EUROSPEECH'93
Berlin, Germany, September 19-23, 1993

AN EFFICIENT ALGORITHM TO ESTIMATE THE INSTANTANEOUS SNR OF SPEECH SIGNALS

Rainer Martin

Institute for Communication Systems and Data Processing (IND), Aachen University of Technology,
Templergraben 55, 52056 Aachen, Germany, Phone: +49 241 806984, Fax: +49 241 806985

ABSTRACT

This contribution presents an efficient algorithm to estimate the instantaneous signal-to-noise ratio of speech signals. The algorithm is capable to track non stationary noise signals and has a low computational complexity. It does not need a speech activity detector nor histograms to learn signal statistics. The algorithm is based on the observation that a noise power estimate can be obtained using minimum values of a smoothed power estimate. This paper will present this algorithm, its performance, its limits, and some applications.

Keywords: SNR, time delay estimation, speech enhancement

1. INTRODUCTION

Instantaneous SNR estimation is an essential component of speech processing algorithms which are sensitive to varying noise levels. An instantaneous SNR estimate is based on short time power estimates with time constants of integration in the range of 0.02 - 0.1 s. Typical applications are time delay estimation and speech enhancement (e.g. spectral subtraction).

To acquire noise statistics the conventional approach to SNR estimation employs a voice activity detector to extract the noise only segments of the disturbed speech signal. The identification of noise segments might be based on the signal power, on a statistical evaluation by means of histograms or on combinations thereof [1]. In all cases the update of the noise power estimate requires a signal segment where no speech is present. Depending on the method tracking of varying noise levels might be slow and confined to periods of no speech activity.

The proposed algorithm, however, does not need an explicit speech/nospeech decision to gather noise statistics and is capable to track varying noise levels during speech activity. The algorithm is based on the observation that the smoothed power estimate of a noisy speech signal exhibits distinct peaks and valleys (see Figure 1). While the peaks correspond to speech activity the valleys of the smoothed noise estimate can be used to obtain a noise power estimate. To estimate the noise floor our algorithm takes the minimum of a smoothed power estimate within a window of finite length. The SNR estimates obtained by this method are fairly accurate.

In section 2 and 3 we will present the algorithm and discuss some of its statistical properties. Section 4 will present experimental results. We conclude in section 5 with two applications.

2. DESCRIPTION OF ALGORITHM

In what follows we assume that the bandlimited and sampled disturbed signal $x(i)$ is a sum of a speech signal $s(i)$ and a noise signal $n(i)$, $x(i) = s(i) + n(i)$, where i denotes the time index. We further assume that $s(i)$ and $n(i)$ are statistically independent, hence $E\{x^2(i)\} = E\{s^2(i)\} + E\{n^2(i)\}$.

$SNR_x(i)$ will denote the estimated signal-to-noise ratio of signal $x(i)$ at time i. The algorithm works on a sample basis, i.e. a new output sample $SNR_x(i)$ is computed for each input sample $x(i)$.

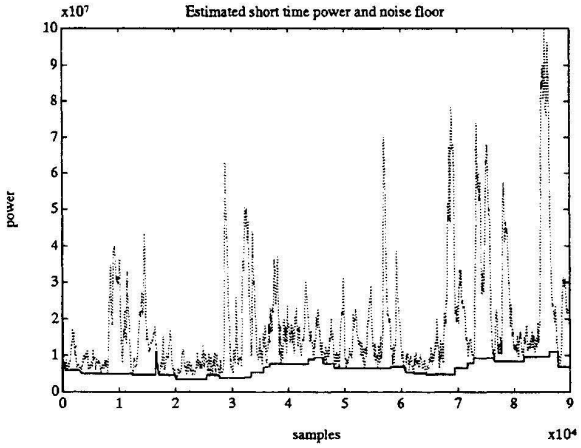

Figure 1: Smoothed power and estimated noise floor of noisy speech signal (f_s=8kHz, segmental SNR ca. 5 dB, car noise)

The computation of $SNR_x(i)$ is based on a noise power estimate $P_n(i)$ which is obtained as the minimum of the smoothed short time power estimate $\tilde{P}_x(i)$ within a window of L samples.

