Noise Estimation and Reduction in Heart Sounds Using Time Frequency Block Thresholding Method

M.Vishwanath Shervegar\textsuperscript{1} and Ganesh V Bhat\textsuperscript{2}

\textsuperscript{1}Assistant Professor, E&C, MIT, Kundapura, Udupi, Karnataka  
E-mail: vishwa@hotmail.co.in, ganeshvbhat@yahoo.com  
\textsuperscript{2}Principal, CEC, Mangalore, D.K., Karnataka

Abstract - In this paper a novel method of de-noising phonocardiogram by time-frequency Overlapping Group Shrinkage method is described. In this method sigma, the standard deviation of the stationary noise present in a noisy phonocardiogram is found using activity detection. This noise is then canceled by attenuating it in the time frequency domain. The accuracy of noise reduction is measured by SNR. Overlapping Group shrinkage algorithm reduces the effect of noise by attenuating them using hard or soft thresholding. Performance of this method was found to be far better compared to other methods such as Soft Thresholding and Block Thresholding.

Keywords- Block thresholding, Activity detection, soft thresholding, overlapping group shrinkage.

I.INTRODUCTION

Phonocardiogram is a tool used by doctors to look at well-being of any person. Often this phonocardiogram is corrupted by noise from the environment. These noises are usually buzzing and humming sounds from environment, hospital sounds and other artifacts. They hinder the detection of low frequency mild sounds and lead to false detection. So enhancement of phonocardiogram along with noise reduction becomes important. Preliminary literature survey shows that there exist many noise reduction algorithms for phonocardiograms with both merits and demerits. This paper discusses time frequency Overlapping Group Shrinkage algorithm along with soft thresholding and Block Thresholding method. Section 2 discusses heart sound model. Section 3 discusses estimation of heart sound using activity detection. Section 4 discusses the heart sound reduction method using Overlapping Group Shrinkage. Section 5 discusses about the obtained results.

II.HEART SOUND MODEL

Consider a noisy heart sound signal $x$. It consists of the stationary noise $n$. The noise is a random noise with an unknown probability density function (pdf) with zero mean \cite{2}. Let the short-time Fourier transform STFT of $x$ be given by (1).

$$X_m^k = \sum_{n=0}^{L-1} x[n + mL]e^{-j\frac{2\pi km}{N}}$$ \hspace{1cm} \text{(1)}

If we consider that the STFT coefficients of $x$ are a weighted sum of samples of length $L$ of the corresponding random process, then as per central limit theorem, as $L \to \infty$, the STFT coefficients $X_m^k$ asymptotically have Gaussian pdf with zero mean \cite{2}. The pdf of the kth frequency bin $X_m^k$ can be expressed as shown in (2).

$$p(X_m^k) = \frac{1}{\pi\lambda_N(k)} e\left(-\frac{|X_m^k|^2}{\lambda_N(k)}\right)$$ \hspace{1cm} \text{(2)}

Thus, the variance of DFT of noise $\lambda_N(k)$ is equivalent to the MMSE estimation of noise power.

The signal $x$ contains heart sound for which $x = s + n$. Activity detection of Phonocardiogram compares the probabilities of presence or absence of heart sound as per the hypothesis stated in (3).

$$H_0: X_m^k = N_m^k, \text{ heart sound absent}$$

$$H_1: X_m^k = S_m^k + N_m^k, \text{heart sound present}$$ \hspace{1cm} \text{(3)}

Where $S_m^k, N_m^k$ and $X_m^k$ are K-dimensional STFT vectors of phonocardiogram (PCG), noise and noisy PCG respectively. Pdf of $x$, given $H_0$ is given by (2). Pdf of $x$, given $H_1$ is given by (4).

$$p(X_m^k) = \frac{1}{\pi(\lambda_N(k)+\lambda_S(k))} e\left(-\frac{|X_m^k|^2}{\lambda_N(k)+\lambda_S(k)}\right)$$ \hspace{1cm} \text{(4)}

Where $\lambda_S(k) = E[|N_m^k|^2]$ and $\lambda_N(k) = E[|S_m^k|^2]$ denote variance of PCG and noise variance respectively.

