Speech Communication 96 (2018) 102–115

Contents lists available at ScienceDirect

Speech Communication

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

SEDA: A tunable Q-factor wavelet-based noise reduction algorithm for multi-talker babble

Roozbeh Soleymani[a,b,*], Ivan W. Selesnick[a], David M. Landsberger[b]

[a] Department of Electrical and Computer Engineering, Tandon School of Engineering, New York University, 2 Metrotech Center, Brooklyn, NY 11201, USA
[b] Department of Otolaryngology, New York University School of Medicine, 550 1st Avenue, STE NBV 5E5, New York, NY 10016, USA

ABSTRACT

We introduce a new wavelet-based algorithm to enhance the quality of speech corrupted by multi-talker babble noise. The algorithm comprises three stages: The first stage classifies short frames of the noisy speech as speech-dominated or noise-dominated. We design this classifier specifically for multi-talker babble noise. The second stage performs preliminary de-nosing of noisy speech frames using oversampled wavelet transforms and parallel group thresholding. The final stage performs further denoising by attenuating residual high frequency components in the signal produced by the second stage. A significant improvement in intelligibility and quality was observed in evaluation tests of the algorithm with cochlear implant users.

1. Introduction

Although cochlear implants (CIs) have been highly successful at providing speech understanding in optimal listening situations to the profoundly deaf (e.g. Friedland et al., 2010), the performance of CI users is severely impacted by the presence of background noise (e.g. Fetterman and Domico, 2002; Muller-Deile et al., 1995). Therefore, signal processing to remove background noise can be highly beneficial for CI users (e.g. Dawson et al., 2011). One type of noise that has a particularly significant effect on CI user speech understanding is “multi-talker babble” which consists of many people talking simultaneously in the background (e.g. Sperry et al., 1997). However, multi-talker babble is one of the most frequently encountered noises that CI users face. Hence, attenuating the speech from competing talkers is expected to provide speech perception benefits for CI users.

Multi-talker babble is an example of a non-stationary noise. Unlike stationary signals (e.g., white noise), in a non-stationary signal, statistical parameters like mean, variance and autocovariance change over time. Hence it is generally more challenging to predict or model the behavior of a non-stationary signal over time. Although many real-time single-channel noise removal methods have been proposed for CI devices, fewer of these methods have provided benefits in non-stationary noises such as multi-talker babble. Spectral similarities between multi-talker babble and target speech (caused by the fact that both the target speech and noise are comprised of speech signals) as well as the non-stationary nature of multi-talker babble make it difficult to differentiate and separate multi-talker babble from the target speech.

Yang and Fu (2005) proposed using pause detection and spectral subtraction for noise reduction and tested the algorithm with seven post-lingually deafened CI users. While a significant effect of the algorithm was detected with speech-shaped noise, no significant effect of the algorithm was detected with 6 talker babble. Another noise reduction method for CI users is to reduce the gain of the envelope of noise-dominated frequency channels (Bentler and Chiou, 2006). This method has been commercially implemented (e.g. ClearVoice) but Holden et al. (2013) was unable to detect a significant benefit using ClearVoice with multi-talker babble.

Mauger et al. (2012) introduced an optimized noise reduction method by increasing the temporal smoothing of the signal to noise ratio estimate and using a more aggressive gain function. This method was tested in real-time on 12 CI users and significant improvement was found in 4 and 20 talker babble.

Goehring et al. (2016) used auditory features extracted from the noisy speech and a neural network classifier to find and retain the frequency channels which have higher signal to noise ratio and attenuate the channels with lower signal to noise ratio. Two versions of the algorithm (i.e., speaker-dependent and speaker-independent) were tested on 14 cochlear implant users for three different noise types including 20 talker babble. Significant improvement was achieved in multi-talker babble specifically with the speaker-dependent algorithm. However, no significant improvement was observed in multi-talker babble with the speaker-independent algorithm (see Table 1).

Sigmoidal-shaped compression functions have been shown to be effective for speech understanding against a background of multi-talker

[*] Corresponding author.
E-mail address: rs4462@nyu.edu (R. Soleymani).

https://doi.org/10.1016/j.specom.2017.11.004
Received 24 April 2017; Received in revised form 16 October 2017; Accepted 8 November 2017
Available online 09 November 2017
0167-6393/ © 2017 Elsevier B.V. All rights reserved.

Table 1
Summary of results and testing conditions for previous studies investigating denoising of multi-talker babble. Note that the improvement observed was not significant for four of the nine tests. The non-significant improvements are indicated with “(n.s.)”.

Method	Babble type / Source	Mean improvement	Comments
Yang and Fu (2005)	6 Talker /Unknown	7.75 % (n.s.)	7 Subjects in 0, 3, 6 and 9 dB SNRs
Hu et al. (2007)	20 Talker / AUDITEC CD	10–25%	9 Subjects 5 dB SNR, 5 Subjects 10 dB SNR
Kasturi and Loizou (2007)	20 Talker / AUDITEC CD	~ 11%	9 Subjects in 5 and 10 dB SNRs
Ye et al. (2013)	20 Talker / Unknown	~ 0.39 dB SRT (n.s.)	9 Subjects, SRT test
Mauger et al. (2012)	4 - 20 Talker / Unknown	5–7%	12 Subjects, SNR(50%) and SNR(50%)-1dB
Toledo et al. (2003)	Unknown / AUDITEC CD	~ 8% (n.s.)	4 Subjects in 5 dB SNR
Dawson et al. (2011)	Cocktail Party / Field Recording	0.87–1.09 dB SRT	13 Subjects, SRT test
Goehring et al. (2016)	20 Talker / AUDITEC S.L.	0.4 dB SRT (n.s.)	14 Subjects, SRT test, speaker-independent
		2 dB SRT	14 Subjects, SRT test, speaker-dependent

babble with 20 background talkers (Hu et al., 2007; Kasturi and Loizou, 2007) by attenuating channels with a low signal-to-noise ratio (SNR). However, the perceptual and statistical properties of multi-talker babble depend on the number of talkers (Krishnamurthy and Hansen, 2009). The more talkers present in a background noise, the more the properties of the noise resemble stationary noise. The performance of the sigmoidal-shaped compression functions for multi-talker babble with smaller number of talkers is not clear.

Toledo et al. (2003) observed speech intelligibility improvement for multi-talker babble in four cochlear implant users. Their method is based on envelope subtraction and estimates the noise envelope using a minimum tracking technique (See Table 1).

Wavelet-based denoising algorithms have also been introduced for cochlear implant devices. Ye et al. (2013) proposed shrinkage and thresholding in conjunction with a critically-sampled dual-tree complex wavelet transform. While significant improvement was observed in speech-weighted noise, no significant benefit was observed for multi-talker babble. This is expected, because the algorithm was designed for and trained with speech weighted noise.

Many other single-channel denoising methods have been proposed for cochlear implant devices (e.g. Loizou et al., 2005; Healy et al., 2013, and Chung et al., 2004). However, only a subset of these single-channel denoising methods have been evaluated with multi-talker babble noise. For those algorithms, which have been evaluated with multi-talker babble, the testing conditions, sentence corpuses, languages and types of babble noise vary across studies and therefore it is difficult to compare the effectiveness across algorithms. It is worth noting that most algorithms provided statistically significant improvements only for high SNRs. For reference, the results and testing conditions for some of these denoising algorithms are summarized in Table 1.

In this paper, we propose and evaluate a front-end babble noise reduction algorithm. Although the algorithm is not necessarily specific for CI users, we evaluate performance of the algorithm with CI users because they stand to benefit greatly from noise reduction for positive SNRs where we expect the algorithm to perform best.

2. Algorithm

The babble noise reduction problem can be summarized as

$$Y = S + \sum_{i=1}^{n} S_i \tag{1}$$

where Y is the noisy signal, S is the target speech and S_1 to S_n are individual background talkers which collectively form the multi-talker babble. For developing the algorithm, we made the following assumptions:

1. Target speech and background babble both consist of human speech. This makes it difficult to distinguish the target speech from the background babble.
2. Babble, which comprises of multiple independent speech signals, is likely to have a different level of information disorder or uncertainty than a single talker. Features such as entropy, which measure the unpredictability of information content of a signal might be helpful to differentiate target speech from the babble.
3. Target speech is louder (i.e., has greater amplitude variance) than each individual background speaker, i.e.:

$$\sigma_S^2 > \sigma_{S_i}^2 \ \ \forall \ \ 1 \le i \le n. \tag{2}$$

Consequently, in a noisy speech frame, samples originating from the target speech are more likely to have a larger amplitude than samples originating from the babble. Hence, thresholding (which can be used to separate large-amplitude samples from the small-amplitude samples) can potentially solve the babble problem. Note that (2) does not imply that the energy of the target speech is necessarily greater than the total energy of the multi-talker babble. In fact, it is possible that the Signal to Noise Ratio (SNR) is negative while (2) still holds.

However, a simple temporal or spectral thresholding cannot adequately solve such a complex problem as separating one talker from a babble background. There are two reasons for the ineffectiveness of simple temporal/spectral thresholding for babble reduction: First, babble and speech are highly overlapping in time and frequency. Second, some target speech coefficients in the time or frequency domain are inevitably smaller than the threshold level and will be attenuated or set to zero by the thresholding. Moreover, in practice the noise level is unknown and this makes it difficult to estimate a suitable threshold level. In the following sections, we propose a solution to these problems by designing a classifier to estimate the noise level and applying adaptive group thresholding in an oversampled wavelet domain to minimize the overlapping and distortion problems.

In our proposed algorithm, SEDA (Speech Enhancement using Dynamic thresholding Approach), every incoming frame of the noisy speech will go through the following three steps: (1) classification, (2) denoising, and (3) enhancement. The classification stage classifies the incoming noisy frames as being either speech-dominated or noise-dominated. The denoising stage performs adaptive group thresholding in a wavelet domain to attenuate components which primarily originate from babble. The threshold levels in the denoising stage are adjusted in real-time based on the results of the classification stage. Finally, in the enhancement stage, a low pass filter is applied to the noise-dominated frames to eliminate high frequency artifacts resulting from the denoising stage (see Fig. 1).

2.1. Classification

The proposed classifier categorizes relatively short frames of the input signal (consisting of the combination of target speech and the background multi-talker babble) as being either noise-dominated or speech-dominated based on the frame's Signal-to-Noise-Ratio (SNR). In contrast to overall SNR which is estimated over the entire length of the

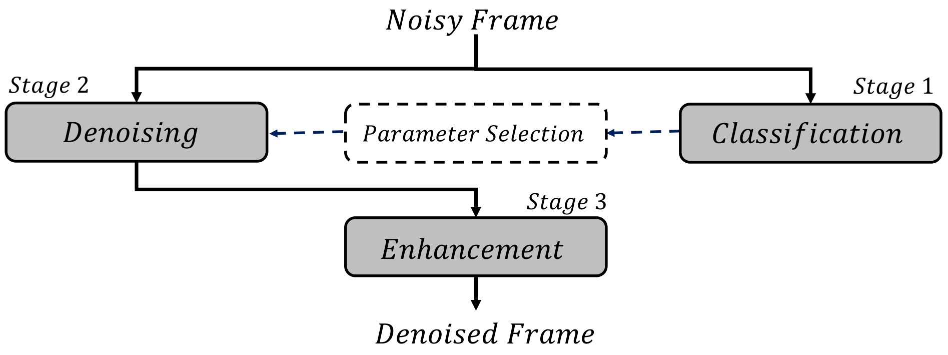

Fig. 1. SEDA overall block diagram.

