Spectral Envelope Estimation Based on Deep Neural Networks and its use for Speech Reconstruction

Christopher Seitz, Mohammed Krini
Technische Hochschule Aschaffenburg, 63743 Aschaffenburg, Germany
Email: christopher.seitz@t-online.de, mohammed.krini@th-ab.de

Abstract

The speech quality achieved by conventional noise suppression methods at high noise conditions is often unsatisfactory. By recovering highly disturbed speech components with speech reconstruction methods, the overall speech quality can be further improved. The speech reconstruction method presented in this paper is based on the so-called source-filter model of speech production. The focus in this contribution will be on the estimation of the vocal tract filter characteristics (spectral envelope) at high noise conditions as this has been proved to be very important for speech reconstruction. For this purpose a deep recurrent neural network (Deep-RNN) which operates as a regression model for given noise features is utilized. The performance of the trained Deep-RNN is compared with a priori trained spectral envelope codebooks. Subsequently, a synthetic speech signal is generated using the envelope estimates of the Deep-RNN which is then combined adaptively with a conventionally noise reduced signal depending on the SNR. The quality of the resulting enhanced speech is analyzed with an objective measure, the log-spectral distance (LSD), as well as with subjective tests. Both tests indicate a significant quality improvement compared to conventional schemes – especially in high noise conditions.

Introduction

In recent years speech enhancement has attracted a considerable amount of research attention. Many real-world applications require a good performance, for example, hearing aids, mobile speech communication and robust speech recognition. There are various noise reduction techniques developed in the last few decades. These methods can be classified by the number of microphones that were used. Single-channel techniques, such as spectral subtraction or Wiener filtering, are now widely used and find use in many applications [1]. Multi-channel systems make use of the spatial structure of speech and noise [2]. All of these methods have in common that the quality of the estimated clean speech is improved for middle and high signal-to-noise ratios (SNR). In case of strong disturbances, often no reliable improvement can be achieved, due to speech distortions and artifacts caused by the noise reduction. The goal of the speech reconstruction is to improve the intelligibility and quality of strongly disturbed speech. The speech reconstruction method presented in this paper is based on the so-called source-filter model of speech production [3, 4].

Based on the source-filter model of vocal production, a speech signal can be split into two components. The source part describes the signal generated by the vocal cords. This signal is then filtered by the resonance characteristics of the vocal tract. As a result, any speech signal can be described by an excitation signal (source part) and a spectral envelope (filter part) [5]. For a successful reconstruction, a reliable estimation of the spectral envelope is of great importance. The focus in this contribution will be on the estimation of the spectral envelope used for enhanced speech reconstruction.

Two methods, on the one hand the codebook method and on the other hand a Deep-RNN, are compared and analyzed for the spectral envelope estimation. Afterwards a synthetic speech signal is generated using the envelope estimates which is then combined adaptively with a conventionally noise reduced signal depending on the SNR. The quality of the resulting enhanced speech is analyzed with an objective measure, the log-spectral distance, as well as with subjective tests.

Analysis Filterbank

It is assumed that the microphone signal $y(n)$ is additively composed of the speech signal $s(n)$ and the noise $b(n)$: $y(n) = s(n) + b(n)$. First, the speech signal is processed with an analysis filterbank. This filterbank splits the signal into overlapping blocks. Each block is then weighted with a window function $h_{\text{ana},k}$ and transformed into the frequency domain by means of a discrete fourier transformation (DFT):

$$Y_\mu(n) = \sum_{k=0}^{N-1} y(nr-k)\; h_{\text{ana},k}\; e^{-j\Omega_\mu k}, \tag{1}$$

with frame length $N = 512$, frame shift $r = 128$, sampling frequency $f_{\text{s}} = 16000$ Hz, and sub-band index $\mu = 0, ..., N-1$.

