FAST NOISE TRACKING BASED ON RECURSIVE SMOOTHING OF MMSE NOISE POWER ESTIMATES

Jan S. Erkelens and Richard Heusdens

Delft University of Technology, Dept. of Mediamatics, Mekelweg 4, 2628 CD, Delft, The Netherlands
Email: {j.s.erkelens, r.heusdens}@tudelft.nl

ABSTRACT

We consider estimation of the noise spectral variance from speech signals contaminated by highly nonstationary noise sources. In each time frame, for each frequency bin, the noise variance estimate is updated recursively with the Minimum Mean-Square Error (MMSE) estimate of the current noise power. For the estimation of the noise power, a spectral gain function is used, which is found by an iterative data-driven training method. The proposed noise tracking method can accurately track fast changes in noise level (up to about 10 dB/s). When compared to the Minimum Statistics method for various noise sources in a speech enhancement system, improvements in segmental signal-to-noise ratio of more than 1 dB are obtained.

Index Terms— Speech enhancement, acoustic noise, tracking filters, least mean square methods.

1. INTRODUCTION

Single-channel speech enhancement methods based on the Discrete Fourier Transform (DFT) have received significant interest due to their low complexity and relatively good performance, e.g., [1–4]. These and other methods need accurate estimates of the noise spectrum to perform optimally. Many natural noise sources are nonstationary and it is therefore necessary to make reliable noise spectrum estimates also during speech activity. It is a challenging problem to avoid speech power leaking into the noise spectrum estimates. In recent years, several methods for tracking of nonstationary noise sources have appeared in the literature. Rangachari and Loizou [5] give an overview. An idea that has been proven quite successful is to track the minima of the smoothed noisy spectrum [6, 7]. Several methods make use of some kind of minimum tracking procedure [5–9].

The limited ability of these and other methods to instantaneously follow rapid increases in noise level stems from having to use a rather long time window for the minimum tracking in order to avoid speech leakage. In Section 3 we will propose to use Minimum Mean-Square Error (MMSE) estimation of the noise power to update the noise spectrum estimates with a reduced risk of speech leakage. The MMSE estimates are obtained with the standard method of multiplying the noisy powers by a spectral gain function. This removes most of the speech contribution from the noisy spectrum, allowing for fast and accurate tracking of changing noise levels. The spectral gain function for noise power estimation is found by an iterative data-driven method. The proposed noise tracking method can follow changes in noise level up to about 10 dB/s. We will evaluate the proposed method in Section 4 and compare with the Minimum Statistics (MS) method, in terms of tracking performance and overall performance in a speech enhancement scheme.

The research is supported by MultimediaN.

2. MODELING ASSUMPTIONS AND DEFINITIONS

2.1. Spectral modeling

We consider an additive-noise signal model of the form

$$X(k,m) = S(k,m) + N(k,m), \tag{1}$$

where $X(k,m)$, $S(k,m)$, and $N(k,m)$ are complex-valued random variables representing the short-time DFT coefficients at frequency index k in signal frame m from the noisy speech, clean speech and noise process, respectively. We apply the standard assumption that $S(k,m)$ and $N(k,m)$ are statistically independent across time and frequency as well as from each other. For ease of notation we therefore drop the time and/or frequency index when this does not cause confusion. The noisy speech amplitude is $R = |X|$, the speech spectral amplitude is $A = |S|$, and the noise amplitude $D = |N|$. The noise DFT coefficients N are assumed to follow a complex Gaussian distribution with variance λ_N. We will call D^2 the *(instantaneous) noise power*. Its expectation is λ_N. Similarly, the speech spectral variance λ_S is the expectation of A^2. The *prior SNR* ξ and the *posterior SNR* ζ are defined as

$$\xi(k,m) = \frac{\lambda_S(k,m)}{\lambda_N(k,m)}, \quad \zeta(k,m) = \frac{R^2(k,m)}{\lambda_N(k,m)} . \tag{2}$$

2.2. Amplitude estimation

Any power A^p of the speech amplitude can be estimated as

$$\widehat{A^p} = G_{A^p}(\xi,\zeta)R^p , \tag{3}$$

where G_{A^p} is a suitable gain function depending on the assumed statistical models for the speech and the noise and on the criterion that is optimized for. Later on, we will estimate the noise power D^2 by means of a gain function G_{D^2}.