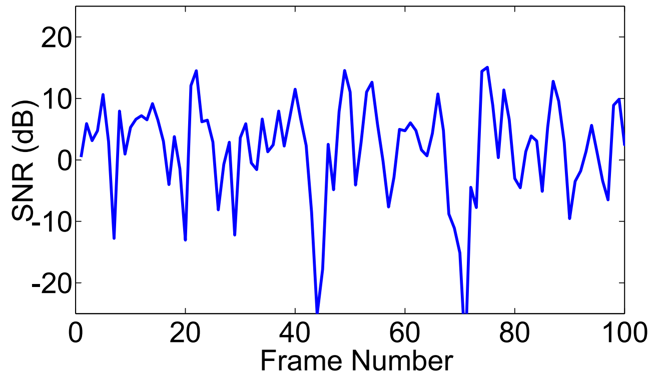

Fig. 2. Local SNR for noisy speech sample with overall SNR = 6. Frame duration = 100 ms.

signal, the local SNR is estimated over relatively short frames of the noisy signal. In the example illustrated in Fig. 2, the short frames are 100 ms in duration. This figure shows the values of the local SNR in 10 seconds of noisy speech (i.e., 100 non-overlapping frames) corrupted by 10 talker babble with an overall SNR of 6 dB. Frames with a positive SNR are considered to be speech-dominated while frames with negative SNRs are considered to be noise-dominated. However, to avoid classifying frames with negligible SNR difference into different classes, a narrow buffer zone is defined between − 1 dB and + 1 dB SNR. Frames with SNRs within this buffer can be correctly classified as either speech-dominated or noise-dominated. A database of 2100 sentences, including 720 male speaker and 720 female speaker IEEE standard sentences (IEEE Subcommittee, 1969), 260 male speaker HINT sentences (Nilsson et al., 1994) and 400 male speaker SPIN sentences (Bilger et al., 1984) was used to create babble and speech samples.

To create each babble sample, the number and gender of talkers were randomly selected. The number of talkers varied from 5 to 10. The frame duration was selected to be 128 ms. As a result, the frame length varies as a function of sampling rate.

2.1.1. Feature selection

Four features sensitive to changes of SNR in short frames of target speech mixed with multi-talker babble noise, were selected. For every incoming noisy speech frame F_i, a feature vector $\mathscr{F}_i = [f_i^{(1)}, f_i^{(2)}, f_i^{(3)}, f_i^{(4)}]$ is formed. Our selected features are as follows:

Entropy $f_{\boldsymbol{i}}^{(1)}$: To compute this feature, we compute the entropy of each frame using its histogram as follows:

$$f_{\boldsymbol{i}}^{(1)} = -\sum_{k=1}^{N} P\ \ (k) log_{10}(P\ \ (k)) = -\sum_{k=1}^{N} \frac{h(k)}{L} log_{10}\left(\frac{h(k)}{L}\right) \tag{3}$$

where h is the amplitude histogram of F_i, $P(k)$ is the probability of the kth bin, N is the number of bins and L is the frame's length. Because L is a constant, to avoid extra computation we simplify (3) and calculate the feature as: $f_{\boldsymbol{i}}^{(1)} = -\sum_{k=1}^{N} h(k) log_{10}(h(k))$. The value of this feature increases with increasing frame SNR.

Post-thresholding to pre-thresholding RMS (Root Mean Square) ratio $f_{\boldsymbol{i}}^{(2)}$: The value of this RMS ratio increases with increasing frame SNR. To compute this feature, first we set a threshold level $\tau(F_i)$:

$$\tau(F_i\ ) = \frac{1}{L} \mathrm{K}\ \|F_i\|_1 \tag{4}$$

where $\|F_i\|_1$ is the l_1 norm of the frame F_i. Then we find F_i^{th} by hard thresholding F_i with threshold level $\tau(F_i)$. Finally, we calculate the ratio of the RMS values of F_i^{th} and F_i.

$$f_i^{(2)} = \frac{rms(F_i^{th})}{rms(F_i)} \tag{5}$$

Envelope Variance $f_{\boldsymbol{i}}^{(3)}$: The variance of the frame's envelope increases with the frame's SNR. To obtain this feature, we first compute the frame's envelope e_i as follows:

$$e_i(n) = \frac{1}{L_w} \sum_{k=-\frac{L_w}{2}}^{\frac{L_w}{2}} |F_i(k + nh)| w(k) \tag{6}$$

where, L_w is the window length, w is the window and h is the hop size. Here we use non-overlapping rectangular windows with h = L_w. Then we find the normalized envelope $\hat{e}_i$:

$$\hat{e}_i(n) = \frac{e_i(n)}{\max(e_i)} \tag{7}$$

and finally, we calculate the envelope variance:

$$f_{\boldsymbol{i}}^{\ (3)} = \mathrm{var}(\hat{\mathrm{e}}_\mathrm{i}) = \frac{1}{N_w} \sum_{\mathrm{n}=1}^{N_w} (\hat{e}_i(n) - \mu_i\ )^2 \tag{8}$$

where, N_w is the total number of windows in a frame and $\mu_i = \frac{1}{N_w} \sum_{\mathrm{n}=1}^{N_w} \hat{e}_i(n)$.

Envelope Mean-Crossing $f_{\boldsymbol{i}}^{(4)}$: The envelope mean-crossing decreases with increasing frame's SNR. To extract this feature first we compute the envelope e_i using (6). Then we calculate the envelope mean-crossing as follows:

$$f_{\boldsymbol{i}}^{\ (4)} = \frac{1}{2N_w} \sum_{\mathrm{k}=2}^{N_w} \left|\mathrm{sign}\left(\hat{\mathrm{e}}_\mathrm{i}(\mathrm{k}) - \mu_{\hat{\mathrm{e}}_\mathrm{i}}\right) - \mathrm{sign}\left(\hat{\mathrm{e}}_\mathrm{i}(\mathrm{k}-1) - \mu_{\hat{\mathrm{e}}_\mathrm{i}}\right)\right| \tag{9}$$

where: $\hat{\mathrm{e}}_\mathrm{i}$ and $\mu_{\hat{\mathrm{e}}_\mathrm{i}}$ are the normalized envelope and its mean respectively and sign(x) is defined as:

$$\mathrm{sign}(\mathrm{x}) = \begin{cases} 1, & x > 0 \\ -1, & x < 0 \\ 0, & x = 0 \end{cases}$$

2.1.2. Feature optimization

For each of the previously discussed features, the quality can be estimated using a Fischer score (Tang and Liu, 2014; Gu et al.,2012; Duda, 2001):

Table 2
Selected values for feature parameters by numerically maximizing Fischer score. Values are selected for frame duration of 128 ms and sampling rate of 16,000 samples per second. B is the bin width in histogram, M is the long term maximum amplitude of the noisy signal (constant), K is the threshold coefficient in Eq. (4) and L_w is the window length in Eq. (6).

Feature	Parameter	Selected value
$f_i^{(1)}$	$B \in \mathbb{R}$	$0.05M$
$f_i^{(2)}$	$K \in \mathbb{R}$	3
$f_i^{(3)}, f_i^{(4)}$	$L_w \in \mathbb{N}$	50

$$S = \frac{\sum_{j=1}^{N_c} n_j(\mu_j - \mu)^2}{\sum_{j=1}^{N_c} n_j\sigma_j^2} \tag{10}$$

where N_c is the number of classes (i.e., $N_c = 2$), μ_j is the mean of the feature in class j, μ is the overall mean of the feature, σ_j is the variance of the feature in class j, and n_j is the number of samples in class j. To optimize the quality of features, (10) was numerically maximized for each feature and the suitable values for feature parameters were selected. Table 2 shows the selected values for parameters which determine the quality of each feature.

2.1.3. Weighted PCA (Principle component analysis)

To reduce the correlation (redundancy) between the features, we use PCA to generate a new smaller set of uncorrelated features. Assuming $\mathscr{F}$ is the feature matrix and N_F is the total number of noisy speech frames, we can write: $\mathscr{F} = [\overline{\mathscr{F}_1}, \overline{\mathscr{F}_2} . ..\overline{\mathscr{F}_{N_F}}]$. First we find $\mathscr{F}_0$ by removing the mean of features as follows:

$$\mathscr{F}_0 = \mathscr{F} - M \tag{11}$$

where M is the mean matrix of features. The goal is to find the transformation matrix T, such that:

$$\mathscr{F}_d = T\mathscr{F}_0 \tag{12}$$

where, $\mathscr{F}_d$ is the de-correlated feature matrix. The covariance matrix C_0 of $\mathscr{F}_0$ can be obtained as follows:

$$C_0 = \frac{1}{N}\mathscr{F}_0\mathscr{F}_0{}^T. \tag{13}$$

Using (12) and (13) we can write (Shlens, 2003; Bishop, 2007; Bello, 2016):

$$C_d = \frac{1}{N}\mathscr{F}_d\mathscr{F}_d{}^T = \frac{1}{N}[T\mathscr{F}_0][T\mathscr{F}_0]^T = T\left[\frac{1}{N}\mathscr{F}_0\mathscr{F}_0{}^T\right]T^T = TC_0T^T \tag{14}$$

where, C_d is a diagonal rank-ordered covariance matrix of uncorrelated feature matrix $\mathscr{F}_d$. In order to diagonalize the symmetric matrix of C_0 we compute the orthogonal matrix of its eigenvectors. Assuming r is the rank of covariance matrix C_0, the eigenvectors of C_0 and their associated eigenvalues can be written as: $\{\bar{v}_1, \bar{v}_2, \cdots \bar{v}_r\}$ and $\{\lambda_1, \lambda_2, \cdots \lambda_r\}$ such that: $C_0\ \bar{v}_i = \lambda_i\bar{v}_i$. Now we define: $V = [\bar{v}_1\ \ \bar{v}_2\ \ .\ .\ .\ \ \bar{v}_r]$ and using (14) we have (Shlens, 2003; Bishop, 2007; Bello, 2016):

$$C_d = V^TC_0V \Rightarrow\ \ T = V^T \tag{15}$$

The transform matrix T is a matrix whose rows are the eigenvectors of the covariance matrix C_0. Having T, we can de-correlate the original feature vector F_0 using Eq. (12). Because we have four original features, in the case of $r < 4$ we select $4 - r$ arbitrary orthonormal vectors and complete the V. These orthonormal vectors do not change the result because they are associated with zero variance features (Shlens, 2003). To take the quality of each feature into account we give a relative weight to each feature based on its Fischer quality score. The weighted covariance matrix C_0 will be obtained as (Yue and Tomoyasu, 2004):

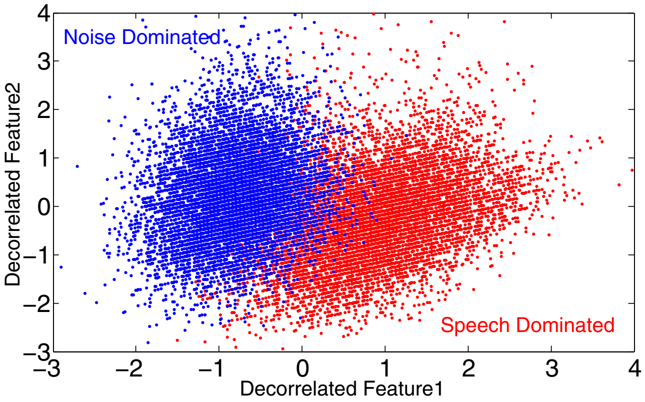

Fig. 3. Scatter plot of de-correlated features 1 and 2 computed over 25,000 randomly generated noisy speech frames corrupted with multi-talker babble with random SNR and number of talkers (Between 5–10). Blue dots represent the noise-dominated frames and red dots represent the speech-dominated frames. Frame duration = 128 ms, Sampling rate = 16,000 samples per second.

$$C_0 = \frac{1}{N}W\mathscr{F}_0\mathscr{F}_0{}^T\mathrm{W}^T \text{ and } W = \begin{bmatrix} S_1 & 0 & 0 & 0 \\ 0 & S_2 & 0 & 0 \\ 0 & 0 & S_3 & 0 \\ 0 & 0 & 0 & S_4 \end{bmatrix} \tag{16}$$

where, W is the weighting matrix and S_1–S_4 are the average Fischer scores of the four original features. After completing this stage, we have four new de-correlated features which are ranked based on their variances. We selected the first two features with the highest Fischer score (see Fig. 3).