Besides initialization the algorithm can be split into three major parts which will be discussed below (see Figure 2):

1. Computation of a smoothed short time power estimate $\bar{P}_x(i)$ of signal $x(i)$
2. Computation of the noise power estimate $P_n(i)$
3. Computation of the $SNR_x(i)$

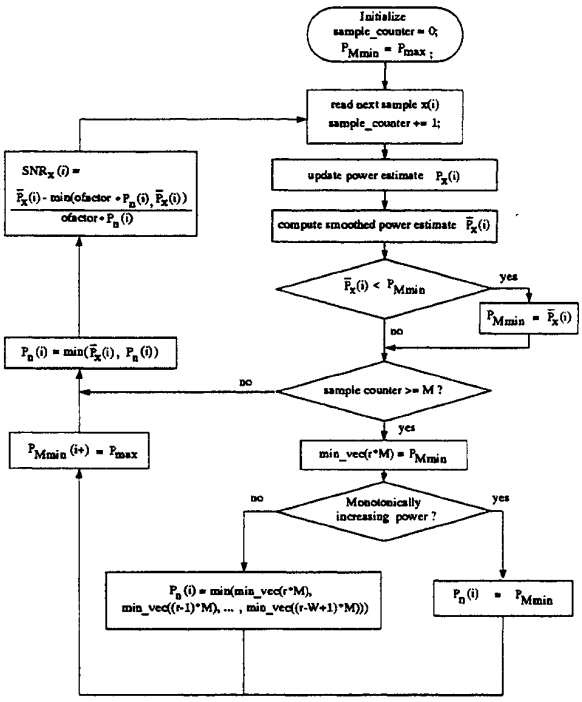

Figure 2: Flowchart of the SNR estimation algorithm

Computation of a smoothed power estimate

Computation of the short time signal power $P_x(i)$ and smoothing of the power estimate is done in two steps. The power estimate may be obtained recursively or non-recursively. We here use a sliding rectangular window of length N with N=128. In many applications, however, a power estimate is already available.

Let $\bar{P}_x(i)$ denote the smoothed short time power estimate at time i. Smoothing of the power estimate is done by means of a first order recursive system. The smoothing constant is typically set to values between $\alpha = 0.95...0.98$. The recursion for $i > N$ is given by equation 1:

$$\begin{aligned} P_x(i) &= P_x(i-1) + x(i) * x(i) - x(i-N) * x(i-N) \\ \bar{P}_x(i) &= \alpha * \bar{P}_x(i-1) + (1-\alpha) * P_x(i) \end{aligned} \tag{1}$$

Noise power estimation

The noise power estimate is based on the minimum of signal power within a window of L samples. For reasons of computational complexity and delay the data window of length L is decomposed into W windows of length M such that $M * W = L$. For a sampling rate of f_s=8 kHz typical window parameters are M=1250 and W=4, thus L=5000 corresponding to a time window of 0.625 s.

The minimum power of the last M samples is found by a samplewise comparison of the actual minimum $P_{Mmin}(i)$ and the smoothed power $\bar{P}_x(i)$.

Whenever M samples have been read, i.e. $i = r * M$, we store the minimum power of the last M samples and reset $P_{Mmin}(i = r * M)$ to its maximum value: $P_{Mmin}(i = r * M+) = P_{max}$.

To determine the noise power estimate we distinguish two cases:

1. slowly varying noise power,
2. rapidly varying noise power.

If the minimum power of the last W windows with M samples each is monotonically increasing we decide on rapid noise power variation. In this case the noise power estimate equals the power minimum of the last M samples $P_n(i) = P_{Mmin}(i = r * M)$.

In case of non monotonic power the noise power estimate is set to the minimum of the length L window, i.e.: $P_n(i) = P_{Lmin}(i)$. The minimum power of the length L window is easily obtained as the minimum of the last W minimum power estimates:

$$\begin{aligned} P_{Lmin}(i) = min(&P_{Mmin}(i = r * M), \\ &P_{Mmin}(i = (r-1) * M), \\ &..., P_{Mmin}(i = (r-W+1) * M)) \end{aligned} \tag{2}$$

If the actual smoothed power is smaller than the estimated noise power $P_n(i)$ the noise power is updated immediately independent of window adjustment: $P_n(i) = min(\bar{P}_x(i), P_n(i))$.