3. USING MMSE ESTIMATION OF THE NOISE POWER TO REDUCE SPEECH LEAKAGE

Many noise tracking algorithms have difficulty in tracking fast increases in noise level [5]. For example, the rather slow response to such increases of methods based on minimum statistics is a result of using a window of considerable length in order to prevent speech power from leaking into the noise variance estimates. The key idea to the method proposed below is to avoid using the noisy power R^2 directly by removing as much as possible of the speech contribution from it, *before* smoothing with an exponential smoother. That is, we propose to estimate the noise variance as follows:

$$\hat{\lambda}_N(k,m) = \alpha_s(k,m)\hat{\lambda}_N(k,m-1) + (1-\alpha_s(k,m))\widehat{D^2}(k,m). \tag{4}$$

The smoothing parameter $\alpha_s(k,m)$ depends on an estimate $\hat{p}(k,m)$ of speech presence probability:

$$\alpha_s(k,m) = \alpha_d + (1-\alpha_d)\hat{p}(k,m), \tag{5}$$

where α_d is fixed between 0 and 1 (All parameter settings are given in Section 4.1.1). Using $\widehat{D^2}$ instead of R^2 causes less speech power to leak into the noise variance estimate. Consequently, the speech presence probability estimator does not need to be extremely accurate, the smoothing parameter does not need to be close to 1 very frequently and faster tracking can be achieved. For $\widehat{D^2}$ we will use the MMSE estimator of the noise power D^2. An iterative data-driven method is used to find the optimal gain function G_{D^2} (Section 3.4.1). An adaptive smoothing parameter such as (5) has been used before, e.g., in [5,9], but here a simplified estimation procedure for $\hat{p}$ will be used that allows for faster tracking.

3.1. Speech presence probability estimation

We propose the following estimator of $\hat{p}$. First, the posterior SNR is smoothed over a few neighboring frequency bins to take into account the strong correlation of speech presence in neighboring frequency bins [9]:

$$\tilde{\zeta}(k,m) = \sum_{i=-w}^{w} b(i)\zeta(k-i,m), \quad \text{with} \sum_{i=-w}^{w} b(i) = 1 \tag{6}$$

Next, a hard decision about speech presence is made:

$$I(k,m) = \begin{cases} 1 \quad \text{(speech presence)} & : \ \tilde{\zeta}(k,m) > T(k,m) \\ 0 \quad \text{(speech absence)} & : \ \tilde{\zeta}(k,m) \le T(k,m) \end{cases} \tag{7}$$

The speech presence probability estimate is updated with $I(k,m)$:

$$\hat{p}(k,m) = \alpha_p \, \hat{p}(k,m-1) + (1-\alpha_p)I(k,m), \tag{8}$$

with α_p between 0 and 1. This estimate is used in (5) to find the smoothing parameter α_s. This procedure for calculating α_s is similar to that in [5]. There, the ratio of the smoothed noisy spectrum and its local minimum is compared in (7) against a threshold. The local minimum in [5] is tracked with an adaptation time of about 0.5 seconds for non-stationary noise. Here we only use the posterior SNR of the current time frame in (7), and we can therefore react almost instantaneously to changing noise levels. The parameter T controls the trade-off between the tracking speed and the amount of speech leakage. The higher its value, the faster the tracking speed, but the higher the risk of speech leakage.

3.2. Prior SNR estimation

The gain functions take the prior and posterior SNRs as arguments. These parameters are unknown in practice and have to be estimated. We have found that the noise tracking performance depends on the particular prior SNR estimator used. While the "decision-directed" estimator [2] is very suitable for speech spectral amplitude estimation, we found that a modified estimator (Section (3.2.2)) improves noise tracking performance. We will therefore use different estimators for the speech estimation and noise tracking tasks.

3.2.1. Prior SNR estimator $\hat{\xi}_{SE}$ for Speech Enhancement

For speech estimation, the "decision-directed" estimator [2] is used, with a bias correction [10]:

$$\hat{\xi}_{SE}(k,m) = \alpha_{SE}\frac{\widehat{A^2}(k,m-1)}{\hat{\lambda}_N(k,m)} + (1-\alpha_{SE})[\frac{R^2(k,m)}{\hat{\lambda}_N(k,m)} - 1], \tag{9}$$

where we will use the latest available estimate of the noise variance $\hat{\lambda}_N(k,m)$.

