De-noising of acoustic breathing signals

René Derkx* and Harm Belt†
*Digital Signal Processing group, Philips Research, High Tech Campus 36, 5656 AE, Eindhoven, The Netherlands
Email: rene.derkx@philips.com
†Video and Image Processing, Philips Research

***Abstract*—For unobtrusive capturing of acoustic breathing signals during sleep, a microphone can be placed in the vicinity of the person. As breathing signals are generally very weak compared to the intrinsic noise of the microphone, the resulting signal-to-noise ratio (SNR) is low. We present a de-noising technique for the enhancement of acoustic breathing signals captured with a microphone. As the intrinsic noise of the microphone is stationary, we can use standard spectral subtraction schemes with stationary noise-floor estimators. For bad SNR conditions however, these techniques suffer from musical tones. To remove these musical tones, we apply median filtering in the standard spectral subtraction scheme. Furthermore, an alternative solution is proposed that has a very low computational complexity.**

I. Introduction

When unobtrusively recording breathing signals with a microphone during sleep, the microphone can be placed in the vicinity of the breathing person, say approximately 50 cm away from the person. In Fig. 1, this setup is shown.

Fig. 1. Recording of breathing via a microphone.

As the acoustic energy from the breathing is generally very weak, the signal-to-noise ratio (i.e. the ratio between the breathing signal and the noise) can be very poor, making it difficult to extract the relevant respiratory parameters, like the respiratory-rate, from the signal. The signal-to-noise ratio (SNR) of the microphone is determined by two types of noises:

- Intrinsic sensor-noise; generated by air-particles on the membrane of the microphone and $1/f$ noise introduced by the pre-amplifier (e.g. a FET) in the microphone,
- Acoustic noise; generated by external sources in the environment, transmitted via the acoustic paths and captured by the membrane of the microphone.

The first type of noise can be reduced by using a microphone with lower intrinsic noise (self-noise). Cheap microphones have self-noise of roughly 34 dBA SPL, while expensive (low-noise) microphone can have self-noise of roughly 14 dBA SPL.

The second type of noise can be reduced by using directional microphones. Most of such microphones have a first-order directive beam-shape, enabling a diffuse noise reduction of maximally 6 dB (hyper-cardioid) [1]. However, it should be noted that this directionality often leads to a degradation in intrinsic noise of the microphone (also known as noise-boost [1]), because an extra amplification is required in the lower-frequency range to maintain a flat frequency spectrum.

In this paper, the aim is to improve the signal-to-noise ratio (to improve the automatic classification of breathing signals) without using a better microphone (i.e. lower intrinsic noise or improved spatial directivity) and without placing the microphone closer to the breathing person. We apply de-noising based on spectral subtraction techniques known from the speech enhancement area [2].

We assume that there is only stationary noise present in the microphone signal, caused by the intrinsic noise of the microphone. For a breathing person in a sleep situation this is a valid assumption, as it can be expected that the undesired acoustic noise from the environment is relatively low.

II. Spectral subtraction

We start by sampling the microphone signal with a sampling-frequency of $F_s = 8$ kHz, which is sufficient for preserving the relevant spectral information of the breathing sounds. We apply the spectral-subtraction method (known from the speech enhancement area) to reduce the stationary noise [2]. The basic scheme (spectral analysis, modification, synthesis) is shown in Fig. 2.

The samples of the discrete time-signal $x(k)$ with time-index k are first concatenated to blocks of B samples. Then, a block of M samples is constructed by concatenating the current B samples with B samples from the previous block[1]. The M samples are windowed with a raised cosine window and converted to the frequency-domain via an FFT operation, resulting in a complex-spectrum $X(\kappa, \omega)$, $\omega = [0; M-1]$ with κ the block-index. As the spectrum originates from real-valued input-data, $X(\kappa, \omega)$ is a two-sided spectrum having complex-conjugate symmetry (Hermitian).