Conventional Noise Suppression

The Wiener filter algorithm aims to suppress noise with dynamic adjusted attenuation coefficients $G_\mu(n)$ applied in the frequency domain as

$$\hat{S}_{\text{nr},\mu}(n) = Y_\mu(n)\, G_\mu(n), \tag{2}$$

resulting in the frequency domain representation of the clean speech estimate, $\hat{S}_{\text{nr},\mu}(n)$. There are several improvements to the Wiener filter algorithm. One includes information from a previous frame into the speech enhancement process and is known as the recursive Wiener filter:

$$G_\mu(n) = \max\left\{G_{\min}, 1 - \frac{\hat{S}_{bb,\mu}(n)}{G_\mu(n-1)|Y_\mu(n)|^2}\right\}, \tag{3}$$

where $\hat{S}_{bb,\mu}(n)$ represents the estimated noise power spectral density (PSD) [6] and $G_{\min}$ is the maximum attenuation.

Speech Activity Detection

For speech activity detection (SAD), the signal-to-noise ratio (SNR) is determined by the input signal in the frequency domain [6]. For the extraction of the SNR, the following ratio is used:

$$\widetilde{SNR}(\mu,n) = \frac{\hat{S}_{ss}(\mu,n)}{\hat{S}_{bb}(\mu,n)}. \tag{4}$$

$\hat{S}_{ss}(\mu,n)$ represents the estimate PSD of the clean speech signal: $\hat{S}_{ss}(\mu,n) = |Y_\mu(n)|^2 - \hat{S}_{bb}(\mu,n)$. At this point it should be mentioned that estimation errors of the noise PSD can result in negative values for the SNR. The problem is counteracted by: $SNR(\mu,n) = \max\{0, \widetilde{SNR}(\mu,n)\}$. The mean SNR is determined to detect the speech activity:

$$\overline{SNR}(n) = \frac{1}{\mu_1 - \mu_0 + 1} \sum_{\mu=\mu_0}^{\mu_1} SNR(\mu,n). \tag{5}$$

The frequency bins μ_0 and μ_1 correspond to the real frequencies 100 Hz and 4000 Hz. Finally, the speech activity detector is determined as follows:

$$c_{\text{SAD}}(n) = \begin{cases} 1, & \text{if } \overline{SNR}(n) > 0.3, \\ 0, & \text{else.} \end{cases} \tag{6}$$

Preliminary Envelope Estimation

A simple way to estimate the spectral envelope is to smooth the magnitude input spectrum in the forward and backward frequency. This bi-directional smoothing is realized by a first order infinite impulse response (IIR) filter. The frequency smoothing in the positive frequency direction can be described as follows:

$$\overline{Y'}_\mu(n) = \begin{cases} |Y_\mu(n)|, & \text{if } \mu = 0, \\ \lambda_f \overline{Y'}_{\mu-1}(n) + (1-\lambda_f)|Y_\mu(n)|, & \text{else.} \end{cases} \tag{7}$$

Then a smoothing in the negative frequency direction is performed in an analogous way, leading to $\overline{Y''}_\mu(n)$. The smoothing constant is chosen by $\lambda_f = 0.8$.

Codebook Approach

The speech data for the codebook and the Deep-RNN are from the CSTR VCTK corpus [7]. Disturbed speech signals with various noise types are generated. By varying the noise power and adding the speech signal to the noise signal, different SNRs can be generated. Fig. 1 shows the function overview for the codebook approach. In the upper part the training process is shown. For the codebook training undisturbed speech signals $s_\text{t}(n)$ are used. These speech signals undergo feature extraction which calculates the logarithmized and averaged envelope $A_\mu^{(s_\text{t})}(n)$. The training uses the *k-Means* algorithm [8]. Whose goal is to assign the input data $A_\mu^{(s_\text{t})}(n)$ into $N_\text{cb} \in \{64, 256, 1024\}$ cluster. The sum of the square deviations from the cluster center should be minimal. The following cost function is iteratively minimized to find the optimal clusters:

$$K_\text{cb} = \sum_{n=0}^{N_a-1} \sum_{k=0}^{N_\text{cb}-1} \sum_{\mu=0}^{N/2-1} \varphi_{nk}^{(j)} \left(A_\mu^{(s_\text{t})}(n) - C_\mu^{(j)}(k) \right)^2. \tag{8}$$