3.2.2. Prior SNR parameter $\hat{\xi}_{NT}$ for Noise Tracking

Errors in the estimated noise variance will affect the estimated prior and posterior SNR. However, the decision-directed prior SNR estimates will be affected more than the posterior SNR estimates for the following reason. If λ_N is overestimated (underestimated), ξ and ζ will be underestimated (overestimated). This means that the gain function $G_{A^2}(\hat{\xi}_{SE}, \hat{\zeta})$ will suppress too much (too little), causing the errors in $\hat{\lambda}_N$ and $\widehat{A^2}$ to become negatively correlated. Therefore, an error in $\hat{\lambda}_N$ will tend to amplify itself in the first term of (9). The errors in numerator and denominator are also negatively correlated in the second term, because $R^2/\hat{\lambda}_N - 1$ equals $(R^2 - \hat{\lambda}_N)/\hat{\lambda}_N$. As input to G_{D^2}, we will therefore use the following parameter $\hat{\xi}_{NT}$, which is less sensitive to errors in $\hat{\lambda}_N$:

$$\hat{\xi}_{NT}(k,m) = \alpha_{NT}\frac{R^2(k,m-1)}{\hat{\lambda}_N(k,m)} + (1-\alpha_{NT})\frac{R^2(k,m)}{\hat{\lambda}_N(k,m)}. \tag{10}$$

It is clear that $\hat{\xi}_{NT}$ is not an unbiased estimator of λ_S. However, this is not a problem, because the gain function G_{D^2} will be adapted to this parameter by means of a data-driven method (Section 3.4) and the bias will be compensated for.

3.3. Safety net

In Section 4, we will show that our noise tracker can easily follow very fast changes in noise level up to about 10 dB/s. However, if the noise level increases even much faster than that, for example, when it suddenly jumps to a high level and stays at that level, $\tilde{\zeta}$ in (6) will be calculated on the basis of a noise variance estimate which is too low. It therefore becomes more likely that a speech presence decision is made in (7), and the algorithm will react more slowly. We therefore propose a simple and effective safety net, which ensures that the algorithm continues to work properly also under such extreme conditions. The idea is to push the noise variance estimate into the right direction when we detect that its value is much too low. As a reference value, we use the minima $P_{min}(k,m)$ of the smoothed values $\overline{P}(k,m)$ of the noisy power $R^2(k,m)$ in a short window of length w_{min}, where $\overline{P}(k,m)$ is given by

$$\overline{P}(k,m) = \eta\overline{P}(k,m-1) + (1-\eta)R^2(k,m), \tag{11}$$

where η is a smoothing parameter close to 0. After updating $\hat{\lambda}_N$ with (4), we check whether it fulfills the following condition:

$$B \cdot P_{min}(k,m) < \hat{\lambda}_N(k,m), \tag{12}$$

where $B > 1$ is a correction factor. In case of a large increase in noise level that the algorithm cannot follow, $B \cdot P_{min}(k,m)$ will become larger than $\hat{\lambda}_N(k,m)$ after a time of the order of the window length. If that happens, we reset the $\hat{\lambda}_N(k,m)$ values that violated (12) to $\max\left[B \cdot P_{min}(k,m), \hat{D^2}(k,m)\right]$, and the corresponding $\hat{p}(k,m)$ to 0. We have observed that the value of B and the window length are not very critical for good performance, but a window length of at least 0.5 seconds is required.

3.4. Finding the gain function for noise power estimation

For $\widehat{D^2}$ we would ideally like to use the MMSE estimator. However the optimal gain function is very hard to derive analytically. The main reason is that the input parameters to the gain function depend on the quantity $\hat{\lambda}_N(k,m)$, which must be computed using the gain function. In other words, a nonlinear recursion is introduced which is usually ignored in the analytical derivation of gain functions. We therefore resort to a data-driven method to find the gain function. We will make use of the method in [10], in an iterative fashion (Section 3.4.1). This method makes no explicit assumptions about the speech statistics and can also take into account the influence of estimation inaccuracies in the estimated speech and noise variances. The method of [10] is briefly recapitulated first.

3.4.1. Data-driven gain optimization

A large training database of speech material is used, contaminated with various levels of stationary white Gaussian noise of known SNR. For all training data, the prior and posterior SNRs are calculated for every time frame and every frequency index. Their values are discretized on a grid, typically in 1 dB steps. Each ($\hat{\xi}$,ζ)-pair has a corresponding (D^2,R^2)-pair associated with it. Statistics are collected for all training data and afterwards one scalar gain value G_{D^2} is computed for each grid cell such that the mean-square error between the D^2 and $\widehat{D^2}$ is minimized.