The likelihood ratio at the kth frequency bin is given by (5) as per the following \cite{4}.

$$\Lambda_k = \frac{p(X_k|H_1)}{p(X_k|H_0)} = \frac{1}{1+\xi_k} e\left(\frac{y_k}{1+\xi_k}\right)$$ \hspace{1cm} \text{(5)}

$\xi_k = \lambda_S(k)/\lambda_N(k)$ and $y_k = |X_m^k|^2/\lambda_N(k)$ are defined as priori and posteriori snr respectively.

III.ESTIMATION OF NOISE IN PCG

In practice, we do not have an infinite length noise sequence. The most common method of noise estimation
given a finite length noise sequence is periodogram estimation given by (6) [2].
\[ \lambda_N^m(k) = |X^k_m|^2 \]  

where \( X^k_m \) is the STFT of noise only signal \( x \) in the \( m \)th frame as defined in (1). We use Bartlett’s theorem to reduce the variance \( \lambda_N^m(k) \) by averaging the M frames.
\[ \hat{\lambda}_N(k) = \frac{1}{M} \sum_{m=1}^{M-1} \lambda_N^m(k) \]  

This method requires a length L sequence of noise only observations. \( \hat{\lambda}_N(k) \) is an unbiased and consistent estimator of \( \lambda_N(k) \): \( E[\hat{\lambda}_N(k)] = \lambda_N(k) \)  

Using Bayes rule, \( p(H_0|X^k) = \frac{p(X^k|H_0)p(H_0)}{p(X^k|H_0)p(H_0) + p(X^k|H_1)p(H_1)} = \frac{\lambda^{N_m}}{1 + \lambda^{N_m}} \)  

Where \( \epsilon = p(H_1|H_0) \) and \( N^m_k = p(X^k_m|H_1)p(X^k_m|H_0) \) is the likelihood ratio of \( m \)th frame given in (4).

Similarly, \( p(H_0|X^k) = \frac{\epsilon N^m_k}{1 + \epsilon N^m_k} \)  

If \( \beta^m_k = p(H_0|X^k) \) then \( \hat{\lambda}^m_N(k) = \beta^m_k E[|N^m_k|^2|H_1] + (1 - \beta^m_k)^2 E[|N^m_k|^2|H_0] \)  

Sohn and Sung [7] proposed that, under the hypothesis \( H_0 \), we can use the current noise observation, \( E[|N^m_k|^2|H_0] = |X^k_m|^2 \)  

Under hypothesis \( H_1 \), \( |X^k_m|^2 \) contains PCG as well as noise, and is therefore not an accurate estimate of the noise power.

Assuming that the activity detection of PCG with probability \( \beta^m_k \) has been correctly estimated in all previous frames, the best available estimate of the noise is \( E[|N^m_k|^2|H_1] = \hat{\lambda}_N^{-1}(k) \)  

From (12), (13) and (14) we have \( \hat{\lambda}^m_N(k) = \beta^m_k \hat{\lambda}_N^{-1}(k) + (1 - \beta^m_k)^2 |X^k_m|^2 \)  

Sohn and Sung [8] proposed that, if \( \beta^m_k \) is an accurate estimate of the PCG presence probability in each frame, then (15) is an equally accurate estimate of the noise power in the \( m \)th frame. Under these circumstances, \( \hat{\lambda}^m_N(k) \) takes into account all information about the underlying noise process that can be extracted from frames up to and including the current frame.