Computation of SNR

The estimated SNR is computed on the basis of the estimated minimum noise power $P_n(i)$. A factor $ofactor$ accounts for the fact that the minimum power estimate is smaller than the true noise power. $ofactor$ is typically set to values between 1.3 and 2 (see section 3):

$$SNR(i) = 10 * log_{10}\left(\frac{\bar{P}_x(i) - min(ofactor * P_n(i), \bar{P}_x(i))}{ofactor * P_n(i)}\right) \tag{3}$$

Figure 1 plots the smoothed power estimate and the estimated noise floor for a noisy speech sample. The window length $L = M * W$ must be large enough to bridge any peak of speech activity, but short enough to follow non stationary noise variations. Experiments with different speakers, different languages, and modulated noise signals have shown that a window length of 0.625 s is a good value.

In case of slowly varying noise power the update of noise estimates is delayed by $L + M$ samples. If a rapid noise power increase is detected this delay is reduced to M samples, thus improving the noise tracking capability of the algorithm.

3. STATISTICS OF MINIMUM ESTIMATES

In this section we compute the density function of the minimum noise power estimate and justify our choice of the overestimation factor $ofactor$. To facilitate the analytical evaluation of minimum estimates we assume that the noise process n is zero mean white Gaussian noise with variance σ^2 and that the computation of the smoothed power estimate is entirely done by means of non recursive accumulation, i.e.:

$$P_x(i) = \sum_{m=0}^{N-1} x^2(i-m) \qquad (4)$$

Then, the power estimate $P_x(i)$ is chi-square distributed [2] with mean $N * \sigma^2$ and density:

$$f_{P_x}(y) = \frac{1}{(\sigma\sqrt{2})^N \Gamma(N/2)} * y^{N/2-1} * e^{-y/2\sigma^2} * U(y) \quad (5)$$

where $\Gamma()$ and $U()$ denote the Gamma function and the unit step function, respectively.

The density of the minimum of L_w independent power estimates is given by [2]:

$$f_{min}(y) = L_w * (1 - F_{P_x}(y))^{L_w-1} * f_{P_x}(y) \qquad (6)$$

where $F_{P_x}(y)$ denotes the distribution function of the chi-square density:

$$F_{P_x}(y) = 1 - e^{-y/2\sigma^2} * \sum_{m=0}^{N/2-1} \frac{1}{m!} * \left(\frac{y}{2\sigma^2}\right)^m * U(y) \quad (7)$$

Clearly, successive values of $P_x(i)$ are correlated but if we shift the sliding window of equ. 4 by $\Delta i > N/2$ we obtain sufficiently uncorrelated power estimates.

Figure 3 plots the density functions $f_{P_x}(y)$ and $f_{min}(y)$ and corresponding histograms of $\bar{P}_x(i)$ and $P_n(i)$ for a car noise signal.

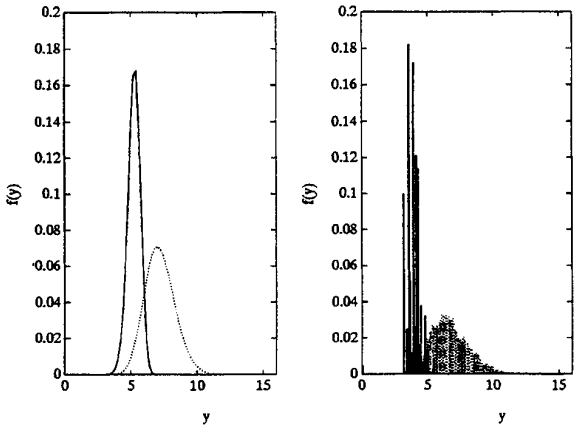

Figure 3: Density functions $f_{P_x}(y)$ (dotted) and $f_{min}(y)$ (solid) for $\sigma^2 = 0.09$, $N = 80$, and $L_w = 20$ (left graph) and corresponding histograms of $\bar{P}_x(i)$ (dotted) and $P_n(i)$ (solid) for car noise signals (right graph)

We now choose the overestimation factor $ofactor$ such that the noise power estimate is approximately unbiased, i.e. $E\{P_n\} * ofactor \approx E\{P_x\}$. Since $f_{P_x}(y)$ and $f_{min}(y)$ are scaled by the noise variance σ^2 $ofactor$ does not depend on σ^2. Figure 4 shows the dependency of $ofactor$ on N and L_w and allows the selection of an appropriate overestimation factor.