2.1.4. Training with GMM

To train the classifier, we use the two dimensional Gaussian Mixture Model (GMM) where each class is modeled as the sum of a n Gaussian distributions as follows: (Reynolds, 2009):

$$\begin{aligned} G(\mathscr{F}_d|\mu, w, C) &= \sum_{i=1}^{n} w_i\mathscr{N}(\mathscr{F}_d|\mu_i, C_i) \\ &= -\sum_{i=1}^{n}\frac{w_i}{(2\pi)^{\frac{d}{2}}\sqrt{|C_i|}}e^{\left\{-\frac{1}{2}[\mathscr{F}_d-\mu_i]^T C_i^{-1}[\mathscr{F}_d-\mu_i]\right\}} \end{aligned} \tag{17}$$

where $\mathscr{F}_d$ is a two-dimensional de-correlated feature matrix using weighted PCA and w_i, μ_i and C_i are the weight factor, mean and covariance of the ith Gaussian distribution respectively. We also should have $\sum_{i=1}^{n} w_i = 1$.

The probability of a data sample k with a feature vector $\mathscr{F}_d(k)$ belonging to a Gaussian j can be calculated as (Bishop, 2007; Bello, 2016):

$$p_j^k = \frac{w_j\mathscr{N}(\mathscr{F}_d(k)|\mu_j, C_j)}{\sum_{i=1}^{n} w_i\mathscr{N}(\mathscr{F}_d(k)|\mu_i, C_i)} \tag{18}$$

In order to train our model, we use the iterative Expectation-Maximization (EM) algorithm (Bishop, 2007, Reynolds et al., 2000; Reynolds, 2009; Bello, 2016). In order to fit a Gaussian to each cluster we should maximize the following logarithmic function (Bishop, 2007; Bello, 2016):

$$\log\{p(\mathscr{F}_d|\mu, C, w)\} = \sum_{k=1}^{N_F}\log\left\{\sum_{i=1}^{n} w_i\mathscr{N}(\mathscr{F}_d(k)|\mu_i, C_i)\right\} \tag{19}$$

where N_F is the number of data samples (i.e., the number of audio frames).

We first initialize w_i, μ_i, C_i and calculate p_i^k, then update w_i, μ_i, C_i using the calculated values of p_i^k (Bishop, 2007; Bello, 2016):

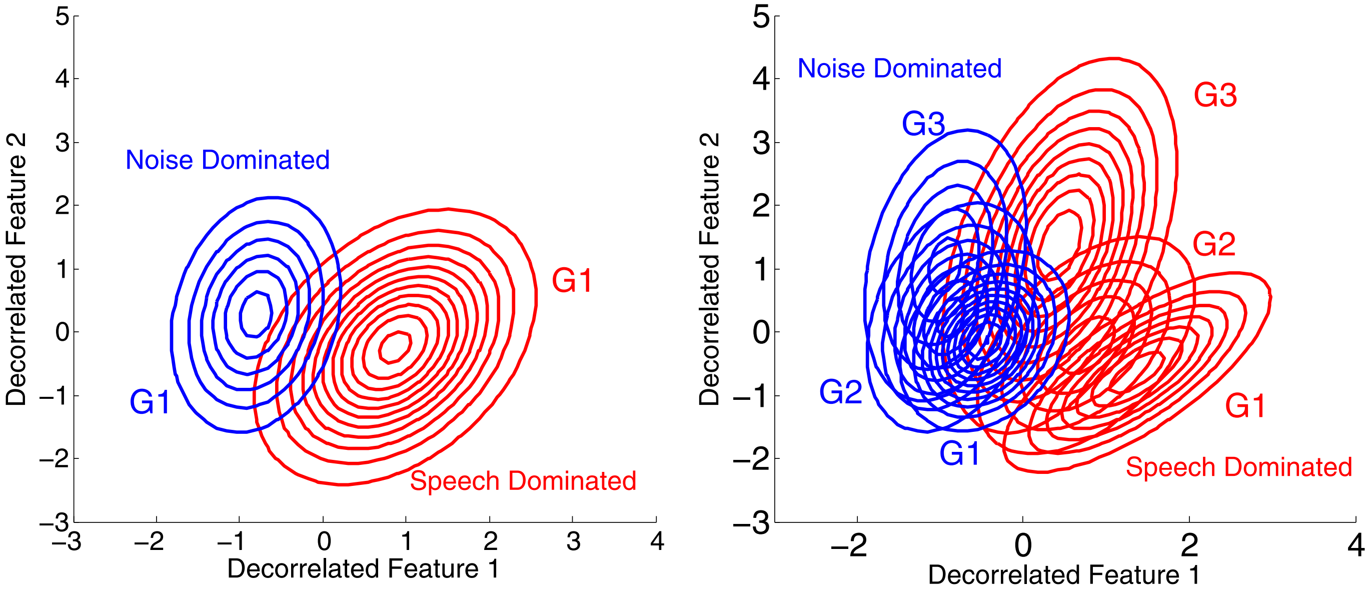

Fig. 4. GMM plots using EM method with only one Gaussian per class (left) and three Gaussians per class (right). computed over 10 h (281,250 frames) of randomly generated noisy speech frames corrupted with multi-talker babble with random SNR and number of talkers (between 5–10). Frame duration = 128 ms, Sampling rate = 16,000 samples per second.

$$\mu_i^{new} = \frac{\sum_k p_i^k \mathscr{F}_d(k)}{\sum_k p_i^k} \quad \omega_i^{new} = \frac{\sum_k p_i^k}{N_F}$$

$$C_i^{new} = \frac{\sum_k p_i^k (\mathscr{F}_d(k) - \mu_i^{new})(\mathscr{F}_d(k) - \mu_i^{new})^T}{\sum_k p_i^k} \tag{20}$$

We repeat the above stages until the convergence of (19). We can train our classifier with only one Gaussian for each class to avoid heavy computation. By increasing the number of Gaussians the classifier accuracy will slightly increase (See Fig. 4). In the final algorithm, a classifier with only one Gaussian for each class was trained.

2.1.5. Classification using MAP (Maximum a posteriori estimation)

After the classifier is trained, we find the probability of each test feature set $\mathscr{F}_d$ belonging to a class X by (Duda, 2001; Bello, 2016):

$$argmax_X \, [P(\mathscr{F}_d|class_X)P(class_X)] \tag{21}$$

$X \in \{S,\ N\}$ S: Speech-Dominated N: Noise-Dominated.

$$\text{where } P(\mathscr{F}_d|class_X) = \sum_{i=1}^{n} w_i \mathscr{N}(\mathscr{F}_d|\mu_i,\ C_i). \tag{22}$$

μ_i, C_i and w_i are also available from the GMM training process.

The values of $P(class_N)$ and $P(class_S)$ change as a function of the overall (long term) SNR and can be obtained during training by computing the number of each class occurrence divided by the total data samples in training data for each overall SNR. If the overall SNR changes very quickly (i.e., fast varying noisy condition) we can assume $(class_N) = P(class_S) = 0.5$. In most of the cases the general noise level does not change quickly (i.e., slow varying overall SNR). In this situation we can estimate more accurate values for $P(class_N)$ and $P(class_S)$ by roughly estimating the overall SNR. To estimate the overall SNR we suggest a very simple classifier which classifies the long frames of the noisy speech (i.e., four seconds long) into one of the 6 classes listed in Table 3 and choose the $P(class_N)$ accordingly.

The overall SNR classifier uses only two of the features mentioned earlier in this section (RMS ratio and envelope mean crossing) calculated over the long frames of the noisy speech without de-correlating the features with PCA. We use GMM with a single Gaussian per class for training the overall SNR classifier (see Fig. 5). Note that the independent accuracy of the overall SNR classifier is not a concern. However, this classifier works as a component of the SEDA classifier and its accuracy will affect the accuracy of SEDA classifier. The SEDA classifier's accuracy is measured in the next section. $P(class_N)$ and $P(class_S)$ should be continuously updated based on the estimated overall SNR and the frequency of overall SNR detection update depends on our assumption of how fast the noisy environment varies. In this work we updated $P(class_N)$ and $P(class_S)$ once every four seconds.

Table 3
Selected values for $P(class_N)$ for various overall SNR classes. Note that $(class_S) = 1 - P(class_N)$.

Overall SNR	$P(class_N)$
SNR < −1.5 dB	0.8171
−1.5 dB < SNR < 1.5 dB	0.6599
1.5 dB < SNR < 4.5 dB	0.4907
4.5 dB < SNR < 7.5 dB	0.3645
7.5 dB < SNR < 10.5 dB	0.2695
SNR > 10.5 dB	0.1941

2.1.6. Performance evaluation

The performance of the classifier was evaluated using two-fold cross validation (Kohavi, 1995). First, the classifier was trained with noisy speech samples randomly created from half of the sentence database (with random number and gender of talkers). Then the resulting classifier was evaluated using test samples created from the second half of the sentence data base. Subsequently, the following accuracy metrics were computed:

$$P = \frac{C}{C + f^+} \quad R_N = \frac{C}{C + f^-} \quad F = \frac{2PR}{P + R} \tag{23}$$

where C, f^+ and f^- are correct, false positive and false negative detection, respectively. Then we swapped the testing and training databases and repeated the same process and obtained new values for accuracy metrics. Finally, we averaged the resulting two values for each accuracy metric as per two-fold cross validation method (Powers 2011; Swets 1988; Bello, 2016).

Fig. 6 shows the calculated F accuracy metric for a classifier trained with a single Gaussian for each class. The same result was achieved by testing the classifier with 10-talker babble extracted from the AzBio testing material which consists of 5 male and 5 female speakers (Roland et al., 2016).

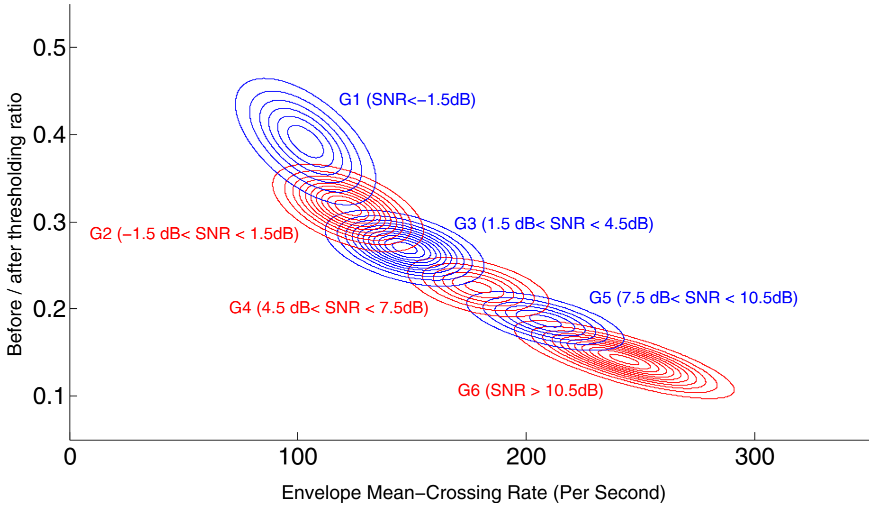

Fig. 5. GMM plots using EM method with only one Gaussian per class for overall SNR classifier. Computed over 50,000 long frames of randomly generated noisy speech corrupted with multi-talker babble with random SNR and number of talkers (between 5–10). Long Frames Duration = 4 s, Sampling rate = 16,000 samples per second.

To train the final SEDA classifier, we used half of the sentence data base to create multi-talker babble samples. We used the other half to create multi-talker babble for the listening test described in Section 3. As was done for the classifier evaluation, we randomized the number and gender of talkers to train the final classifier. IEEE standard sentences with male speakers were used to create the target speech for the listening test (see Section 3). Hence for training the final SEDA classifier we did not use IEEE sentences with male speaker as target speech.

2.2. Denoising

Sparsification using an oversampled wavelet transform is an effective way to minimize the overlapping between signal and noise coefficients. However, sparsification is an iterative process which often cannot be implemented in real-time algorithms due to its high computational requirements. Moreover, human speech cannot be efficiently sparsified in most wavelet domains unless we implement additional measures (e.g., Morphological Component Analysis; MCA) (Selesnick, 2010; Selesnick, 2011a). The representation of the clean speech samples in an oversampled Tunable Q-factor Wavelet Transform (TQWT; Selesnick, 2011b) exhibits some degree of group sparsity which does not exist in babble samples. SEDA takes advantage of this property (among others) to denoise the speech samples which are corrupted by multi-talker babble.