Next, the complex-spectrum $X(\kappa, \omega)$ is converted to a magnitude spectrum. Based on this magnitude spectrum the spectral noise-floor is estimated. As we will assume that the noise in the microphone signal $x(k)$ is mainly originating from the intrinsic sensor-noise, we can assume that this noise is

[1]Here we use $M = 2B$ which is a 50% overlap situation.

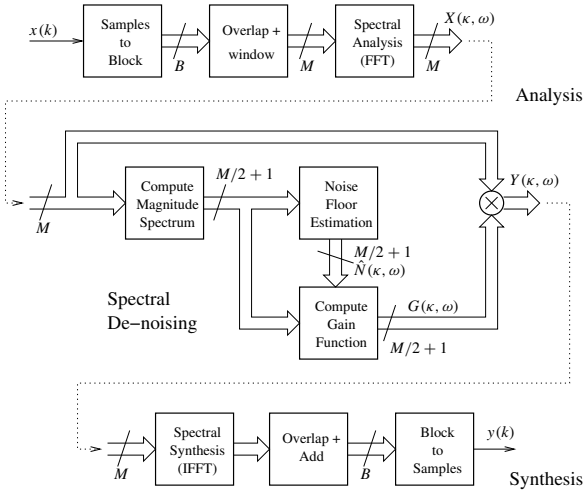

Fig. 2. Basic spectral subtraction.

long-term stationary. Hence the noise-floor can be determined by tracking of the spectral minimum [3] over a long period of time (say 10 seconds or more). This spectral noise-floor estimation will be indicated with $\hat{N}(\kappa, \omega)$, with $\omega = [0; M/2]$. The bins with $\omega = [M/2 + 1; M - 1]$ are redundant because the spectrum is symmetrical. With the noise-floor estimation, a gain-function [2] is computed which has spectral-values in the range between 0 and 1. The gain-function $G(\kappa, \omega)$ is computed for $0 \leq \omega \leq M/2$ as:

$$G(\kappa, \omega) = \max \left\{ \frac{|X(\kappa, \omega)| - \gamma \hat{N}(\kappa, \omega)}{|X(\kappa, \omega)|}, 0 \right\}, \qquad (1)$$

where the maximum operation is required to ensure positive values for the gain-function and γ is a parameter that enables so-called over-subtraction when choosing $\gamma > 1$. This is usually required to improve the amount of noise-reduction (at the cost of more distortion of the desired signal).

Next, $X(\kappa, \omega)$ is multiplied by this gain-function[2]:

$$Y(\kappa, \omega) = \begin{cases} X(\kappa, \omega)\, G(\kappa, \omega) & \text{for:} \quad 0 \leq \omega \leq M/2 \\ X(\kappa, \omega)\, G(\kappa, M - \omega) & \text{for:} \quad M/2 < \omega < M. \end{cases} \qquad (2)$$

The resulting complex spectrum $Y(\kappa, \omega)$ is now converted to time-domain via an IFFT. Finally, the overlap-add procedure is applied to obtain the output-signal $y(k)$ [2].

It is known from speech-enhancement that enhancing signals with a bad SNR via the spectral subtraction method gives rise to musical tones (also known as musical noise) that have a highly stochastic character in both time- and frequency [2]. Sometimes time- and/or frequency-averaging of the gain-function is applied to reduce these musical tones. Also median-filtering is sometimes used [4]. For speech signals however, too much averaging leads to distortion of the speech signal, as the quasi-stationarity of speech is around a few tens of milliseconds and speech can have vowels with a formant structure having lots of peaks and valleys.

We captured a breathing signal by using a microphone (AKG CK31) with intrinsic noise level of 20 dBA SPL, placed at approximately 50 cm away from the breathing person while asleep. If we look at the de-noising of this breathing signal by the regular spectral subtraction scheme as discussed above, we get the time-frequency spectrum for the original and de-noised breathing fragment[3] as shown in Fig. 3. Here, we used $B = 256$ as block-size and $F_s = 8$ kHz. For obtaining the result of the de-noising algorithm shown in Fig. 3 (b), we used a *very large* over-subtraction factor $\gamma = 8$ to minimize musical tones which are nevertheless still visible in Fig. 3 (b).