The autoregressive noise estimator \( \hat{\lambda}^m_N(k) \) proposed in Eq. (15) is optimal, if and only if the PCG presence probability estimate \( \beta^m_k \) is accurate. Unfortunately, under low SNR conditions \( \beta^m_k \) is a random variable with high variance. \( \beta^m_k \) is a sigmoid transformation of a random variable \( |X^k_m|^2 \) given by
\[ \beta^m_k = \frac{e^{-a_k|X^k_m|^2}}{e^{-a_k|X^k_m|^2} + e^{-a_k|X^k_m|^2}} \]  

threshold to sigmoid function \( \theta_k = a_k \lambda_N(k) \log(\frac{\epsilon}{\epsilon}) \) obtained by finding the value of \( |X^k_m|^2 \) at \( \beta^m_k = 0.5 \). In noise only frames where \( |N^m_k|^2 > \theta_k \) the value of \( \beta^m_k = 1 \) indicates a false positive even in the absence of PCG. Therefore, autoregressive estimator underestimates noise and overestimates PCG in any given frame. To solve this problem \( \beta^m_k \) is modelled as binary random variable-a unit step function of \( |N^m_k|^2 \). Let us define \( p = \frac{1}{\epsilon} \) (18). The parameter evaluates to \( \rho = \int_{a_k \log(\frac{\epsilon}{\epsilon})}^{\infty} e^{-t} dt = (\frac{\epsilon}{\epsilon})^{-a_k} \)  

In high noise PCG there is noise propagation error as seen in (16). If the noise process is known to be stationary, and if the first M frames of the signal are known to contain no PCG, then an \( a \) priori periodogram estimate \( \hat{\lambda}_N(k) \) of E \( [|X^k_m|^2] \) with known standard error may be computed using Eq. (6). If we assume that intervening frames provide no further information about E \( [|X^k_m|^2] \), then
\[ E[|N^m_k|^2|H_1] = \hat{\lambda}_N(k). \]  

This method does not propagate error. Instead, a false-positive frame is treated just like any other frame about which we have no certain knowledge of the noise spectrum: the noise estimate is backed off to the \( a \) priori noise estimator \( \hat{\lambda}_N(k) \) [8]. The present noise spectrum estimation method can be interpreted as a \( a \) posteriori MMSE estimate of the noise power in the current frame, when the noise process is stationary but with high variance.
IV. TIME FREQUENCY OVERLAPPING GROUP SHRINKAGE ALGORITHM

In recent years, many algorithms based on sparsity have been developed for signal de-noising. These algorithms often utilize nonlinear scalar shrinkage/thresholding functions of various forms which have been devised so as to obtain sparse representations. Examples of such functions are the hard and soft thresholding functions [8], and the nonnegative garrote [9,10]. Numerous other scalar shrinkage/thresholding functions have been derived as MAP or MMSE estimators using various probability models, e.g. [11,12,13].

For the purpose of de-noising, the regularization parameter \( \lambda \) is chosen analogous to the 'three-sigma' rule. The method allows for \( \lambda \) to be selected so as to ensure that the noise variance is reduced to a specified fraction of its original value. This method does not aim to minimize the mean square error or any other measure involving the signal to be estimated, and is thus non-Bayesian. [6] Let \( y \) be the standard deviation of the Gaussian noise in the PCG. Then, \( y \sim N(0,1) \) and let us define \( x = \text{soft}(y, T) \). Then the variance of \( x \) as a function of threshold \( T \) is given by, \( \sigma_x^2(T) = E[x^2] = \int_{|y| \geq T} (|y| - T)^2 p_y(y) \, dy = 2(1 + T^2) Q(T) - T \sqrt{\frac{2}{\pi}} e^{-\frac{T^2}{2}} (21) \), where \( p_y(y) \) is the standard normal pdf \( N(0,1) \) and \( Q(T) = \frac{1}{\sqrt{2\pi}} \int_T^\infty e^{-\frac{t^2}{2}} \, dt = 0.5 \left(1 - \text{erf} \left( \frac{T}{\sqrt{2}} \right) \right) \). [6]

Figure 1a shows that standard deviation \( \sigma_x(t) \) is a function of threshold \( T \). Soft thresholding uses 3\( \sigma \) rule to attenuate noise. The '3\( \sigma \) rule' states that nearly all values of a Gaussian random variable lie within three standard deviations of the mean (in fact, 99.7\%). Since the variance of \( x \) is unity here, the 3\( \sigma \) rule suggests setting the threshold to \( T = 3 \) which leads to \( \lambda_x(3) = 0.020 \).