Note that increasing the oversampling rate of a wavelet transform will increase number of samples and consequently the required computation by the same factor. Hence using a conventional filter bank in which each output channel has the same sampling frequency as the input signal has the disadvantage of increasing the computational costs in real-time applications. TQWT provides the ability to optimize the oversampling rate. A TQWT is defined by three parameters which can be adjusted independently: Q-factor, the redundancy, and the number of levels (Fig. 7). The Q-factor is a measure of the oscillatory behavior of a pulse; it is defined in terms of the spectrum of the pulse as the ratio of its center frequency to its bandwidth. The redundancy is the oversampling rate of the wavelet transform and is always greater than 1. By changing these parameters, we can obtain different representations of the signal in the wavelet domain. We use this property later in this paper in parallel denoising technique. Another advantage of the TQWT is in its spectral properties, namely the frequency responses of its subbands, are consistent with the human auditory system. The distribution of the center frequencies of the sub-bands and the shape of the frequency responses of the TQWT resemble Mel-scale and Gammatone filter banks that are designed to reflect the human auditory system (Fig. 7).

2.2.1. Adaptive group thresholding

We propose an adaptive group thresholding of the TQWT domain coefficients of the noisy speech, based on the following strategies:

1. For each sub-band i in the TQWT domain, the threshold level should be just enough to remove most of the babble noise with minimum

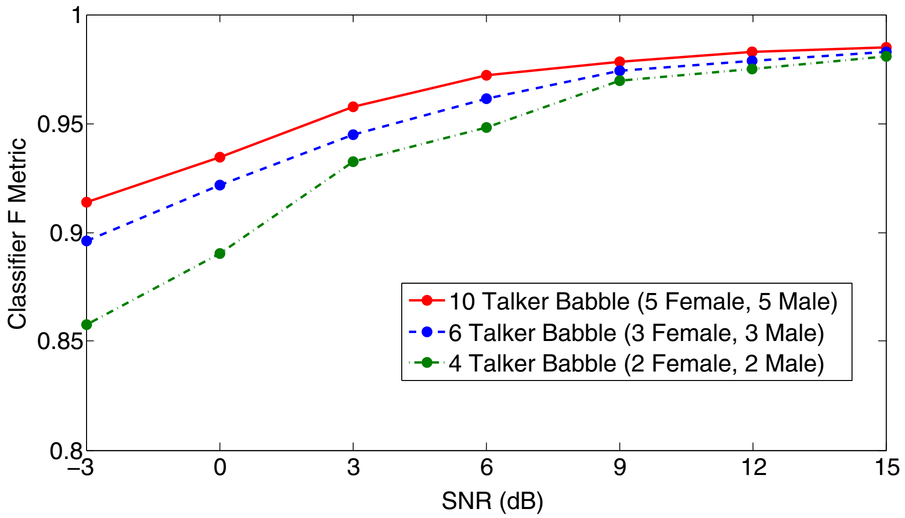

Fig. 6. Accuracy metric (F) of SEDA classifier measured over 1 h of noisy speech corrupted with multi-talker babble for each overall SNR and babble type using two-fold cross validation method.

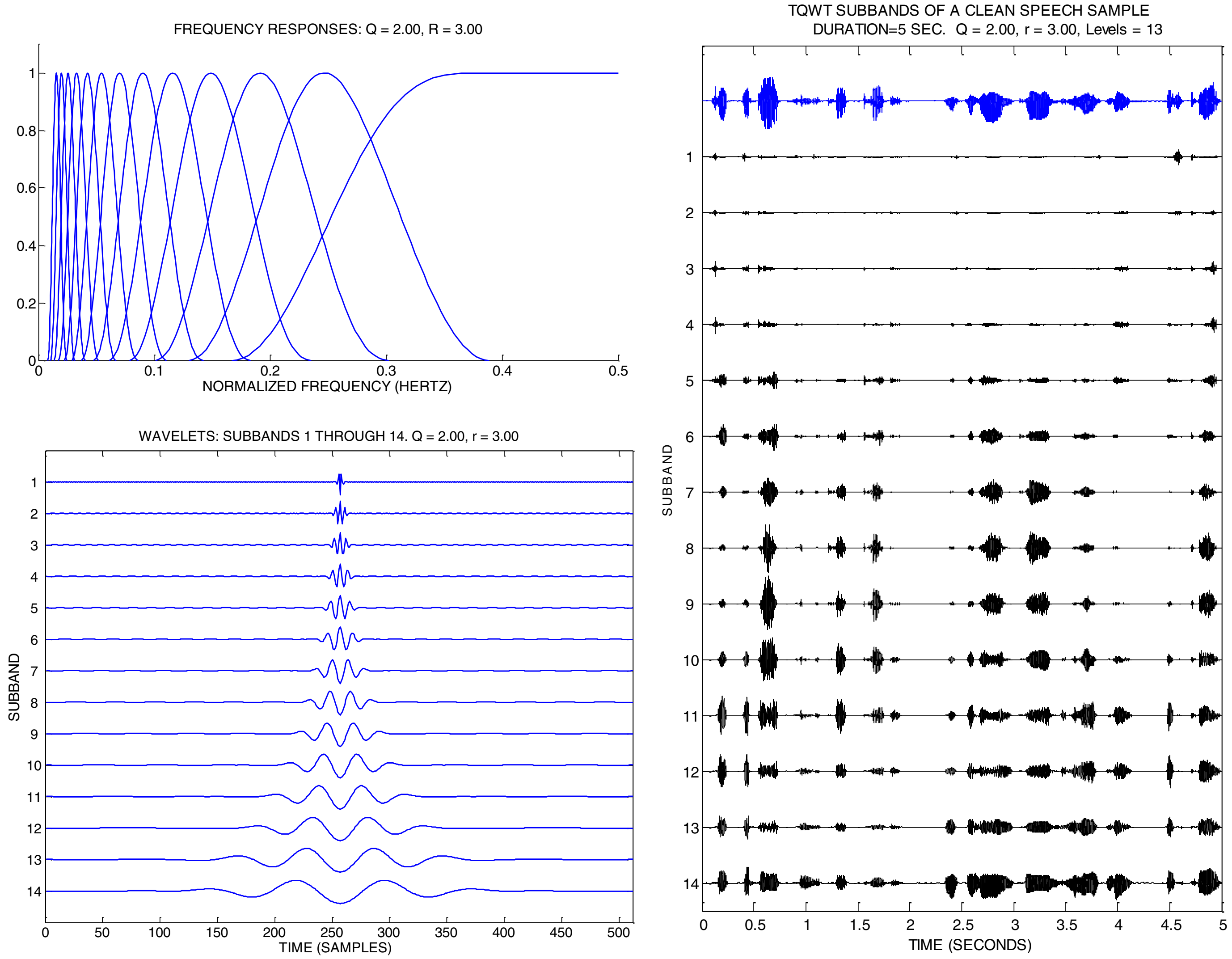

Fig. 7. Frequency response (top left) sub-band wavelets (bottom left) and sub-band coefficients (right) of a TQWT with $Q = 2,\ r = 3, J = 13$.

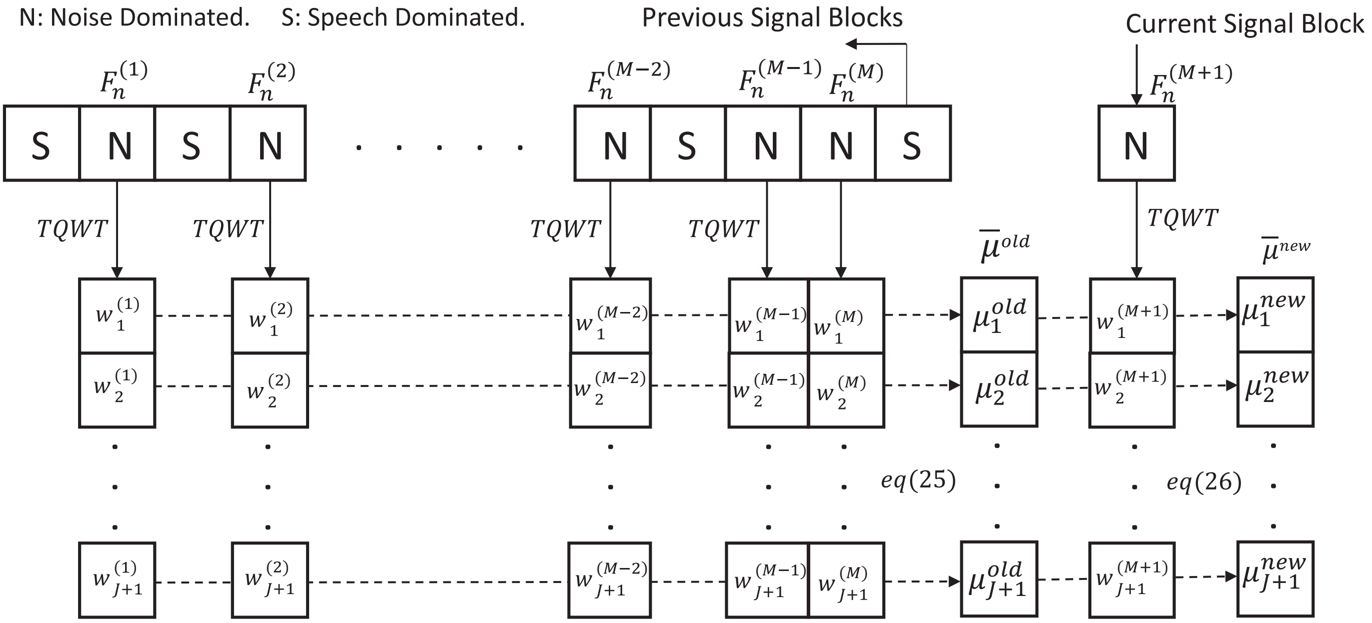

Fig. 8. Block diagram of average noise level updating process.

distortion of the target speech. Hence for a given sub-band i we need to know the noise level in order to select the appropriate threshold level. If the current noisy speech frame is speech-dominated, we estimate the noise level based on the average noise level in the same sub-band over the last few noise-dominated frames.

2. For every frame, we divide each TQWT sub-band into multiple shorter segments (i.e., coefficient-group) where each coefficient-group consists of a few coefficients (SEDA works with 16 coefficients per coefficient-group). Hard and soft thresholding will be used alternatively for different coefficient-groups.

   For a real-valued signal x, hard and soft thresholding with threshold level T are defined with $H_T(x)$ and $S_T(x)$ as follows:

$$H_T(x) = \begin{cases} 0, & |x| \le T \\ x, & |x| > T \end{cases} \quad S_T(x) = \begin{cases} x + T, & x < -T \\ 0, & -T \le x \le T \\ x - T, & x > T \end{cases} \tag{24}$$

   Hard thresholding will be used for coefficient-groups with small l_1 norm value. This will remove many small coefficients originating from the noise source. Recall that target speech is louder than any individual background talker and has some degree of group sparsity in TQWT domain, therefore low amplitude coefficients scattered across the sub-band without forming a distinct group of coefficients, are more likely to originate from the babble source. A milder soft thresholding (with a smaller threshold level) will be used for coefficient-groups with large l_1 norm. This will prevent distortion when a mixture of large and small coefficients coming from target speech are concentrated in a group/cluster (see Fig. 8). Using an aggressive hard thresholding in these cases would eliminate the smaller coefficients and would lead to distortion.

3. General thresholding aggressiveness (level) for each frame is also determined based on the result of the classification. A more aggressive thresholding is used for noise-dominated frames whereas a less aggressive thresholding is used for speech-dominated frames. Details are given in following sub-sections.

