore, the approximate complexity of the proposed compression is $0.5{*}L^2{*}\log_2 L$.

The computational complexity for commonly used WT methods is mainly an issue in WT decompositions, that is, WT filtering. One WT decomposition requires 2 FFT operations, 1 multiply and 1 inverse FFT. Thus, the computational complexity of the WT method is $(0.5{*}3{*}L{*}\log_2 L+L){*}P$ where $P$ is the WT decomposition level. It is close to $0.5{*}3{*}P{*}L{*}\log_2 L$ because $P{*}L$ is much less than $0.5{*}3{*}P{*}L{*}\log_2 L$.

So, the ratio of the proposed method to the WT method is $L/(3\,P)$. $L$ is 800 (0.2 s) in this study, and $P$ is in the range 4–9. Thus, the ratio is in 29–66. If the time resolution of the occurrence time reduces to 1 ms, the computational complexity of the proposed method will be one fourth of the analysis above. Then the ratio will be 7.25–16.5. Although the computational complexity of the proposed method is relatively high, the compression performance is greatly improved. It has become cheaper and cheaper to access fast computation with the rapid development of microelectronics, which makes it economical to improve compression performance even if that increases computational complexity.

## 5. Conclusions

We are proposing a novel two-stage method of compressing PCG signals for effective storage and transmission. Heart sounds and murmurs repeat from one cycle to the next and so make much of the signal redundant. In stage one the basic components of the signal are obtained by clustering and entered into a dictionary from which every sound or murmur can be reconstructed by index and occurrence time. Part of the signal, the residual, cannot be compressed and reconstructed in that way. In the second stage the residual is extracted and compressed by vector quantization. The residual may reflect ambient noise and is smaller in a regular than in an irregular heartbeat, which requires some fine-tuning of the compression parameters according to circumstance and clinical need. The proposed method allows the operator to manually preselect a target admissible degree of signal distortion and to optimize compression by reading the appropriate $k$ and $N$ values off a generated table associated with that degree of distortion. This method was evaluated by using it with various PCG signals and by comparing its results with those of the two main compression methods currently in use. Our method was shown to achieve a much better trade-off between degree of distortion and compression ratio than the two now commonly used methods.

## Conflict of interest statement

The authors claim that there is no conflict of Interest.

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Nos. 81000643 and 61471081) and Fundamental Research Funds for the Central Universities under Grant no. DUT15QY60. Some of the PCG records used in the paper were from the cardiology center at Dundee University, United Kingdom. The authors wish to thank the data collector(s). The authors thank Dr. Artur Bohnet for proofreading.

## References

[1] J. Martínez-Alajarín, R. Ruiz-Merino, Wavelet and wavelet packet compression of phonocardiograms, Electron. Lett. 40 (2004) 1040–1041.

[2] J. Martínez-Alajarín, J. Garrigós-Guerrero, R. Ruiz-Merino, Optimization of the Compression Parameters of a Phonocardiographic Telediagnosis System Using Genetic Algorithms. Bio-inspired Modeling of Cognitive Tasks, Springer, Berlin Heidelberg (2007), p. 508–517.

[3] F.J. Toledo-Moreo, A. Legaz-Cano, J.J. Martínez-Álvarez, J. Martínez-Alajarín, R. Ruiz-Merino. Compression system for the phonocardiographic Signal, in: Proceedings of International Conference on Field Programmable Logic and Applications, 2007, pp. 770–773.

[4] J. Martínez-Alajarín, J. Martínez-Rosso, R. Ruiz-Merino, Encoding technique for binary sequences using vector tree partitioning applied to compression of phonocardiographic signals, Electron. Lett. 44 (2008) 84–85.

[5] N. Boukhennoufa, K. Benmahammed, R. Benzid, Effective PCG signals compression technique using an enhanced 1-D EZW, Int. J. Adv. Sci. Technol. 48 (2012) 89–102.

[6] W. Kao, W. Chen, C. Yu, C. Hong, S. Lin, Portable real-time homecare system design with digital camera platform, IEEE Trans. Consum. Electron. 51 (2005) 1035–1041.

[7] M.S. Manikandan, K.P. Soman, S. Dandapat, Quality-driven wavelet based PCG signal coding for wireless cardiac patient monitoring, in: Proceedings of International Conference on Wireless Technologies for Humanitarian Relief, 2011, pp. 519–526.

[8] M.S. Manikandan, S. Dandapat, Wavelet energy based compression of phonocardiogram (PCG) signal for telecardiology, in: Proceedings of IET-UK International Conference on Information and Communication Technology in Electrical Sciences, Tamil Nadu, India, 2007, pp. 650–654.

[9] K. Rohoden Jaramillo, P. Ludena Gonzalez, Heart sounds compression through wavelet transform coding, in: Proceedings of the 10th Iberian Conference on Information Systems and Technologies, Aveiro, June 17–20, 2015, pp. 1–5.

[10] J.-L. Ma, M.-B. Chen, M.-C. Dong, High-fidelity data transmission of multi vital signs for distributed e-health applications, in: Proceedings of IEEE International Symposium on Bioelectronics and Bioinformatics, Chung-Li, Taiwan, 2014.

[11] W. Qin, P. Wang, A remote heart sound monitoring system based on LZSS lossless compression algorithm, in: Proceedings of the IEEE 4th International Conference on Electronics Information and Emergency Communication, 2013, pp. 109–112.

[12] X. Zhang, L.-G. Durand, L. Senhadji, H.C. Lee, J.-L. Coatrieux, Analysis-synthesis of the phonocardiogram based on the matching pursuit method, IEEE Trans. Biomed. Eng. 45 (2010) 962–971.

[13] W. Wang, Z. Guo, J. Yang, Y. Zhang, L.-G. Durand, M. Loew, Analysis of the first heart sound using the matching pursuit method, Med. Biol. Eng. Comput. 39 (2001) 644–648.

[14] T.S. Leung, P.R. White, J. Cook, W.B. Collis, E. Brown, A.P. Salmon, Analysis of the second heart sound for diagnosis of paediatric heart disease, IEE Proc. Sci. Meas. Technol. (1998) 285–290.

[15] H. Tang, T. Li, Y. Park, T. Qiu, Separation of heart sound signal from noise in joint cycle Frequency–Time–Frequency domains based on fuzzy detection, IEEE Trans. Biomed. Eng. 57 (2010) 2438–2447.

[16] J. Vesanto, E. Alhoniemi, Clustering of the self-organizing map, IEEE Trans. Neural Netw. 11 (2000) 586–600.

[17] Y. Linde, A. Buzo, R.M. Gray, An algorithm for vector quantizer design, IEEE Trans. Commun. 28 (1980) 84–95.

[18] N.M. Nasrabadi, Y. Feng, Vector quantization of images based upon the Kohonen self-organizing feature maps, in: Proceedings IEEE International Conference on Neural Networks, 1988, pp. 101–108.

[19] Cardiology, Division of Medicine and Therapeutics Ninewells Hoptital & Medical School Dundee. ⟨http://www.dundee.ac.uk/medther/Cardiology/⟩.