The graph in Fig. 1a generalizes the 3\( \sigma \) rule: Given a specified output standard deviation \( \sigma_x \), the graph shows how to set the threshold \( T \) in the soft threshold function so as to achieve it, i.e., so that \( E[\text{soft}(y, T)^2] = \sigma_x^2 \) where \( y \sim N(0,1) \). For example, to reduce the noise standard deviation \( \sigma \) to one percent of its value, we solve \( \sigma_x(T) = 0.01 \) for \( T \) to obtain \( T = 3.36 \sigma \). This threshold is greater than that suggested by the 3\( \sigma \) rule. OGS provides the alternate solution for this type of problem. In OGS we set the regularization parameter \( \lambda \) as the threshold for PCG detection in the presence of noise. However, for OGS there is no explicit formula such as (21) relating \( \lambda \) to \( \sigma_x \). Indeed, in the overlapping group case [6], neither is it possible to reduce \( E[x^2] \) to a univariate integral as in (21) due to the coupling among the components of \( y \), nor is there an explicit formula for \( x \) in terms of \( y \), but only a numerical algorithm. Although no explicit analogue of (21) is available for OGS, the functional relationship can be found numerically. Let \( y \) be i.i.d. \( N(0,1) \) and define \( x \) as the output of the OGS algorithm: \( x = \text{ogs}(y; \lambda, K) \). The output standard deviation \( \sigma_x \) can be found by simulation as a function of \( \lambda \) for a fixed group size. For example, consider applying the OGS algorithm to a two-dimensional array \( y \) using a group size of 3X3.

For this group size, \( \sigma_x \) as a function of \( \lambda \) is illustrated in Fig. 1b. The graph is obtained by generating a large two-dimensional array of i.i.d. standard normal random variables, applying the OGS algorithm for a discrete set of \( \lambda \), and then computing the standard deviation of the result for each \( \lambda \). Once this graph is numerically obtained, it provides a straight forward way to set \( \lambda \) so as to reduce the noise to a specified level. For example, to reduce the noise standard deviation \( \lambda \) down to one percent of its value, we should use \( \lambda = 0.43 \sigma \) in the OGS algorithm according to the graph in Fig. 1b. It can be observed in Fig. 1 that the function \( \sigma_x(\lambda) \) has a sharper 'knee' in the case of OGS compared with soft thresholding.

Graphs for numerous group sizes show that in general the larger the group, the sharper is the knee. Note that in practice \( \lambda \) should be chosen large enough to reduce the noise to a sufficiently negligible level, yet no larger so as to avoid unnecessary signal distortion. That is, suitable values of \( \lambda \) are somewhat near the knee. Therefore, due to the sharper knee, the de-noising process is more sensitive to \( \lambda \) for larger group sizes; hence, the choice of \( \lambda \) is more critical. Similarly, it can be observed in Fig. 1 that for OGS, the function \( \sigma_x(\lambda) \) follows a linear approximation more closely to the left of the 'knee' than it does in the case of soft thresholding.

The preceding sections described how the parameter \( \lambda \) may be chosen so as to reduce additive white Gaussian noise to a desired level. However, in many cases the noise is not white. For example, in the PCG de-noising example where the OGS algorithm is applied directly in the STFT domain. However, the STFT is an over-complete transform; therefore, the noise in the STFT domain will not be white, even if it is white in the original signal domain. In the PCG de-noising the noise is more highly correlated, the values of \( \lambda \) will be somewhat inaccurate.