Updating the threshold level

As previously mentioned, threshold levels in each sub-band depend on the average noise level over the last few noise-dominated frames. To update the noise level estimation for every incoming frame we define an array $\overline{\mu}$ as follows:

$$\overline{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_{J+1} \end{bmatrix} \text{ where: } \mu_i = \frac{1}{M}\sum_{k=1}^{M} \|w_i^{(k)}\|_1 \quad \text{and}$$

$$w^{(k)} = \{w_1^{(k)},\ w_2^{(k)},\ \dots,\ w_{J+1}^{(k)}\} = \varphi(F_n^{(k)}) \tag{25}$$

where μ_i is the estimated noise level for sub-band i, obtained by averaging $l_1$norm of that sub-band over the last M noise-dominated frames, $F_n^{(k)}$ is the kth frame in the last M noise-dominated frames and $w_i^{(k)}$ is its ith sub-band in TQWT domain and J is the total number of levels in TQWT (denoted with φ).

We estimate the current noise level at each sub-band of the TQWT by averaging the noise level in that sub-band over the last M noise-dominated frames. If the ambient noise is relatively steady, the average noise level in TQWT sub-bands does not change quickly. Conversely, the noise level in sub-bands changes quickly in response to relatively fast varying and non-stationary noise such as multi-talker babble. As we increase the number of talkers in babble, the noise level in TQWT sub-bands becomes steadier. To keep up with the fast variation of the babble noise in TQWT sub-bands, a relatively small value for M (e.g., $M \le 5$) is preferred. Choosing large values for M would decrease the sensitivity of the algorithm to the transient variations of the babble level in TQWT sub-bands. In our implementation, M is set to 5 as our experiments suggest that this value provides a relatively accurate noise level estimation for a wide range of multi-talker babble conditions. However, further investigation is required to determine the optimal value of M.

In the event that a new noise-dominated frame $F_n^{(M+1)}$ is detected, we update each element of array $\overline{\mu}$ as follows:

$$\mu_i^{new} = \frac{(M-1)\mu_i^{old} + \ \|w_i^{(M+1)}\|_1}{M} \tag{26}$$

This updating process is shown in Fig. 8.

Thresholding

The previous steps produce an updated array of estimated noise levels for all sub-bands. Using this array, we implement the adaptive group thresholding for each sub-band as follows: Denoting by F an incoming frame of the noisy speech, we write:

$$w = \varphi(F) \quad \text{where } w = \{w_1, w_2, \dots, w_{J+1}\}$$

As discussed above, each TQWT sub-band i will be divided into n_i coefficient-groups as follows:

$$w_i = \{c_1,\ c_2,\ \dots,\ c_{n_i}\}$$

where, $c_1$to c_{n_i} are coefficient-groups of w_i. For each coefficient-group c_k of sub-band w_i we define $r_k^{(i)}$ as:

$$r_k^{(i)} = n_i \frac{\|c_k\|_1}{\|w_i\|_1}. \tag{27}$$

Using $r_k^{(i)}$ we classify each coefficient-group as either high-amplitude or low-amplitude, and apply hard and soft thresholding to low and high amplitude coefficient-groups respectively, as follows:

$$\hat{c}_k = \begin{cases} H_{T_1}(c_k), & r_k^{(i)} \le \gamma \\ S_{T_2}(c_k), & r_k^{(i)} > \gamma \end{cases}, \quad T_1 = \frac{\rho\tau\mu_i}{L_i}, \quad T_2 = \epsilon T_1 \tag{28}$$

where μ_i is the updated average noise level of the sub-band i over the last M noise-dominated frames, L_i is the length of sub-band i, τ controls the thresholding aggressiveness based on the frame's class (we selected $\tau = 1$ for speech-dominated frames and $\tau = 1.5$ for noise-dominated frames), ϵ is a reduction factor for soft thresholding which should always be smaller than 1 (we selected $\epsilon = 0.3$), γ should always be greater than 1 (we selected $\gamma = 5$) and ρ determines our desired overall denoising aggressiveness which mainly depends on the overall SNR. Our experiments with various values for ρ shows that in noisier situations (i.e., lower SNR) where the target speech is not sufficiently stronger than the background noise, we should select slightly smaller values for ρ to avoid target speech distortion. Conversely, in higher SNRs we can select slightly larger values for ρ which maximizes the noise reduction without major distortion of the target speech. As a part of the SEDA classifier, we roughly classified the overall SNR into one of the six SNR ranges listed in Table 2 as discussed in Section 2.1.5. The main purpose of that classification was to estimate the values of $P(class_N)$ and $P(class_S)$ for maximum a posteriori estimation. In addition to MAP estimation, we use this estimated SNR range to select the value of ρ in the denoising stage (we selected $\rho = 2$ for overall SNR < 4.5, $\rho = 3$ for 4.5 < overall SNR < 10.5 and $\rho = 3.5$ for overall SNR > 10.5). Note that the selected values for SEDA parameters are tuned for multi-talker babble noise with number of talkers between 4–20 and we used the same values for SEDA parameters during the listening tests described in Section 3. Fig. 9 shows that soft thresholding preserves the shape of the clusters (by keeping smaller coefficients) in speech originated high amplitude coefficient-groups c_1, c_2, c_7 and c_8.

2.2.2. Parallel denoising

Adaptive group thresholding usually inflicts some distortion to the original speech. We propose a parallel denoising approach to recover the distorted parts of the speech. Parallel denoising also changes the

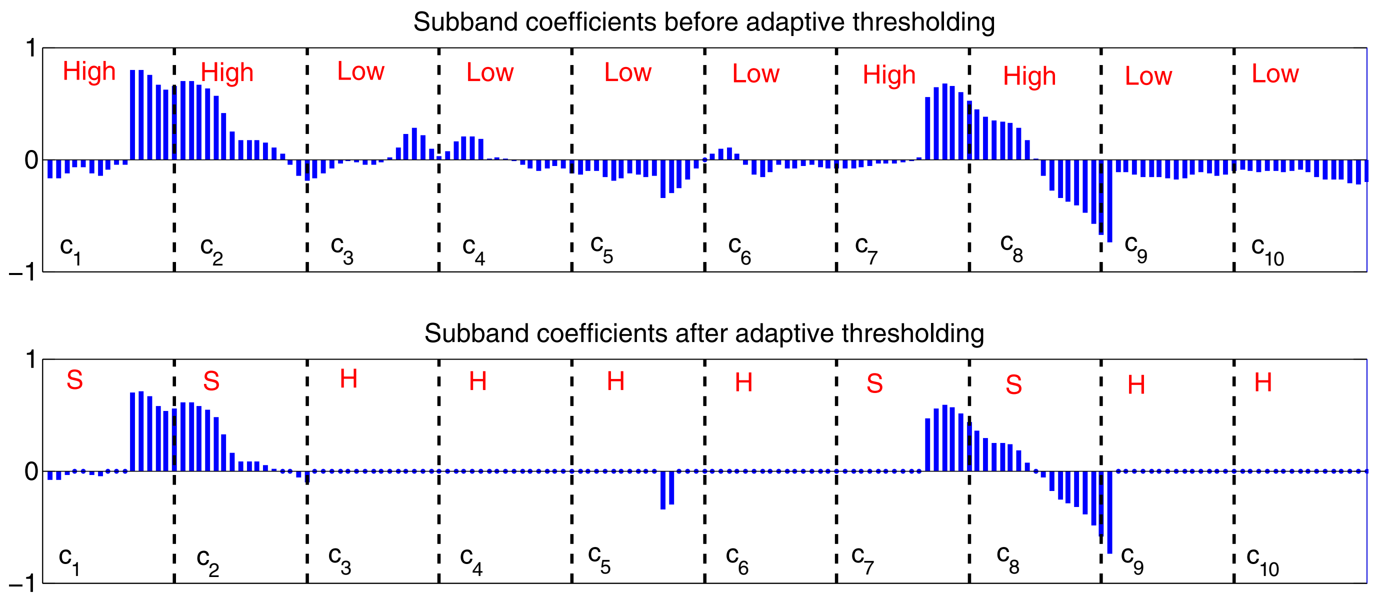

Fig. 9. An example of adaptive group thresholding in a sub-band of TQWT. Coefficient-groups numbers (c_i), High amplitude coefficient-groups (High), Low amplitude coefficient-groups (Low), Hard thresholding (H) and Soft thresholding (S) are shown before and after adaptive thresholding.

behavior of the residual babble noise. We will employ this property in the next section to further de-noise the signal.

First we create three distinct representations of the signal in the wavelet domain using three TQWTs with different settings. Then we apply adaptive group thresholding to each representation and create three slightly different de-noised versions of the same signal. The three resulting de-noised signals will eventually be averaged. It is likely that some areas which are distorted in one de-noised version will be recovered by another and this potentially reduces the overall distortion.

To increase the denoising performance, three TQWTs should have low, medium and high Q factors respectively. This will assure three different representations in the wavelet domain. The redundancy and number of levels in each TQWT should be selected so that the signal's energy is distributed over many sub-bands. Our selected values for a frame duration of 128 ms and sampling rate of 16,000 samples per second (i.e., frame length of 2048 samples) are: $Q_1 = 9$, $r_1 = 3$, $J_1 = 67$, $Q_2 = 5$, $r_2 = 3$, $J_2 = 43$, $Q_3 = 2$, $r_3 = 3$, $J_3 = 20$. Using three TQWTs, for an incoming noisy speech frame F we have: $\dot{w} = \varphi_1(F)$, $\ddot{w} = \varphi_2(F)$, $\dddot{w} = \varphi_3(F)$ where, $\dot{w}$, $\ddot{w}$ and $\dddot{w}$ are three different wavelet domain representations of frame F. If we denote the adaptive group thresholding process with Ŧ, we have: $\dot{w}_{\text{Ŧ}} = \text{Ŧ}(\dot{w})$, $\ddot{w}_{\text{Ŧ}} = \text{Ŧ}(\ddot{w})$ and $\dddot{\mathrm{w}}_{\text{Ŧ}} = \text{Ŧ}(\dddot{w})$ and applying inverse TQWT to $\dot{w}_{\text{Ŧ}}$, $\ddot{w}_{\text{Ŧ}}$ and $\dddot{\mathrm{w}}_{\text{Ŧ}}$ we have: $\dot{F}_{\text{Ŧ}} = \varphi_1^{-1}(\dot{w}_{\text{Ŧ}})$, $\ddot{F}_{\text{Ŧ}} = \varphi_2^{-1}(\ddot{w}_{\text{Ŧ}})$ and $\dddot{F}_{\text{Ŧ}} = \varphi_3^{-1}(\dddot{w}_{\text{Ŧ}})$ where $\dot{F}_{\text{Ŧ}}$, $\ddot{F}_{\text{Ŧ}}$ and $\dddot{\mathrm{F}}_{\text{Ŧ}}$ are three different de-noised versions of F. Finally, the averaged result will be:

$$\hat{F}_{avg} = \alpha\left(\dot{F}_{\text{Ŧ}} + \ddot{F}_{\text{Ŧ}} + \dddot{\mathrm{F}}_{\text{Ŧ}}\right) \tag{29}$$

where α is a gain parameter to control the output signal's energy (Fig. 10). We selected $\alpha = \frac{1}{3}$ to roughly equalize the loudness of the target speech in input and output of the SEDA algorithm. To measure the effect of the parallel denoising on reducing the denoising distortion we use normalized Euclidean distance applied to the magnitude of the spectrograms which is defined as:

$$E_d(X_1, X_2) = \frac{\| |S_1| - |S_2| \|_2}{\|S_2\|_2} \tag{30}$$

where, S_1 and S_2 are Short Time Fourier Transforms (STFT) of audio signals X_1 and X_2 respectively. Our experiments show that parallel denoising effectively reduces the normalized spectral Euclidean distance between de-noised and clean speech. This means that on average, the normalized spectral Euclidean distance between the output of the parallel denoising ($\hat{F}_{avg}$) and clean speech is smaller than each of the three Euclidean distances between denoised versions of $\dot{F}_{\text{Ŧ}}$, $\ddot{F}_{\text{Ŧ}}$ and $\dddot{\mathrm{F}}_{\text{Ŧ}}$ and clean speech.