The grid used in this paper covers the range [-19 dB, 40 dB] for ξ and [-30 dB, 40 dB] for ζ, both in steps of 1 dB. The training speech data consisted of about 25% of the TIMIT-TRAIN database. To each file, white noise has been added at several SNRs, from -12.5 dB to 27.5 dB in steps of 5 dB. Noise only frames are not taken into account: frames with a clean energy more than 40 dB below the maximum clean frame energy of a speech sentence are not taken into account for optimization of the gain function.

During the training with the above method, the noise variance λ_N is known. When the resulting gain table G_{D^2} is used with the noise tracking method outlined earlier in this section, the noise variance is unknown but is estimated using G_{D^2}. Fortunately, the gain function can still be optimized while taking into account this recursion by means of an iterative scheme explained next.

The idea is to iteratively update the gain function G_{D^2}, while taking into account the estimation of λ_N. Let the value of any quantity in the i-th iteration be denoted by a subscript i. For example, $G_{D^2,i}$ is the gain function for D^2-estimation in the i-th iteration. To break the recursion, $G_{D^2,i}$ is only used to compute data to be used in the *next* iteration. The input parameters of $G_{D^2,i}$, $\hat{\xi}_{NT,i}(k,m)$ and $\hat{\zeta}_i(k,m)$, depend only on data computed in the *previous* iteration. The optimization procedure is as follows:

0) Initialization ($i=0$): $\widehat{D^2}_i(k,m) = D^2(k,m)$, $\hat{\lambda}_{N,i}(k,m) = \lambda_N(k,m)$
1) Compute $\hat{\zeta}_i, \hat{\xi}_{NT,i}$: $\hat{\zeta}_i(k,m) = R^2(k,m)/\hat{\lambda}_{N,i}(k,m)$, $\hat{\xi}_{NT,i}(k,m) = \alpha_{NT}\frac{R^2(k,m-1)}{\hat{\lambda}_{N,i}(k,m)} + (1-\alpha_{NT})\frac{R^2(k,m)}{\hat{\lambda}_{N,i}(k,m)}$
   Collect (D^2, R^2) statistics per grid cell;
   Update $\alpha_{s,i}(k,m)$ and $\hat{\lambda}_{N,i}$: $\hat{\lambda}_{N,i}(k,m+1) = \alpha_{s,i}(k,m)\hat{\lambda}_{N,i}(k,m) + (1-\alpha_{s,i}(k,m))\widehat{D^2}_i(k,m)$
   $m := m+1$;
   Complete step 1) for all training data;
2) Minimize the MSE in $\widehat{D^2}$ for each grid cell $\Rightarrow G_{D^2,i+1}(\xi_{NT},\zeta)$
3) Compute data for the next iteration: $\widehat{D^2}_{i+1}(k,m) = G_{D^2,i+1}(\hat{\xi}_{NT,i}(k,m),\hat{\zeta}_i(k,m))R^2(k,m)$
4) $i := i+1$; Go to step 1) if $i < i_{max}$.

This scheme typically converges in less than $i_{max} = 7$ iterations.

The $\widehat{D^2}_i$ are initialized with the true noise powers D^2. Alternatively, they can be initialized with the noisy power R^2 or even the speech power A^2. In all cases does the gain table converge to the same end result. The question arises how this scheme optimizes for the practical case when there are recursions (i.e., the output of the gain function in the current time frame is used in the calculation of the inputs for the next time frame). Convergence means that $G_{D^2,i}$ changes less and less from one iteration to the next when i increases. It also means that the differences between $\widehat{D^2}_{i+1}$ and $\widehat{D^2}_i$ become smaller and smaller. But when $\widehat{D^2}_{i+1}$ and $\widehat{D^2}_i$ become almost equal, we have nearly the *same* situation as *with* the recursion. In fact, we have verified that applying the optimized gain functions from step 2) to the training data recursively decreases the mean-square error with each iteration step for all intializations [11]. Iterative optimization of the gain function decreased the MSE by 65% compared to non-iterative optimization with the basic method in [10].

4. EXPERIMENTAL RESULTS

4.1. Experimental set up

To evaluate the noise tracking performance of our method, we concatenated 8 sentences from the TIMIT-TEST database, without intervening pauses (about 29 seconds of speech). Four male and four female speakers have been used. All signals have been limited to 8 kHz sampling frequency and telephone bandwidth (300-3400 Hz). The noise recordings have been taken from the ETSI EG 202 396-1 Background Noise Database [12]. In addition, computer-generated white noise is used. Noise tracking performance is measured directly and also in an enhancement system (DFT-based MMSE speech spectral amplitude estimation under a generalized-Gamma speech prior [4]). We used 50%-overlapping frames of 32 ms and a cosine-squared analysis window.