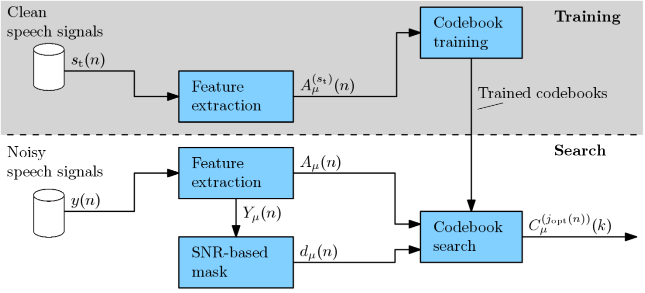

Figure 1: Function overview for the codebook training (top) as well as the search (below) of the spectral envelopes.

$C_\mu^{(j)}(k)$ describes the centroids of the clusters at the time of the iteration j. The binary mask $\varphi_{nk}^{(j)} \in \{0,1\}$ indicates to which cluster the input characteristics $A_\mu^{(s_\text{t})}(n)$ belong. N_a describes the number of spectral envelopes used for training.

In the lower part of Fig. 1 the function overview for codebook search is displayed. For the feature extraction, disturbed speech signals with an SNR in the range 0–15 dB are used. Then the best codebook entry, based on the square distance, is selected as follows:

$$k_\text{opt}(n) = \underset{0 \le k \le N_\text{cb}-1}{\operatorname{argmin}} \sum_{\mu=0}^{N/2} d_\mu(n) \big(A_\mu(n) - C_\mu^{(j_\text{opt}(n))}(k) \big)^2. \tag{9}$$

$C_\mu^{(j_\text{opt}(n))}(k)$ describes the prototype envelope, which is the best replacement for the noisy envelope $A_\mu(n)$. The feature extraction is shown in more detail in Fig. 2. The speech signals $y(n)$ are first processed with an analysis filterbank and transformed into the frequency domain. Subsequently, speech activity detection takes place for the codebook approach. It estimates envelopes only in blocks in which speech activity is detected. Afterwards the envelopes $\overline{Y''}_\mu(n)$ are logarithmized and the mean is subtracted.

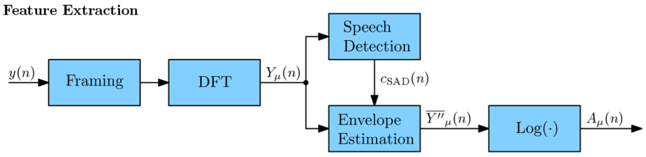

Figure 2: Procedure for the feature extraction.

Deep Recurrent Neural Network

As with the codebook approach, speech signals from the same corpus are used for training the Deep-RNN. In doing so, both the envelopes during speech activity and speech pauses are used for the training. Fig. 3 shows the function overview. The upper part represents the procedure for the training. For the supervised training, clean and noisy speech features are needed. Envelope pairs with different SNRs in the range 0–20 dB are generated and Deep-RNN models are trained with different parameters. The parameters that have been varied are the number of layers, neurons, and time steps. Whereas the number of time steps indicate the number of envelopes $(A_\mu(n), A_\mu(n-1), \ldots)$ that are used for estimating the clean envelope. Furthermore, Deep-RNN models with the *Leaky ReLU* [9] activation function and the *ELU* [10] activation function are trained. In all models the *Long Short-Term Memory* (LSTM) cells were used. LSTM cells have shown to be very robust against *exploding gradient* [11]. Then He-initialization and L2-regularization are applied to all trained models [11]. A normalization [12] of the training data, as well as dropout [13] and a layer normalization [14] are only used for some models. The following cost function is used to optimize the model parameters of the Deep-RNN:

$$F_{\mathrm{rnn}} = \frac{1}{N_{\mathrm{bs}}} \frac{1}{N/2} \sum_{n=0}^{N_{\mathrm{bs}}-1} \sum_{\mu=0}^{N/2} \left(A_{\mathrm{est},\mu}(n) - A_\mu^{(s_{\mathrm{t}})}(n)\right)^2. \quad (10)$$