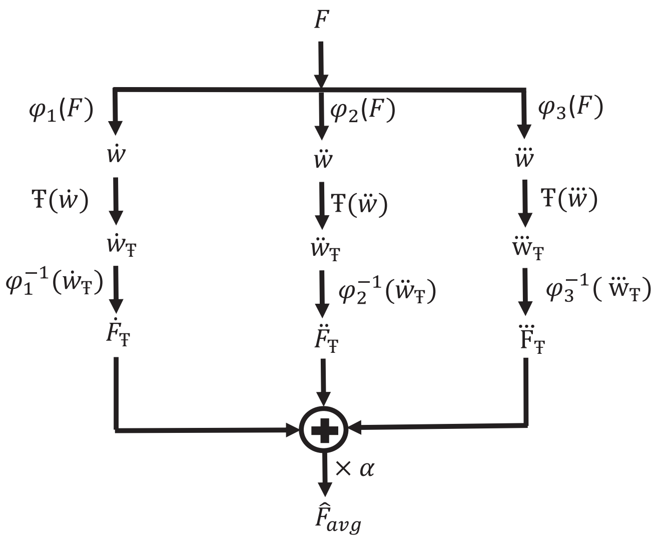

Fig. 10. Parallel de-nosing.

2.3. Enhancement

Even though adaptive group thresholding and parallel denoising do not eliminate all the babble mixed with the target speech, they alter the babble properties. Adaptive group thresholding is adjusted based on the noise level. Hence coefficients originating from target speech are less affected by the thresholding whereas coefficients originated from babble are more likely to be attenuated or set to zero. Parallel denoising and adaptive group thresholding significantly alter the babble structure and reduce it to sporadic and isolated coefficients with high frequency content. (See impulse shape coefficients in coefficient-groups c_5 after thresholding in Fig. 9) To investigate this, we measured the high frequency content of speech and noise-dominated frames, after and before denoising. Our experiments show that the energy of high frequency components remains almost constant in speech-dominated frames, after and before parallel denoising whereas it drastically increases in noise-dominated frames. To exploit the above mentioned property, after parallel denoising we apply a suitable low-pass filter only to the noise-dominated frames to remove the high frequency residual components resulting from the previous denoising steps and further enhance the speech quality. In SEDA we used a 6th order Butterworth low pass filter with cut-off frequency of 4000 Hz.

3. Methods

The effect of SEDA on speech intelligibility and sound quality was evaluated for cochlear implant users. IEEE sentences were presented against randomly generated multi-talker background noise at SNRs between 0 and 9 dB with and without SEDA processing. In the first experiment, subjects were asked to repeat as much of each sentence as they could understand. In the second experiment, subjects were asked to rate the sound quality of each sentence.

3.1. Subjects

Seven post-lingually deafened CI subjects were tested. None of the subjects had usable residual hearing in either ear without amplification. Nevertheless, all subjects who used hearing aids in daily life wore foam earplugs during the experiment. Specific subject demographics are given in Table 4.

All subjects provided informed consent in accordance with the Institutional Review Board at the New York University Langone Medical Center. For all subjects, intelligibility in quiet was measured as a reference and its average was 63.6%.

3.2. IEEE Sentences in Noise Intelligibility

IEEE standard sentences (IEEE Subcommittee, 1969) with and without processing at SNRs of 0, 3, 6, and 9 dB were used to evaluate SEDA. As a baseline, understanding of IEEE sentences in quiet were also evaluated. The noise used for testing was the 10-talker (5 male and 5 female speakers) babble randomly created from a database of 2100 sentences (excluding the sentences which were used for training the classifier) as described in Section 2.1.6. For each of the eight speech in noise conditions (processed and unprocessed at four different SNRs), 4 sentence lists were randomly selected (without replacement) from 72 male speaker IEEE standard sentence lists. An additional 2 sentence lists were also randomly selected to evaluate speech in quiet performance (i.e. with no background talkers or SEDA processing) for all subjects. The resulting 34 lists of sentences (340 sentences total) were presented in a random order to the subject. Before starting the experiment, the subject practiced the test using a randomly selected sentence list where each of the sentences was presented in a different condition. Speech material was played in free field in a double-walled sound booth at 65 dBa. Subjects were instructed to face the loudspeaker and were positioned approximately 1 m from the loudspeaker. Speech understanding was performed using *i-STAR* software (TigerSpeech Technology and Emily Fu Foundation, 2015). Subjects listened using their clinical settings of their cochlear implant. If applicable, subjects were instructed to remove their hearing aid devices during the test and wear a foam earplug. Subjects were instructed to repeat as much of the sentence as they could. Each sentence was presented only once. For each subject, the randomization process was repeated and new sentence lists were assigned. Subjects' responses were recorded and the percent correct of all words (combined key and non-key) for each condition was calculated by *i-STAR*. Note that *i-STAR* software works with databases of pre-processed audio samples. To simulate the real-time condition, we used the SEDA algorithm which receives and processes the noisy signal frame by frame without knowledge of the entire signal. The resulting denoised samples were saved for use by *i-STAR* for the listening test.

3.3. Sound quality rating

After evaluating each subject's understanding of IEEE sentences, the sound quality of the IEEE sentences in noise (with and without SEDA processing) was measured using the MUSHRA (MUltiple Stimuli with Hidden Reference and Anchor) scaling test. The open source MUSHRAM interface (Vincent, 2005) was used to conduct the experiment. Subjects were presented with a reference sound, which was speech in quiet, and were told that the quality of this sound should be rated as 100 on a scale from 0 to 100. Subjects were also presented with 10 other variations of the sentence and were asked to scale the sound quality of the speech in each of those variations along the same scale. The variations consisted of the 8 speech in noise conditions previously evaluated for intelligibility (i.e. SNRs of 0, 3, 6, and 9 dB with and without SEDA processing), an unlabeled repetition of the reference (speech in quiet) and a sample with only the background babble noise used as an anchor. Subjects were allowed to listen to each sample as many times as desired and similarly were also able to replay the reference stimulus as desired to facilitate the comparison of the sound qualities. Responses for each variation were entered by the subject using a slider in the interface. When the subject was satisfied with his/her rating of all of the samples, he/she would press a button to save all of the values and proceed to the next set of sentences. The interface used is presented in Fig. 11. The process was repeated for 5 different sentences for each subject. The sentences used were randomly selected for each subject from the 720 male speaker IEEE standard sentences. Speech material was played in free field in a double-walled sound booth at 65 dBa. Subjects were instructed to face the loudspeaker and were

Table 4
Subject information.

Subject	Age	Sex	Etiology	Ear	Implantation Year	Type of implant	Strategy / noise reduction
M107	61	M	Unknown	Left	2013	MED-EL Concert - Flex 28	FS4
				Right	N/A	N/A (Hearing Aid)[a]	N/A
N103	60	F	Genetic	Left	N/A	N/A (Hearing Aid)[a]	N/A
				Right	2008	Cochlear CI24RE (CA)	ACE
C106	39	M	Unknown	Left	N/A	N/A (Hearing Aid)[a]	N/A
				Right	2010	Advanced Bionics HiRes90K / HiFocus 1J	HiRes-S with Fidelity 120 / ClearVoice
C114	70	F	Meniere's Autoimmune	Left	N/A	N/A (Hearing Aid)[a]	N/A
				Right	2014	Advanced Bionics HiRes90K / HiFocus MS	HiRes-Optima-S / ClearVoice
C118	45	F	Ushers	Left	2010	Advanced Bionics HiRes90K / HiFocus 1J	HiRes-P with Fidelity 120 / ClearVoice
				Right	N/A	N/A (Hearing Aid)[a]	N/A
N102	64	F	Lyme Disease and head trauma	Left	N/A	N/A (Hearing Aid)[a]	N/A
				Right	2013	Cochlear Freedom CI24RE CA	ACE
C101	71	M	Unknown	Left	N/A	N/A (Hearing Aid)[a]	N/A
				Right	2012	Advanced Bionics HiRes90K / HiFocus 1J	HiRes-P with Fidelity 120 / ClearVoice

[a] Subjects were instructed to remove their hearing aid devices and insert a foam earplug during the test.

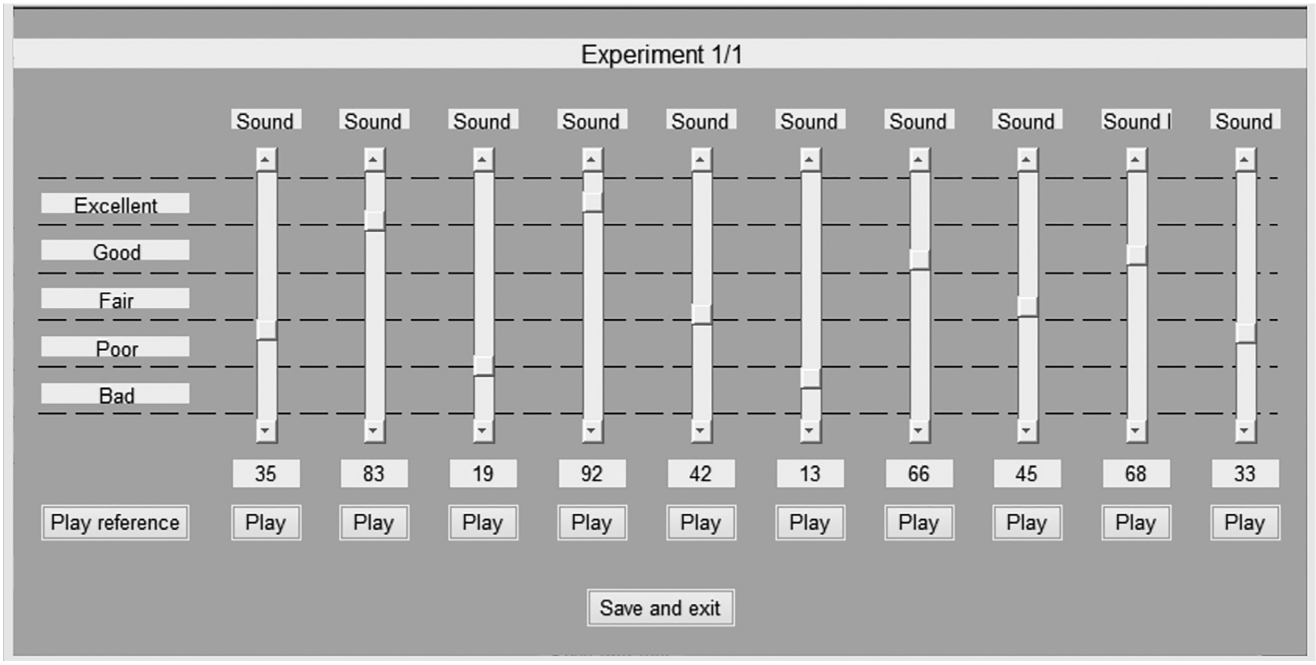

Fig. 11. Capture of the user interface for the MUSHRA experiment.

positioned approximately 1 m from the loudspeaker. Because subject C118's vision was insufficient to use the MUSHRAM interface, she was instructed to scale the sounds orally.

4. Results

4.1. Speech in noise intelligibility

The percent of words correct for each condition is presented for each subject in Fig. 12. As the test results show, the performance generally increases as a function of SNR.

Furthermore, performance with SEDA noise reduction was higher for all subjects at all SNRs except for C101 at 0 dB SNR, where word recognition was 0 for both processed and processed samples. There was, however, a great deal of variability in the magnitude of the improvement from SEDA noise reduction across subjects and SNRs. The average improvement for 0, 3, 6, and 9 dB SNR were 7.19, 11.23, 17.19, and 15.96 percentage points respectively (bottom-right panel of Fig. 12). A repeated measures two-way ANOVA detects main effects of noise reduction [F(1,6) = 31.242, p < .001] and signal to noise ratio [F(3,18) = 21.090, p < .001]. Additionally, the interaction between signal to noise ratio and noise reduction is significant [F(3,18) = 10.564, p < .001]. Post-hoc one-sample *t*-tests detected that the improvements were significant at each SNR (SNR 0 dB: t(6) = 4.605, p = .00367; SNR 3 dB: t(6) = 3.579, p = .0117; SNR 6 dB: t(6) = 5.816, p = .00113; SNR 9 dB: t(6) = 6.384, p = .000695). All four post-hoc *t*-tests remain significant after Type I error correction using Rom's method (Rom, 1990) which determines that if the p-value for all comparisons is below the critical alpha, then all comparisons are considered significant. Nevertheless, there is still room for improvement with the SEDA noise reduction algorithm. The ideal performance for a noise reduction algorithm would be to produce performance equivalent to performance in quiet (as if the all of the noise were removed without inflicting any distortion to the original speech). However, even at the highest SNR tested (9 dB SNR), performance was significantly below that of performance in quiet [t(6) = 11.664, p = .0000239] which is indicated by the purple dashed lines in Fig. 12.