4.1.1. Parameter settings

The following parameter settings are used in the experiments: α_d in (5) is set to 0.85, $\alpha_p = 0.1$ in (8), and $T(k,m) = 4$ in (7) independent of time and frequency. We have used $w = 1$ and $b(i) = 1/(2w+1)$ in (6). The same value 0.98 is used for the smoothing parameters α_{SE} in (9) and α_{NT} in (10). Both $\hat{\xi}_{SE}$ and $\hat{\xi}_{NT}$ are limited in this work to values larger than $\xi_{min} = -19$ dB. We use $\eta = 0.1$ in (11), $B = 1.5$ in (12), and the length of w_{min} spans 0.8 seconds. A generalized-Gamma speech amplitude prior with $\gamma = 1$ and $\nu = 1$ is used to derive $\widehat{A^2}$ and $\hat{A}$ [4].

4.1.2. Performance measures

Two quality measures will be used to evaluate the noise variance estimation. The first one is the segmental logarithmic estimation error $LogErr$, defined as [13]:

$$LogErr = \frac{1}{|\mathcal{M}|K}\sum_{m\in\mathcal{M}}\sum_k \left| 10\log_{10}\left[\frac{\overline{\lambda}_N(k,m)}{\hat{\lambda}_N(k,m)}\right]\right|, \quad (13)$$

where K is the number of frequency bins. We left out frames which don't contain noise in the computation of $LogErr$, that is, frames with a noise energy more than 40 dB below the noise energy of the frame with maximum noise energy were not included in the index set $\mathcal{M}$. $|\mathcal{M}|$ is the cardinality of $\mathcal{M}$. This measure is used instead of the relative estimation error, because it penalizes errors at increasing and decreasing noise levels more symmetrically. The relative estimation error has been found to correlate poorly with subjective preference

Table 1. $LogErr$ [dB] of estimated noise variance for various noise types and levels, obtained using MS and the proposed method.

SNR	Stat. WGN		Car interior		Traffic		Train station	
[dB]	MS	Prop.	MS	Prop.	MS	Prop.	MS	Prop.
0	1.00	0.68	1.43	1.12	2.14	0.89	1.89	1.05
5	1.15	0.79	1.72	1.49	2.26	1.06	2.07	1.23
10	1.26	0.93	2.20	2.00	2.32	1.34	2.22	1.54
15	1.45	1.12	3.15	2.70	2.40	1.79	2.48	2.01
20	1.74	1.50	4.60	3.59	2.77	2.41	3.06	2.62

tests [5]. Prior to evaluating the distortion, the true noise power is smoothed in time [7]:

$$\overline{\lambda}_N(k,m) = 0.9\overline{\lambda}_N(k,m-1) + 0.1D^2(k,m). \qquad (14)$$

$\overline{\lambda}_N$ is used as the ideal reference in (13). Objective enhancement quality was measured by means of the improvement in Segmental Signal-to-Noise Ratio over the noisy signal ($SSNR+$). Only frames that contain speech are taken into account.

4.2. Evaluations

4.2.1. Highly nonstationary white Gaussian noise

The proposed noise tracking method allows for very fast noise tracking. As a first example, a highly nonstationary white noise has been added to the speech material. The noise level is varied between SNRs of 0 and 20 dB and changes at a maximum rate of 0.16 dB/frame, i.e., 10 dB/s. Fig. 1 shows the ideal reference noise power level (thin solid line), and the estimated noise levels from our method (dashed line) and MS (dotted line). All results are averages over all frequency bins. As expected, MS cannot track the rapid increases in noise level. The performance measures for MS are $LogErr = 3.84$ dB and $SSNR+ = 3.01$ dB. Our method can handle both fast increases and decreases in noise level, resulting in much better performance figures: $LogErr = 1.46$ dB and $SSNR+ = 4.36$ dB.

4.2.2. Other noise sources

Tables 1-2 show the performance measures for our method and MS, for speech contaminated with stationary white Gaussian noise (WGN), interior car noise (at 100 km/h), traffic noise, and train station noise, at overall SNRs of 0, 5, 10, 15, and 20 dB. The car, traffic, and train station noises are from the ETSI database [12]. The traffic and train station noises were the most nonstationary, with passing vehicles and the arrival of a train. Our method clearly outperforms the MS method in terms of $LogErr$ and $SSNR+$. We achieve the largest improvements for the most nonstationary noise sources.