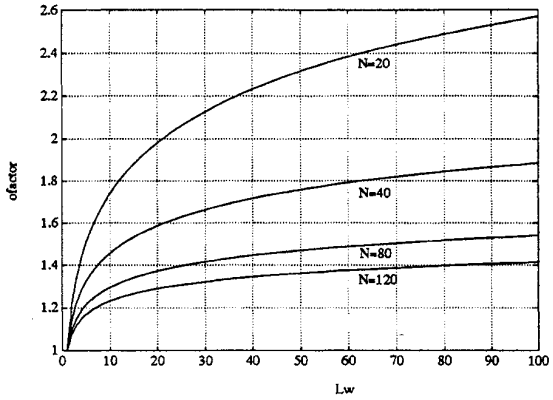

Figure 4: Overestimation factor $ofactor$ versus N and L_w

4. EXPERIMENTAL RESULTS

Figure 5 plots the true and the estimated instantaneous SNR of the same noisy speech signal as in Figure 1. The true SNR was computed on the basis of separate speech and noise signals. Our SNR estimate shows good agreement with the true SNR during speech activity. In agreement with the statistical evaluation the estimate is biased when no speech is present.

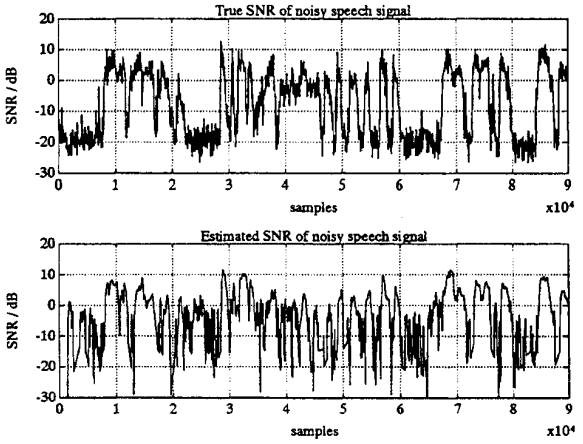

Figure 5: True and estimated instantaneous SNR of noisy speech signal ($ofactor = 1.5$)

To test the algorithm with non stationary noise the noise signal was modulated with a sine function and then added to a speech signal: $x(i) = s(i) + n(i) * \left(1.5 + sin\left(\frac{2*\pi*0.33*i}{8000}\right)\right)$. The modulation frequency was set to $f_m = 0.33\ Hz$.

Figure 6 plots the corresponding short time power and the estimated noise floor. Note the delay of the noise power values in case of increasing noise power. Figure 7 shows the true and estimated SNR. Due to the window length of 0.625 s rapid noise variations might result in erroneous SNR estimates.

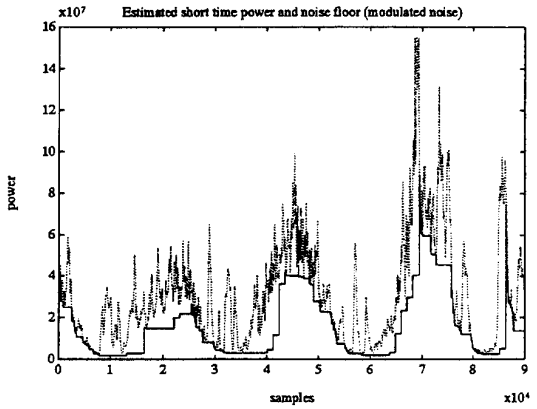

Figure 6: Short time power of modulated noisy speech signal and noise estimate for f_m=0.33 Hz