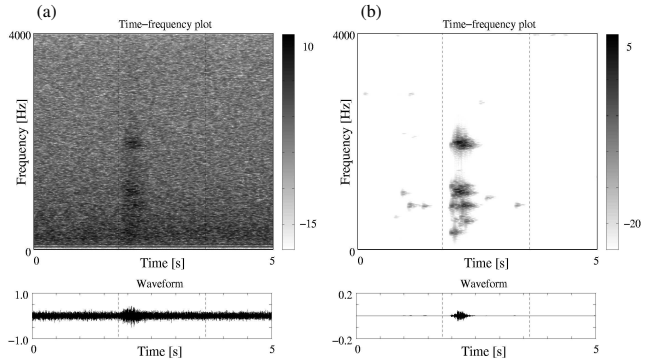

Fig. 3. Original time-frequency spectrum (a) and de-noised time-frequency spectrum (b) of a breathing fragment during sleep.

The SNR improvement is limited due to the bad SNR of the original fragment. Even by using a large over-subtraction, musical tones are still present in the time-frequency spectrogram area where there is no breathing event (before and after the dashed lines in Fig. 3). As the musical tones can negatively influence the breathing event detection we focus on the suppression of musical tones. Although we are aware of the fact that there are many methods to remove musical tones, we focus on the median filtering in the time-frequency plane, as shown in the remainder of this paper.

III. Modified spectral subtraction

To reduce musical tones in the regular spectral subtraction scheme, we exploit specific characteristics of breathing signals. Most breathing-signals are quasi-stationary over a relatively long time-inverval (several hundreds of milliseconds) and have a broad and spectrally filled frequency spectrum. In other words: the breathing increases and decreases very slowly over time and also the frequency-content does not show peaks and valleys like in vowels of speech. Therefore, we can apply averaging techniques and median-filters over a large time-span, but also over a large frequency range. In this section, we will extend the spectral subtraction scheme (see Section II) with 2D

[2]Only a single-sided spectrum has to be computed in practical realizations, due to the complex-conjugate symmetry.

[3]This breathing fragment was recorded during sleep and is actually an ex-hale fragment. It was found that for normal breathing sounds (i.e. no snoring or wheezing sounds), mainly ex-hales are present in the acoustic breathing signal during a full night sleep.

median filtering, where the time-frequency kernels are chosen in such a way that musical tones are reduced sufficiently, while still preserving the breathing signal.

In the modified de-noising scheme, a modified gain-function $\tilde{G}(\kappa,\omega)$ is computed based on the gain-function data $G(\kappa,\omega)$ and a kernel of size $T\times F$, with T and F being the time-order and frequency-order respectively:

$$\tilde{G}(\kappa,\omega)=\begin{cases}\text{med}\{\mathbf{G}(\kappa,\omega)\} & \text{if: } \Omega\le\omega\le\frac{M}{2}-F+\Omega+1,\\ 0 & \text{otherwise}\end{cases} \tag{3}$$

with:

$$\Omega=\Big\lfloor\frac{F}{2}\Big\rfloor, \tag{4}$$

where the matrix $\mathbf{G}(\kappa,\omega)$ is constructed as:

$$\mathbf{G}(\kappa,\omega)=\begin{pmatrix} G(\kappa-T+1,\omega_0) & \dots & G(\kappa,\omega_0)\\ \vdots & \ddots & \vdots\\ G(\kappa-T+1,\omega_0+F-1) & \dots & G(\kappa,\omega_0+F-1)\end{pmatrix},$$

with:

$$\omega_0=\omega-\Omega, \tag{5}$$

and where med{$\mathbf{A}$} outputs the median value of all matrix elements of matrix $\mathbf{A}$. This involves sorting a list of all array elements and the middle element of this sorted list is taken as output-value. It is noted that at the boundaries of the gain-function we will not compute the median and set the modified gain-function value $\tilde{G}(\kappa,\omega)$ to zero.