4.2. Sound quality

Sound quality ratings for each condition are shown for each subject in Fig. 13. As the test results show, the sound quality generally increases as a function of SNR. Furthermore, sound quality with SEDA noise reduction was higher for all subjects. The average increase in MUSHRA scores for 0, 3, 6, and 9 dB SNR were 12.46, 11.02, 23.55, and 28.95 (bottom-right panel of Fig. 13). A repeated measures two-way ANOVA detects main effects of noise reduction [F(1,6) = 200.070, p < .001] and signal to noise ratio [F(3,18) = 36.195, p < .001]. Additionally, a significant interaction between noise reduction and signal to noise ratio was detected [F(3,18) = 3.189, p = .049]. After Type I error correction using Rom's method (Rom, 1990), post-hoc one-sample *t*-tests detected a significant improvement in sound quality at SNRs of 6 dB (t(6) = 7.089, p = .000395) and 9 dB (t(6) = 6.176, p = .000828). However, improvements in sound quality approached but failed to reach significance for SNR 0 dB (t(6) = 2.289, p = .0621) and SNR 3 dB (t(6) = 2.932, p = .0262) after Type I error correction.

5. Discussion

Considering the particularly difficult nature of the babble noise reduction for CI devices and limited number of previous works in this field, babble noise reduction is a worthwhile area for CI research. SEDA is an effort to address the babble problem for cochlear implant users. It provides intelligibility and sound quality benefits for CI users in babble noise environments by employing a new approach. SEDA uses a classifier which is specifically tuned for multi-talker babble. It also employs a new wavelet-based approach combined with parallel denoising for multi-talker babble noise reduction in cochlear implant devices.

The evaluation of SEDA suggests that it can improve both the intelligibility and sound quality of speech in the presence of multi-talker babble for CI listeners. Although post-hoc tests showed an improvement in intelligibility at all SNRs, after Type I error control, significant improvements in sound quality were only detected for SNRs of 6 and 9 dB. Although SEDA was effective at 0 dB SNR, the improvements in intelligibility and sound quality increased with larger SNRs. The smaller observed improvements in intelligibility at lower SNRs are expected to be partially caused by floor effects at lower SNRs (e.g. C101 and C114)

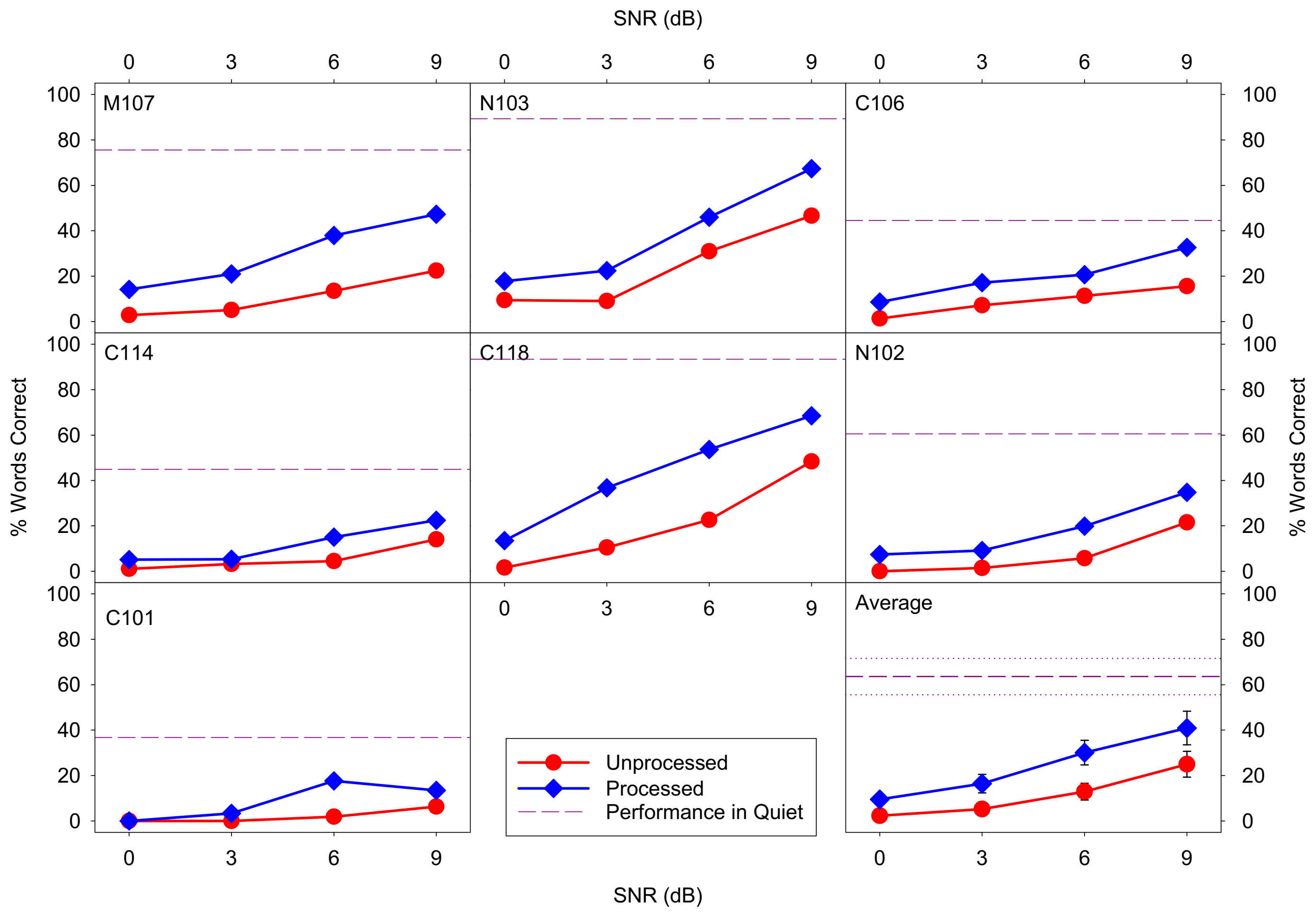

Fig. 12. Performance on the intelligibility test for all 7 subjects. The bottom right panel represents the results averaged across all subjects. All other panels represent the results from individual subjects. Red circles indicate performance with the unprocessed speech in noise samples while blue diamonds indicate performance with SEDA processing. The purple dashed line indicates performance for the subject in quiet, representing the best possible score for that subject. Error bars represent ± 1 standard error of the mean. For the average plot (bottom right), the purple dashed line indicates the average speech in quiet performance and the dotted purple lines indicate ± 1 standard error of the mean.

which cause the benefit of SEDA to be underestimated.

In designing of the experiment, it was decided that SEDA should be tested against performance with the clinical settings to determine how much additional benefit would be obtained by cochlear implant users. One ramification of this decision was that all of the Advanced Bionics users in this experiment (C101, C106, C114, and C118) already had a noise reduction algorithm (ClearVoice Medium) in their maps. Therefore, all benefits observed for these subjects in this study are in addition to any potential benefits that were obtained from ClearVoice. Had SEDA been evaluated for these subjects without additional noise reduction, we would predict that the effect of SEDA would have been greater, although it is possible that the effect might have been smaller. While removing ClearVoice would have provided a better estimate of the absolute capabilities of SEDA, keeping ClearVoice provides a better estimate of the clinical relevance of SEDA. Nevertheless, one would not expect to see a large difference between SEDA with and without ClearVoice Medium as Holden et al. (2013) failed to detect an improvement of performance with multi-talker babble using ClearVoice Medium. Note that other users did not use any noise reduction algorithm in their Med-El or Cochlear devices (see Table 4).

While SEDA has been demonstrated to be effective, there are limitations involved in the algorithm. Because SEDA is based on thresholding, the quieter components of a signal are more likely to be removed. Therefore, SEDA only works when the target speech is noticeably louder than any of the other individual background talkers. This can partially explain the poorer absolute performance observed at lower SNRs. While not formally evaluated, we expect that performance at SNRs below 0 dB would continue to degrade. Similarly, at a fixed SNR, we expect that decreasing the number of talkers would reduce the performance of the algorithm. Moreover, SEDA is less likely to perform well when the noise has a high concentration of energy in a short duration (e.g., burst noise). Further tests are required to investigate the effectiveness of SEDA for different noise situations including stationary noises (e.g. white noise), other non-stationary noises such as speech-weighted noise and multi-talker babble with smaller number of talkers.

The tested version of SEDA has a relatively high latency but there remains opportunity for improving the algorithm and optimizing the parameters to decrease the latency of SEDA. Assuming t_c, t_d and t_e are the required processing times for an incoming frame F_{in} by the three steps respectively, to maintain real time operation we should have $t_c + t_d + t_e \leq t_f$ where t_f is the duration of F_{in}. Moreover, to minimize latency, given a fixed overlap between frames, t_f should be as short as possible. As the audio processing latency produces a discrepancy between audio cues and visual cues in real time applications, it is important to minimize the latency. For example, the current version of SEDA has been evaluated with 50% overlapping Hanning windows and frame duration of 128 ms for sampling rate of 16,000 samples per second (i.e., frame length of 2048 samples). This imposes a minimum latency of 64 ms due to the frame length. However, processing time of the overlapping frames will further increase the total latency. This latency may be problematic, especially when used as a front end to a cochlear implant processor which has its own latency. Increased