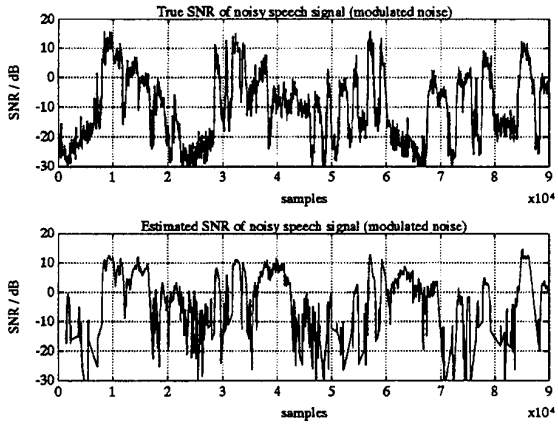

Figure 7: True and estimated SNR of modulated noisy speech signal for f_m=0.33 Hz

5. APPLICATIONS

The algorithm was tested with varying noise levels and successfully incorporated in several speech processing systems. In what follows we briefly discuss two applications, namely time delay estimation and spectral subtraction.

TIME DELAY ESTIMATION

Time delayed speech signals originate e.g. from microphone arrays where the speaker is in a non symmetric position relative to the array and possibly moving. In-phase summation or adaptive processing of these microphone signals usually requires a time delay compensation.

The SNR estimator was implemented to support time delay estimation by means of (generalized) correlation. To determine the delay between microphone signals we compute the maximum of a smoothed cross correlation estimate. Whenever the SNR is below a preset threshold the update of smoothed correlation functions is frozen. Figure 8 plots the delay estimate without and with SNR estimation. The enhanced algorithm clearly eliminates all large deviations of the time delay estimate.

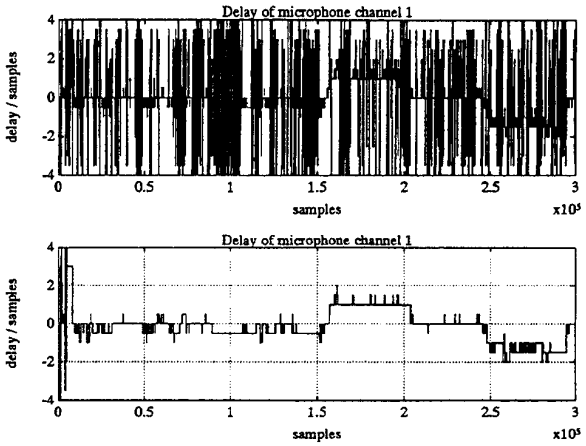

Figure 8: Time delay of microphone channel 1 with respect to channel 2 of a noisy speech sample with moving speaker without (upper graph) and with (lower graph) SNR estimation.

SPECTRAL SUBTRACTION

To reduce the noise level within a disturbed speech signal the spectral subtraction method modifies the short time spectral magnitude of the disturbed speech signal. In our experiments we used a filter bank with 256 channels and estimated the minimum power in each of these channels.

Our informal listening test reveal relatively few annoying musical tones. However, due to the fact that we subtract slightly biased noise power estimates (ofactor = 1.5) the noise suppression is limited. Power spectra of the disturbed and of the improved signal show an improvement of about 10 dB.

6. CONCLUSION

Varying noise levels have a significant impact on the performance of many speech processing algorithms. The algorithm proposed in this paper provides a computational inexpensive and effective mean to cope with this problem. The algorithm is accurate for medium to high SNR conditions but necessarily biased when no speech is present. A priori knowledge of noise variation and noise correlation is helpful to adapt window length and to control the estimation bias.

ACKNOWLEDGMENTS

Part of this work was supported by Philips Kommunikations Industrie, Germany. Spectral subtraction using minimum power estimates was investigated by Peter Kocybik.

References

[1] R. McAulay and M. Malpass: "Speech Enhancement Using a Soft-Decision Noise Suppression Filter ", IEEE Trans. ASSP, Vol. 28, No. 2, pp. 137-145, April 1980.

[2] A. Papoulis: "Probability, Random Variables, and Stochastic Processes", 2nd ed., McGraw-Hill, 1984.