The penalty function (22) is suitable for stationary noise; however, in many applications, noise is not stationary. For example, in the problem of de-noising PCG corrupted by stationary coloured noise, the Variance of the noise in the STFT domain will vary as a function of frequency. In particular, some noise components may be narrowband and therefore occupy a narrow time-frequency region. The OGS penalty function and algorithm, as described in this paper, do not apply to this problem directly. The penalty function (4) and the process to select \( \lambda \) must be appropriately modified. The OGS algorithm as described above uses the same block size over the entire signal. In some applications, it may be more appropriate that the block size varies. For
example, in PCG de-noising, as noted and developed, it is beneficial that the block size in the STFT domain varies as a function of frequency (e.g., for higher temporal resolution at higher frequency). This problem is solved using Block thresholding algorithm.

![Fig 1a. standard deviation vs threshold](image)

**Fig 1a. standard deviation vs threshold**

![Fig 1b Overlapping group shrinkage (OGS) with group size 3 X 3](image)

**Fig 1b Overlapping group shrinkage (OGS) with group size 3 X 3**

### V. TIME FREQUENCY BLOCK THRESHOLDING ALGORITHM

A time frequency block estimator regularises the power subtraction by calculating a single attenuation factor. The time frequency plane is divided into I blocks $B_i$ with arbitrary shape. For each $B_i$ a single estimator $\hat{f}$ is calculated with constant attenuation $a_i$ for the noisy signal. The noise characteristics are changed during the passage from time field to time-frequency field. It is still Gaussian (for all frequencies, the noise follows a centred normal law) but $\sigma^2$ changes. Consider the discrete Fourier transform of the windowed noise. The Fourier coefficients of the noise is given by $\hat{\eta}_k = \frac{1}{\sqrt{W}} \sum_n w_n \eta_n \exp\left(\frac{-2\pi i kn}{W}\right)$ (23). $\text{Var} (\hat{\eta}_k) = \frac{1}{W} \text{var} \left( \sum_n w(n) \eta_n \exp\left(\frac{-2\pi i kn}{W}\right) \right) = 0.375 \sigma^2$. (24)

The coefficients matrix is partitioned into macro-blocks and as the signal is real. This matrix has a symmetry between negative and positive frequencies thus it is enough to only treat the negative frequencies. The frequency 0 is treated separately. For the zero frequencies, we treat the points from the beginning to the end eight by eight (blocks 1x8): attenuation coefficient $a_i$ is $a_i = 1 - \frac{1}{\xi_i+1}$ with $\xi_i = \frac{\hat{\eta}_i^2}{\sigma_i^2}$. Where $\hat{\eta}_i^2$ is the empirical mean on the block $i$. The real $\lambda$ is
a parameter depending on the block size. The real $\lambda$ controls the variance term which is due to the noise variation. It is computed with the following expression: $P(\hat{e}^2 > \lambda \sigma^2) < \delta$. In this expression, $\delta$ is a parameter such as, with $\delta = 10^{-3}$, musical noises are barely audible. The blocks inside macro-the block as degree of freedom. Due to discretization effects, $\lambda$ takes roughly the same values for $W_i = 1$ and $W_i = 2$. So, to compute $\lambda$ for $W_i = 1$, we are doing the same as if $W_i = 2$. The following matrix gives the computed values of blocks are rectangles. Their sizes are $L_i \times W_i$ where $L_i$ and $W_i$ are respectively the length in time and the block width in frequency. The smallest rectangle has the size 1x2, 1 in frequency and 2 in times. With $k = 1$ (the redundancy factor), $\hat{e}^2$ is following a $\chi^2$ distribution with the size of $\lambda$ for different size of blocks (computed thanks to table I):

$$M = 1.8 \quad 2.0 \quad 2.5 \quad 2.5 \quad 2 \quad 2.5 \quad 3.5 \quad 3.5 \quad 4.7 \quad 4.7$$

<table>
<thead>
<tr>
<th>$B_i$</th>
<th>4</th>
<th>8</th>
<th>16</th>
<th>32</th>
<th>64</th>
<th>128</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda$</td>
<td>4.7</td>
<td>3.5</td>
<td>2.5</td>
<td>2.0</td>
<td>1.8</td>
<td>1.5</td>
</tr>
</tbody>
</table>