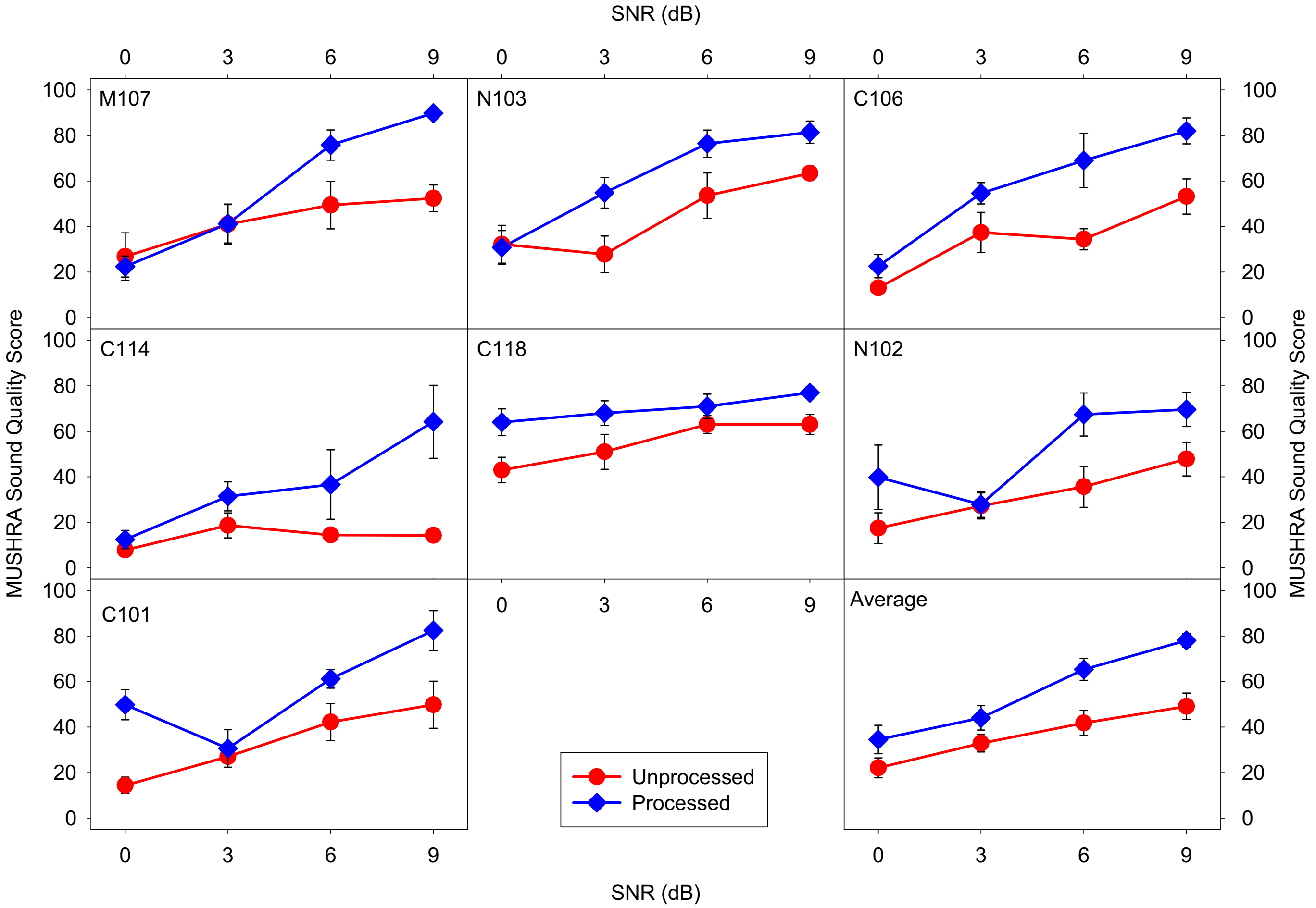

Fig. 13. Performance on the MUSHRA sound quality test for all 7 subjects. The bottom right panel represents the results averaged across all subjects. All other panels represent the results from individual subjects. Red circles indicate performance with the unprocessed speech in noise samples while blue diamonds indicate performance with SEDA processing. Error bars represent ± 1 standard error of the mean.

latencies can cause a disassociation between visual and auditory stimuli (e.g. Stevenson et al., 2012). The frame length could be reduced to a smaller number of samples to reduce latency. However, the effects of shorter frame lengths on performance have yet to be evaluated. For optimal results with a shorter time window, the SEDA classifier would need to be re-optimized accordingly. Note that a smaller number of TQWT levels (sub-bands) should be selected for shorter frames while other TQWT parameters (Q factors and redundancy) could remain unchanged.

Using different frame lengths for the classifier and denoising stages also can potentially reduce the latency of SEDA. In the present implementation, the frame length limitation is mainly due to the SEDA classifier as the effectiveness of features degrades rapidly with shorter frames. However, the denoising stage can be implemented using a shorter frame length and might be less susceptible to the shortening of frame length than the classifier. Because the frame length used for denoising stage determines the SEDA latency, using shorter frames for denoising and longer frames for classifier would reduce the SEDA latency. However, further investigation is required to evaluate the effect of reducing the frame length in the denoising stage on performance. Note that having different frame lengths for classifier and denoising stages requires modification of the SEDA algorithm.

Performance of the SEDA classifier might be enhanced for shorter frames by using additional features which are sensitive to the level of babble in speech, have a relatively low computational cost, and perform well for short frames of the noisy speech. For example, kurtosis-based features, as used in Hazrati et al. (2013), might be beneficial for a future version of the SEDA classifier.

The evaluation of SEDA is promising for CI users, especially in the context of previous work. Nevertheless, because of differences in subject population, difficulty of varying sentence corpuses, language, and testing methods, it is inappropriate to directly compare results of SEDA with other noise reduction algorithms. Further research in which each of the above factors are controlled is needed if a direct comparison between noise reduction algorithms is to be made.

The implementation of SEDA evaluated in the present manuscript was implemented on a Windows computer in a sound booth. However, for SEDA to be beneficial to cochlear implant users in their daily life, SEDA needs to be implemented on a smaller platform. Ideally, SEDA would be implemented directly into the sound processor. An alternative would be to use a smartphone as an external pre-processor to clean up the signal using SEDA and stream the signal into the sound processor.

Acknowledgments

We appreciate the efforts of all of subjects who provided their valuable time. The authors also thank Natalia Stupak for coordinating the tests. Support for this research was provided by the National Institutes of Health/National Institute on Deafness and Other Communication Disorders (R01 DC012152; PI: Landsberger) as well as an NYU School of Medicine Applied Research Support Fund internal grant.

References

Bello J.P. Spring 2016. EL9173 selected topics in signal processing: audio content analysis [Online] http://www.nyu.edu/classes/bello/ACA.html.

Bentler, R., Chiou, L.K., 2006. Digital noise reduction: an overview. Trends Amplification 10, 67–82.

Bilger, R.C., Nuetzel, J.M., Rabinowitz, W.M., Rzeczkowski, C., 1984. Standardization of a test of speech perception in noise. J. Speech Hear Res. 27, 32–48.

Bishop, C.M., 2007. Pattern Recognition and Machine Learning. Springer.

Chung, K., Zeng, F.G., Waltzman, S., 2004. Utilizing advanced hearing aid technologies as pre-processors to enhance cochlear implant performance. Cochlear Implants Int. 5 (Suppl 1), 192–195.

Dawson, P.W., Mauger, S.J., Hersbach, A.A., 2011. Clinical evaluation of signal-to-noise ratio-based noise reduction in Nucleus(R) cochlear implant recipients. Ear Hear 32, 382–390.

Duda, R.O., Hart, P.E., Stork, D.G., 2001. Pattern Classification, 2nd ed. Wiley, New York.

Fetterman, B.L., Domico, E.H., 2002. Speech recognition in background noise of cochlear implant patients. Otolaryngol. Head Neck Surg. 126, 257–263.

Friedland, D.R., Runge-Samuelson, C., Baig, H., Jensen, J., 2010. Case-control analysis of cochlear implant performance in elderly patients. Arch. Otolaryngol. Head Neck Surg. 136, 432–438.

Goehring, T., Bonler, F., Monaghan, J.J.M., van Dijk, B., Zarowski, A., Bleeck, S., 2016. Speech enhancement based on neural networks improves speech intelligibility in noise for cochlear implant users. Hearing Res. 344, 183–194.

Gu Q., Li Z., Han J. 2012. Generalized Fisher Score for feature selection. arXiv preprint arXiv:1202.3725.

Hazrati, O., Lee, J., Loizou, P.C., 2013. Blind binary masking for reverberation suppression in cochlear implants. J. Acoust. Soc. Am. 133, 1607–1614.

Healy, E.W., Yoho, S.E., Wang, Y., Wang, D., 2013. An algorithm to improve speech recognition in noise for hearing-impaired listeners. J. Acoust. Soc. Am. 134, 3029–3038.

Holden, L.K., Brenner, C., Reeder, R.M., Firszt, J.B., 2013. Postlingual adult performance in noise with HiRes 120 and clearvoice low, medium, and high. Cochlear Implants Int. 14, 276–286.

Hu, Y., Loizou, P.C., Li, N., Kasturi, K., 2007. Use of a sigmoidal-shaped function for noise attenuation in cochlear implants. J. Acoust. Soc. Am. 122 EL128-34.

IEEE Recommended Practice for Speech Quality Measurements, in IEEE No 297-1969, pp. 1–24, June 11 1969 doi:10.1109/IEEESTD.1969.7405210.

Kasturi, K., Loizou, P.C., 2007. Use of S-shaped input-output functions for noise suppression in cochlear implants. Ear Hear 28, 402–411.

Kohavi, R., 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: Proceedings of the 14th International Joint Conference On Artificial Intelligence. Montreal, Quebec, Canada. 2. Morgan Kaufmann Publishers Inc., pp. 1137–1143.

Krishnamurthy, N., Hansen, J.H.L., 2009. Babble noise: modeling, analysis, and applications. IEEE Trans. Audio, Speech, Lang. Process. 17, 1394–1407.

Loizou, P.C., Lobo, A., Hu, Y., 2005. Subspace algorithms for noise reduction in cochlear implants. J. Acoust. Soc. Am. 118, 2791–2793.

Mauger, S.J., Arora, K., Dawson, P.W., 2012. Cochlear implant optimized noise reduction. J. Neural Eng. 9, 065007.

Muller-Deile, J., Schmidt, B.J., Rudert, H., 1995. Effects of noise on speech discrimination in cochlear implant patients. Ann. Otol. Rhinol. Laryngol. Suppl. 166, 303–306.

Nilsson, M., Soli, S.D., Sullivan, J.A., 1994. Development of the hearing in noise test for the measurement of speech reception thresholds in quiet and in noise. J. Acoust. Soc. Am. 95, 1085–1099.

Powers, D.M.W., 2011. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness & correlation. J. Mach. Learn. Technol. 2, 37–63.

Reynolds, D., 2009. Gaussian Mixture Models. In: Li, S.Z., Jain, A. (Eds.), Encyclopedia of Biometrics. Springer US, pp. 659–663.

Reynolds, D.A., Quatieri, T.F., Dunn, R.B., 2000. Speaker verification using adapted gaussian mixture models. Digit. Signal Process. 10, 19–41.

Roland Jr., J.T., Gantz, B.J., Waltzman, S.B., Parkinson, A.J., Multicenter Clinical Trial, G., 2016. United States multicenter clinical trial of the cochlear nucleus hybrid implant system. Laryngoscope 126, 175–181.

Rom, D.M., 1990. A sequentially rejective test procedure based on a modified Bonferroni inequality. Biometrika 77, 663–665. http://dx.doi.org/10.1093/biomet/77.3.663.

Selesnick, I.W., 2010. A new sparsity-enabled signal separation method based on signal resonance. In: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 4150–4153.

Selesnick I.W. 2011a. Sparse signal representations using the tunable Q-factor wavelet transform, Vol. 8138. pp. 81381U-81381U-15.

Selesnick, I.W., 2011b. Wavelet transform with tunable Q-factor. IEEE Trans. Signal Process. 59, 3560–3575.

Shlens J. 2003. A tutorial on principal component analysis.

Sperry, J.L., Wiley, T.L., Chial, M.R., 1997. Word recognition performance in various background competitors. J. Am. Acad. Audiol. 8, 71–80.

Stevenson, R.A., Zemtsov, R.K., Wallace, M.T., 2012. Individual differences in the multisensory temporal binding window predict susceptibility to audiovisual illusions. J. Exp. Psychol. Hum. Percept. Perform. 38, 1517–1529.

Swets, J.A., 1988. Measuring the accuracy of diagnostic systems. Science 240, 1285–1293.

Tang, J., Aleyani, S., Liu, H., 2014. Feature Selection for Classification: A Review, Data Classification: Algorithms and Applications. CRC Press.

TigerSpeech Technology and Emily Fu Foundation. 2015. Internet-based speech testing, Assessment, & Recognition [Online] http://istar.emilyfufoundation.org/.

Toledo, F., Loizou, P., Lobo, A., 2003. Subspace and envelope subtraction algorithms for noise reduction in cochlear implants, engineering in medicine and biology society, 2003. In: Proceedings of the 25th Annual International Conference of the IEEE. Vol. 3. pp. 2002–2005 Vol. 3.

Vincent E. 2005. MUSHRAM: a MATLAB interface for MUSHRA listening tests [Online] https://members.loria.fr/EVincent/software-and-data/ (verified August 2nd, 2016).

Yang, L.P., Fu, Q.J., 2005. Spectral subtraction-based speech enhancement for cochlear implant patients in background noise. J. Acoust. Soc. Am. 117, 1001–1004.

Ye, H., Deng, G., Mauger, S.J., Hersbach, A.A., Dawson, P.W., Heasman, J.M., 2013. A wavelet-based noise reduction algorithm and its clinical evaluation in cochlear implants. PLoS One 8, e75662.

Yue, H.H., Tomoyasu, M., 2004. Weighted principal component analysis and its applications to improve FDC performance, Decision and Control, 2004. In: CDC. 43rd IEEE Conference on. Vol. 4. pp. 4262–4267 Vol. 4.