By taking the median of the gain-values within a certain kernel, we can intuitively understand that the outliers (musical tones) are removed when the number of outliers is less than half of the kernel-size. This is generally satisfied when choosing $\gamma>1$.

The optimal kernel size will also depend on the signal (breathing) type. For example, normal breathing sounds require different kernel sizes compared to wheezing or snoring sounds, which have more peaks and valleys in the spectrum.

The computational complexity of the median filtering is very large. To sort a list of N elements, we require a complexity in the order of $N\log N$. This means that for every block-iteration, the median filtering requires a computational complexity in the order of $T\cdot F\cdot(M/2-F+2)\log(T\cdot F)$. Therefore, we seek for an algorithm that has a lower computational complexity.

IV. Low-cost solution

In this section, we present a low-cost solution for the modified spectral subtraction as discussed in Section III. The low-cost solution consists of three basic steps:

- Binary quantization of gain-function values $G(\kappa,\omega)$,
- Compute kernel-area integral via the summed area table,
- Compute median via the majority operator.

First, the gain-function values $G(\kappa,\omega)$ are quantized as:

$$G_q(\kappa,\omega)=\begin{cases}0 & \text{if: } G(\kappa,\omega)=0\\ 1 & \text{if: } G(\kappa,\omega)>0.\end{cases} \tag{6}$$

Next, we compute the kernel area (integral) of the quantized gain-function:

$$I_q(\kappa,\omega)=\sum_{i=0}^{i=T-1}\sum_{j=0}^{j=F-1}\big[\mathbf{G}_q(\kappa,\omega)\big]_{i,j}, \tag{7}$$

with the matrix $\mathbf{G}_q(\kappa,\omega)$ defined similar to $\mathbf{G}(\kappa,\omega)$ and $[\mathbf{G}(\kappa,\omega)]_{i,j}$ indicates the i'th column-index and the j'th row-index of this matrix.

As this straightforward computation of the integral still has a quite large computational complexity, we will propose to compute the integral by means of the so-called summed area table [5]. This method is also known as the integral-image, because of the use of this technique in picture processing. The (circular) summed area table in our case will be denoted by matrix $\mathbf{S}(\kappa)$ and has a dimension of $(T+1)\times(M/2+1)$. The values of the summed area table $\mathbf{S}(\kappa)$ are computed as:

$$\big[\mathbf{S}(\kappa)\big]_{\kappa\,(\mathrm{mod}\,T+1),\omega}=\sum_{i=0}^{\kappa}\sum_{j=0}^{\omega}G_q(i,j), \tag{8}$$

where $[\mathbf{S}(\kappa)]_{i,j}$ indicates the element of matrix $\mathbf{S}(\kappa)$ with column-index i and row-index j.

We can compute the summed area table efficiently [5] by starting with an empty summed area table:

$$\big[\mathbf{S}(0)\big]_{i,j}=0,\qquad \text{for: } 0\le i\le T \text{ and: } 0\le j\le M/2 \tag{9}$$

and updating the contents of the summed area table for each iteration κ by a single pass from $\omega=0$ to $\omega=M/2$ and computing:

$$[\mathbf{S}(\kappa)]_{\kappa\,(\mathrm{mod}\,T+1),\omega}=G_q(\kappa,\omega)+[\mathbf{S}(\kappa)]_{(\kappa-1)\,(\mathrm{mod}\,T+1),\omega}+\begin{cases}[\mathbf{S}(\kappa)]_{\kappa\,(\mathrm{mod}\,T+1),\omega-1}\\ -[\mathbf{S}(\kappa)]_{(\kappa-1)\,(\mathrm{mod}\,T+1),\omega-1} & \text{if: } \omega>0\\ 0 & \text{otherwise.}\end{cases} \tag{10}$$

As the values in the summed area table will increase over time, we apply a modulo operation on these values and compute through the overflow [6]. As all values in the table are either 0 or 1, we know that the integral can have maximum values up to $(T+1)\cdot(M/2+1)$. Therefore, we store the values of the summed area table modulo X where $2^n=X\ge(T+1)\cdot(M/2+1)$, with n integer.