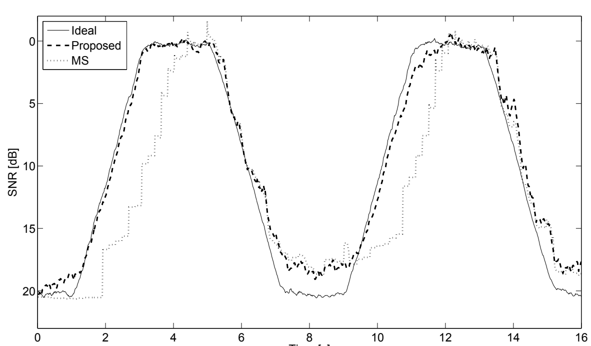

Fig. 1. Noise tracking performance of the proposed method and MS on speech contaminated with highly-nonstationary white Gaussian noise. The noise level changes at a maximum rate of 10 dB/s.

Table 2. $SSNR+$ [dB] of enhanced speech for various noise types and levels, obtained using MS and the proposed method.

SNR	Stat. WGN		Car interior		Traffic		Train station	
[dB]	MS	Prop.	MS	Prop.	MS	Prop.	MS	Prop.
0	8.35	7.94	7.86	7.83	5.41	6.67	5.40	6.42
5	6.54	6.60	6.20	6.48	4.12	5.31	4.09	5.04
10	4.92	5.21	4.64	4.94	3.02	3.91	2.90	3.64
15	3.34	3.70	3.22	3.37	1.97	2.63	1.84	2.30
20	1.93	2.25	1.93	2.06	0.89	1.40	0.75	1.21

5. CONCLUSION

Fast and accurate tracking of highly nonstationary noises becomes feasible with smoothing of MMSE noise power estimates. The recursive nature of the estimation problem is dealt with by means of an iterative data-driven gain function optimization method.

6. REFERENCES

[1] S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 27, no. 2, pp. 113–120, 1979.

[2] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 32, no. 6, pp. 1109–1121, 1984.

[3] R. Martin, "Speech enhancement based on minimum mean-square error estimation and supergaussian priors," *IEEE Trans. Speech, Audio Processing*, vol. 13, no. 5, pp. 845–856, 2005.

[4] J. S. Erkelens, R. C. Hendriks, R. Heusdens, and J. Jensen, "Minimum mean-square error estimation of discrete Fourier coefficients with generalized-Gamma priors," *IEEE Trans. Acoustics, Speech and Language Processing*, vol. 15, no. 6, pp. 1741–1752, 2007.

[5] S. Rangachari and P. C. Loizou, "A noise-estimation algorithm for highly non-stationary environments," *Speech Communication*, vol. 48, pp. 220–231, 2006.

[6] R. Martin, "Spectral subtraction based on minimum statistics," in *Proc. EUSIPCO*, 1994, pp. 1182–1185.

[7] R. Martin, "Noise power spectral density estimation based on optimal smoothing and minimum statistics," *IEEE Trans. Speech, Audio Processing*, vol. 9, no. 5, pp. 504–512, 2001.

[8] G. Doblinger, "Computationally efficient speech enhancement by spectral minima tracking in subbands," in *Proc. Eurospeech*, 1995, pp. 1513–1516.

[9] I. Cohen, "Noise spectrum estimation in adverse environments: improved minima controlled recursive averaging," *IEEE Trans. Speech, Audio Processing*, vol. 11, no. 5, pp. 466–475, 2003.

[10] J. S. Erkelens, J. Jensen, and R. Heusdens, "A data-driven approach to optimizing spectral speech enhancement methods for various error criteria," *Speech Communication, Special Issue on Speech Enhancement*, vol. 49, pp. 530–541, 2007.

[11] J. S. Erkelens and R. Heusdens, "Tracking of nonstationary noise based on data-driven recursive noise power estimation," *submitted to the IEEE Trans. Acoustics, Speech and Language Processing*, 2007.

[12] "ETSI EG 202 396-1 background noise database," available online: http://portal.etsi.org/docbox/STQ/Open/ EG 202 396-1 Background noise database/.

[13] R. C. Hendriks, J. Jensen, and R. Heusdens, "DFT-domain subspace-based noise tracking for speech enhancement," in *Proc. Interspeech*, Antwerp, Belgium, 2007, pp. 830–833.