$A_{\mathrm{est},\mu}(n)$ describes the estimated envelope of the Deep-RNN in the training phase. N_{bs} denotes the size of the minibatch [11]. In the lower part of Fig. 3 the estimation procedure is shown. Within the feature extraction only those blocks with speech activity are considered. The Deep-RNN then estimates the clean envelope $A_{\mathrm{RNN},\mu}(n)$ from the noisy envelope $A_\mu(n)$.

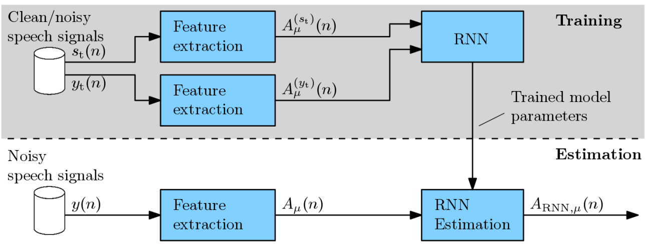

Figure 3: Functional overview for the Deep-RNN: The upper part shows the training sequence and the lower part the estimation structure of the spectral envelopes using trained Deep-RNN model parameters.

Speech Reconstruction

For generating the synthetic signal spectrum $\hat{S}_{\mathrm{syn},\mu}(n)$, the excitation signal spectrum extracted from clean speech data is used and combined with the spectral envelope estimates:

$$\hat{S}_{\mathrm{syn},\mu}(n) = \exp\left(A_{\mathrm{rec},\mu}(n) + E_\mu^{(s_{\mathrm{t}})}(n) + m^{(y_{\mathrm{t}})}(n)\right), \quad (11)$$

where $A_{\mathrm{rec},\mu}(n) \in \left\{A_{\mathrm{RNN},\mu}(n), C_\mu^{(j_{\mathrm{opt}}(n))}(k), \overline{Y''}_\mu(n)\right\}$, $m^{(y_{\mathrm{t}})}(n)$ describes the current mean of the noisy envelope, and $E_\mu^{(s_{\mathrm{t}})}(n)$ represents the logarithmized spectrum of the excitation signal. Finally, the synthetic and the conventionally noise reduced signals are combined adaptively depending on the SNR:

$$\hat{S}_{\mathrm{rec},\mu}(n) = \begin{cases} \hat{S}_{\mathrm{syn},\mu}(n), & \text{if } d'_\mu(n) = 1 \wedge \mu < \mu_{\mathrm{g}}, \\ \hat{S}_{\mathrm{nr},\mu}(n), & \text{else,} \end{cases} \quad (12)$$

where the parameter μ_{g} describes the cutoff frequency of 2000 Hz. $d'_\mu(n)$ is a binary mask used to indicates subbands with low SNR for speech reconstruction. Then the reconstructed speech signal $\hat{S}_{\mathrm{rec},\mu}(n)$ is processed with a synthesis filterbank leading to $\hat{s}(n)$.

Objective Evaluation

In order to compare the two methods, a large number of speech signals with different SNR, ranging from 0–15 dB, are generated. The envelopes are estimated in the low frequency range 0–2000 Hz, since the noise in the vehicle has the strongest effect in this range. Fig. 4 shows the LSD of the different RNN models. The LSD measure is defined as follows:

$$K_{\mathrm{lsd}} \quad (13)$$
$$= \frac{1}{N_{\mathrm{TD}}} \sum_{n=0}^{N_{\mathrm{TD}-1}} \sqrt{\frac{1}{N_{\mathrm{g}}(n)} \sum_{\mu=0}^{\mu_{\mathrm{g}}-1} d'_\mu(n)\left(A_\mu^{(s_{\mathrm{t}})}(n) - A_\mu(n)\right)^2},$$