Even if an upper bound of the risk can be found, then it cannot be computed while the signal $f$ is unknown. That is why we use an estimator of the risk which is found with the SURE theorem. This theorem is used to find the best block shapes into a macro-block by minimising this estimated risk. This is the SURE (Stein Unbiased Risk Estimate) theorem: Let $Y$ be the noisy signal. It's a normal random vector with the identity as covariance matrix and of expectation $F$, which is the signal searched, without noise.

$$R_k = \sum E[(F(i,j) - a_k F(i,j))^2] = \sum E[(F(i,j) - Y(i,j) - hY(i,j))^2]$$ (26)

Finally, the blocks are chosen to minimize $R_k$. A macroblock is 8 points in time (horizontally) and 16 points in frequency (vertically). The beginning is time 1 and $R_k = \sigma^2(B_k + \frac{\alpha^2}{\sigma} 1_{Y \leq \lambda \sigma^2} + B_k (Y \sigma - 2) 1_{Y > \lambda \sigma^2})$ (27).

This formula gives the estimation of the risk of the block $i$ of size $B_i$. It is obtained using the SURE theorem with: $p=B_i$, $h(Y_i) = (a_i - 1) Y_i$. For a given subdivision, the estimation of the risk of the macro-block, is the sum of the risk estimations of each block of the subdivision. All the 15 subdivisions are tested. The one with the minimal risk estimation is chosen. The attenuation coefficients are computed in the same way as for the zero frequency (formula (1)). For the last blocks, which are not full in frequency, all the coefficients of each block are treated together like for the zero frequency. For the last few coefficients that do not make a block, do hard thresholding. For positive frequencies, conjugate from the negative frequencies.

VI. RESULTS AND DISCUSSION

The experiments presented below have been performed on various types of PCG signals obtained from Peter Bentley’s PCG database [1]. The sounds are wav files sampled at 44.1 kHz. They were corrupted by Gaussian noise of different amplitude. For each sound, de-noising with maximum noise removal were applied. The noise power was estimated using activity detection. [2] The database is a mixture of normal and abnormal PCG along with clicks and murmurs. Three methods were used for comparison namely soft thresholding (ST) [6][8], Overlapping group shrinkage (OGS) [6] and Block Thresholding.

Figure 2a shows a noisy heart sound 201108222231 from the database. [1]. Figure 2b illustrates the STFT of the above sound calculated with 50% overlapping blocks of length of 512 samples. A well-known problem arising in many audio enhancement algorithms is that the residual noise is audible as ‘musical noise’ [14] [15]. Musical noise may be attributed to isolated noise peaks in the time-frequency domain that remain after processing. Figure 2c illustrates the STFT obtained by soft thresholding the noisy STFT, with threshold T. T is selected such that as the noise standard deviation reduces down to 0.1% of its value. T = 3.26$\sigma$, where $\sigma$ is the noise standard deviation of the noise in the STFT domain. The noise is sufficiently suppressed and the musical noise is clearly inaudible; however, the signal is distorted due to the relatively high threshold that is used. This is evident from sSNR of 3.07 dB and SNR of 10.91 dB from tables 2 and 3.
Figure 2a shows a noisy heart sound 201108222231 from the database. [1]. Figure 2b illustrates the STFT of the above sound calculated with 50% overlapping blocks of length of 512 samples. A well-known problem arising in many audio enhancement algorithms is that the residual noise is audible as ‘musical noise’ [14] [15]. Musical noise may be attributed to isolated noise peaks in the time-frequency domain that remain after processing. Figure 2c illustrates the STFT obtained by soft thresholding the noisy STFT, with threshold T. T is selected such that as the noise standard deviation reduces down to 0.1% of its value. T = 3.26σ, where σ is the noise standard deviation of the noise in the STFT domain. The noise is sufficiently suppressed and the musical noise is clearly inaudible; however, the signal is distorted due to the relatively high threshold that is used. This is evident from sSNR of 3.07 dB and SNR of 10.91 dB from tables 2 and 3.