The integral is computed by only accessing four values from the summed area table:

$$I_q(\kappa,\omega)=\begin{cases}S & \text{if: } \Omega\le\omega\le\frac{M}{2}-F+\Omega+1,\\ 0 & \text{otherwise}\end{cases} \tag{11}$$

with:

$$S=[\mathbf{S}(\kappa)]_{\kappa\,(\mathrm{mod}\,T+1),\omega_0+F-1}-[\mathbf{S}(\kappa)]_{(\kappa+1)\,(\mathrm{mod}\,T+1),\omega_0+F-1}+\begin{cases}[\mathbf{S}(\kappa)]_{(\kappa+1)\,(\mathrm{mod}\,T+1),\omega_0-1}\\ -[\mathbf{S}(\kappa)]_{\kappa\,(\mathrm{mod}\,T+1),\omega_0-1} & \text{if: } \omega_0>0\\ 0 & \text{otherwise.}\end{cases} \tag{12}$$

The low-cost solution for the median filtering to obtain the modified gain-function is now given by computing the so-called majority function [7] (also known as median operator):

$$\tilde{G}_q(\kappa,\omega)=\Big\lfloor\frac{1}{2}+\frac{I_q(\kappa,\omega)-1/2}{T\,F}\Big\rfloor, \tag{13}$$

where:

$$\tilde{G}_q(\kappa, \omega) \in \{0, 1\}. \tag{14}$$

As the majority function only outputs values of 0 or 1, the resulting gain function $\tilde{G}_q(\kappa, \omega)$ will also be quantized to these levels. Therefore, an alternative is to let the modified gain-function also have values between 0 and 1 when the majority function outputs 1. This computation of the modified gain-function is given by:

$$\tilde{G}_c(\kappa, \omega) = \frac{2}{T \cdot F} \cdot \max\left\{ I_q(\kappa, \omega) - \frac{T \cdot F}{2}, 0 \right\}, \tag{15}$$

where:

$$\tilde{G}_c(\kappa, \omega) \in [0, 1]. \tag{16}$$

Application of Eq. (15) leads to a more gradual behaviour of the modified gain-function $\tilde{G}_c(\kappa, \omega)$ over time and frequency.

Since in Eq. (15), the values $2/(T \cdot F)$ and $(T \cdot F)/2$ can be pre-computed, the computational complexity of the proposed filtering scheme requires only $8 \cdot (M/2 - F + 2)$ operations for every block-iteration (linear complexity). The gain Ψ_G in computational complexity with respect to the median filtering of Section III is therefore:

$$\Psi_G = \frac{T \cdot F \cdot \log(T \cdot F)}{8}. \tag{17}$$

For large kernel sizes $T \cdot F$, the gain in computational complexity is large compared to the median filtering and experiments have shown that this gain is obtained without sacrificing in performance.

V. Experiments

Due to the limited length of this paper, only the results of the low-cost solution are shown, which are comparable with the median-filtering solution. For the same breathing fragment as in section II (with the same conditions), the results for the low-cost solution with several kernel sizes $T = F$ and using $\gamma = 1.25$ are shown in Fig. 4. Statistical evaluation is omitted, as our contribution is related to the complexity reduction.

By visual inspection, we can see musical tones in Fig. 4 (b) and we can see the severe signal attenuation in Fig. 4 (c) and (d). Experiments show that we have optimal results for a kernel size $(T, F) = (10, 40)$ (see Fig. 5), which does not show any musical tones. As the optimal kernel size is dependent on the breathing type, we require a model-order selection method, which will be a separate topic for future research.