Fig. 2c is overly thinned. Methods to reduce musical noise includes over estimating the noise variance, imposing a minimum spectral noise floor [16], and improving the estimation of model parameters [13]. To avoid isolated spurious time-frequency noise spikes (to avoid musical noise), the grouping/clustering behaviour of STFT coefficients of PCG waveforms can be taken into account. To this end, a recent algorithm by Yu et al. [5] for speech/audio enhancement consists of time-frequency block thresholding.

We note that the algorithm [5] is based on non-overlapping blocks. Similar to it [5], the OGS algorithm aims to draw on the grouping behaviour of STFT coefficients so as to improve the overall de-noising result, but it uses a model based on fully overlapping blocks. Figure 2d illustrates the result of block thresholding [3] using the software provided by the authors. It can be seen that block thresholding (BT) produces blocking artefacts in the spectrogram. Figure 2d illustrates the result of Overlapping Group Shrinkage (OGS) applied to the noisy STFT. 25 iterations of the OGS algorithm were used. Based on listening to HS audio signals de-noised with various group sizes, a group size 8 * 2 (i.e., eight frequency bins * two time bins) was chosen. Other group sizes may be more appropriate for other sampling rates and STFT block lengths. As in the soft thresholding experiment, the parameter λ was selected so as to reduce the noise standard deviation down to 0.1% of its value. Regularization parameter λ was fixed as per λ = 0.32σ. While the sSNR 3.9 dB and SNR 11.9 dB is lower than block thresholding (sSNR 11.81 and SNR 18.63), the artefacts of the OGS de-noised PCG are less audible and musical noise is not audible. This was clearly evident from figure 2e (OGS) and 3d (BT). It was found in [16] that empirical Wiener post-processing (EWP), introduced in [6], improves the result of the block thresholding (BT) algorithm. This post-processing, which is computationally very simple, improves the result of OGS by an even greater degree than for BT, as measured by SNR improvement. The Wiener post-processing raises the SNR for BT from 15.35 dB to 15.75 dB, while it raises the SNR for OGS from 11.9 dB to 16.9 dB while for BT it goes from 18.63 dB to 20.0 dB. Hence, the two methods give almost the same SNR after Wiener post-processing. The substantial SNR improvement in the case of OGS can be explained as follows: the OGS algorithm has the effect of slightly shrinking (attenuating) large coefficients which produces a bias and negatively affects the SNR of the de-noised signal. The Wiener post-processing procedure largely corrects that bias. It has the effect of rescaling (slightly amplifying) the large coefficients appropriately.

<table>
<thead>
<tr>
<th>Sounds &amp; SNR (dB)</th>
<th>ST</th>
<th>(OGS-no wiener filtering)</th>
<th>(OGS-wiener filtering)</th>
<th>(BT-no wiener filtering)</th>
<th>(BT-wiener filtering)</th>
</tr>
</thead>
<tbody>
<tr>
<td>201108222231 (-4.24)</td>
<td>10.91</td>
<td>11.9</td>
<td>16.19</td>
<td>18.63</td>
<td>20.0</td>
</tr>
<tr>
<td>201108222232 (-9.58)</td>
<td>6.25</td>
<td>7.01</td>
<td>10.43</td>
<td>13.8</td>
<td>15.22</td>
</tr>
<tr>
<td>201108222242 (-9.91)</td>
<td>6.15</td>
<td>7.47</td>
<td>12.5</td>
<td>15.39</td>
<td>17.29</td>
</tr>
<tr>
<td>201108222257 (-10.33)</td>
<td>7.97</td>
<td>8.21</td>
<td>10.76</td>
<td>13.47</td>
<td>14.75</td>
</tr>
<tr>
<td>201108222258 (-3.08)</td>
<td>11.44</td>
<td>11.87</td>
<td>15.96</td>
<td>19.72</td>
<td>21.62</td>
</tr>
<tr>
<td>201108222243 (5.12)</td>
<td>17.3</td>
<td>19.78</td>
<td>27.63</td>
<td>18.59</td>
<td>18.68</td>
</tr>
<tr>
<td>201108222256 (-14.91)</td>
<td>6.48</td>
<td>6.58</td>
<td>8.49</td>
<td>11.56</td>
<td>12.55</td>
</tr>
</tbody>
</table>
### TABLE III SSNR OF SOUNDS FOR DIFFERENT METHODS