VI. Conclusions

We presented a modified spectral subtraction scheme to enhance breathing signals captured by a microphone. Breathing signals are quasi-stationary over long time-intervals (in the order of several hundreds of milliseconds) and have a broad and flat frequency spectrum. Hence, 2D median filtering can be applied with relatively large kernel-sizes in the time-frequency plane to suppress musical tones. As the computational complexity of median filtering is very large, we present an alternative filtering scheme that approximates the median filtering scheme. This efficient scheme uses the summed area table and the majority function.

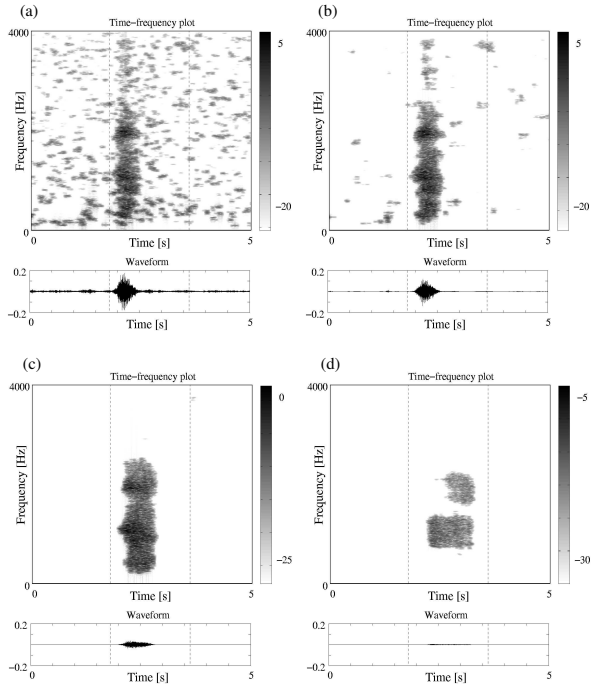

Fig. 4. Spectrogram for filtering scheme with $(T, F) = (5, 5)$ **(a)**, $(10, 10)$ **(b)**, $(20, 20)$ **(c)** and $(40, 40)$ **(d)**.

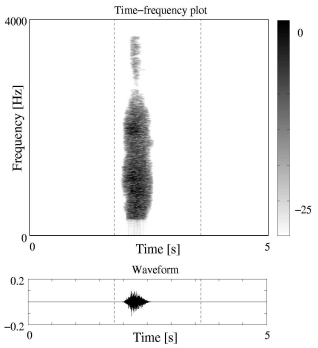

Fig. 5. Spectrogram for filtering scheme with $(T, F) = (10, 40)$.

References

[1] R.M.M. Derkx and C.P. Janse, "Theoretical Analysis of a First-order Azimuth-Steerable Superdirective Microphone Array," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 17, no. 1, pp. 150–162, jan 2009.

[2] S.F. Boll, "Suppression of Acoustic Noise in Speech using Spectral Subtraction," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 27, pp. 113–120, Apr. 1979.

[3] R. Martin, "Spectral subtraction based on minimum statistics," in *Signal Processing VII, Proc. EUSIPCO*, Edinburgh (Scotland, UK), Sep. 1994, pp. 1182–1185.

[4] K-C. Tan, et.al., "Postprocessing Method for Suppressing Musical Noise Generated by Spectral Subtraction," *IEEE Trans. on Speech and Audio Processing*, vol. 6, no. 3, pp. 287–292, May 1989.

[5] F.C. Crow, "Summed-area tables for texture mapping," in *11th conference on Computer graphics and interactive techniques*, Jul. 1984, pp. 207–212.

[6] H.J.W. Belt, "Word length reduction for the integral image," in *IEEE International Conference on Image Processing*, Oct. 2008, pp. 805–808.

[7] L-W. Chang and J-H. Lin, "A Bit-Level Systolic Array for Median Filter," *IEEE Trans. on Signal Processing*, vol. 40, no. 8, pp. 2079–2083, Aug. 1992.