<table>
<thead>
<tr>
<th>Sounds &amp; sSNR (dB)</th>
<th>ST</th>
<th>OGS-no wiener filtering</th>
<th>OGS-wiener filtering</th>
<th>BT-no wiener filtering</th>
<th>BT-wiener filtering</th>
</tr>
</thead>
<tbody>
<tr>
<td>201108222231 (-8.81)</td>
<td>3.07</td>
<td>3.5</td>
<td>7.36</td>
<td>11.81</td>
<td>12.93</td>
</tr>
<tr>
<td>201108222232 (-9.61)</td>
<td>1.73</td>
<td>2.11</td>
<td>4.6</td>
<td>7.51</td>
<td>8.63</td>
</tr>
<tr>
<td>201108222242 (-9.62)</td>
<td>1.93</td>
<td>2.45</td>
<td>5.57</td>
<td>7.9</td>
<td>9.22</td>
</tr>
<tr>
<td>201108222257 (-9.66)</td>
<td>1.1</td>
<td>1.16</td>
<td>2.45</td>
<td>4.48</td>
<td>5.33</td>
</tr>
<tr>
<td>201108222258 (-8.89)</td>
<td>3.46</td>
<td>3.7</td>
<td>8.27</td>
<td>11.93</td>
<td>13.72</td>
</tr>
<tr>
<td>201108222243 (-4.56)</td>
<td>11.73</td>
<td>13.66</td>
<td>26.64</td>
<td>27.31</td>
<td>29.68</td>
</tr>
<tr>
<td>201108222256 (-9.87)</td>
<td>0.47</td>
<td>0.48</td>
<td>1.04</td>
<td>1.86</td>
<td>2.27</td>
</tr>
</tbody>
</table>

**Fig. 2a** noisy heart PCG 20110822231

**Fig. 2b** STFT of noisy signal
Noise Estimation and Reduction in Heart Sounds Using Time Frequency Block Thresholding Method

Fig. 2c Soft thresholded PCG

Fig. 2d Overlapping Group Shrinkage PCG

Fig. 2e BT PCG
VII. CONCLUSION

From the above work it is clear that both the methods Block thresholding algorithm [3] and OGS algorithm produces de-noised signal with high SNR and sSNR. It is very evident that the so called time frequency structures namely musical noise rarely reoccur in BT and OGS methods. In case of BT algorithm, the SNR and sSNR are quite higher and the sounds are much louder with no artefacts as compared to OGS algorithm. Hence OGS algorithm is the recommended method for de-noising PCG signals especially in hospitals where background noise is a major hindrance in sound acquisition.

REFERENCES

[2] Bowon Lee Hewlett-Packard Laboratories 1501 Page Mill Rd.Palo Alto, CA 94304 bowon.lee@hp.com, Mark Hasegawa-Johnson University of Illinois at Urbana-Champaign Electrical and Computer Engineering 405 N. Mathews Ave., Urbana, IL 61801 jhasegaw@uiuc.edu, MINIMUM MEAN-SQUARED ERROR A POSTERIORI ESTIMATION OF HIGH VARIANCE VEHICULAR NOISE


[6] Po-Yu Chen and Ivan W. Selesnick, Polytechnic Institute of New York University, 6 Metrotech Center, Brooklyn, NY 11201, USA. Email: poyupaulchen@gmail.com, selesi@poly.edu. Tel: +1 718 260-3416. Translation-Invariant Shrinkage/Thresholding of Group Sparse Signals.