N_{TD} denotes the number of blocks of all test data. $A_\mu(n)$ describes the estimated envelopes of the Deep-RNN models or codebooks. $N_{\mathrm{g}}(n) = \sum_{\mu=0}^{\mu_{\mathrm{g}}-1} d'_\mu(n)$ is the sum of the binary mask $d'_\mu(n)$. The model that performed best used the *Leaky ReLU* activation function with all optimizations of the RNN. Two layers with 32 neurons each and 4 time steps proved to be favorable. Fig. 5 compares the three methods used for spectral envelope estimation. The black line describes the simple procedure of IIR smoothing. It can be seen that does not perform as well as the other methods at high noise levels. Below is the codebook approach with different sizes. At high SNR the mean deviation decreases with increasing codebook size. At low SNR the search algorithm has difficulty selecting the appropriate codebook entry. This is due to the fact that fewer frequency bins with a satisfactory SNR are available for the codebook search. It also shows that the estimated envelopes of the Deep-RNN with the *Leaky ReLU* activation function, normalization, 2 layers, 32 neurons and 4 time steps perform best.

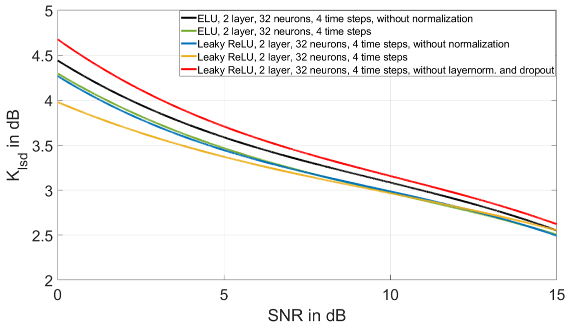

Figure 4: LSD for different Deep-RNN models.

Subjective Evaluation

For the subjective evaluation, 10 people including audio experts are asked to carry out 2 tests for speech quality and intelligibility. In the first test 10 highly disturbed speech signals with SNR=0 dB are used. A speech reconstruction based on the 3 different spectral envelope estimations (Deep-RNN, IIR, CB) is compared to a conventional noise reduction method, the Wiener filter. In all 3 subtests, the proposed speech reconstruction performed better. In the second test, the different speech

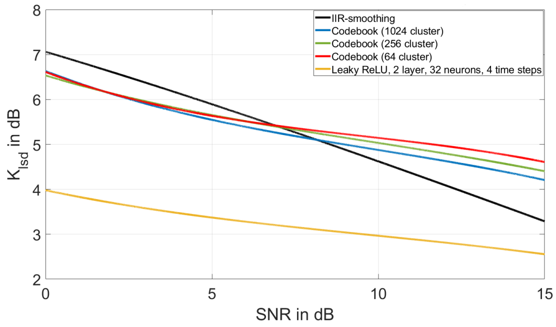

Figure 5: LSD for codebook, Deep-RNN, and IIR-smoothing.

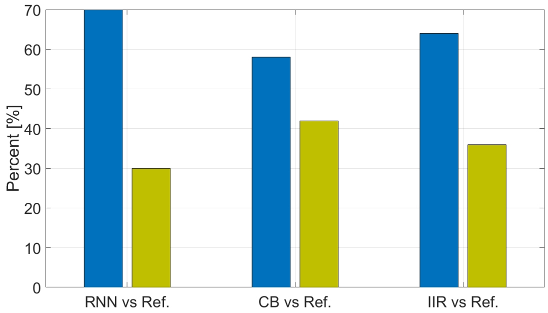

Figure 6: Compare of adaptive combination vs. Wiener filter.

reconstruction methods are compared and the subjects are asked to select the best reconstruction method. As before, 10 speech signals from 5 different female and male speakers are used. It can be seen that the speech reconstruction based on the Deep-RNN has the best result. This is because artefacts can be heard during the reconstruction with the codebook approach and a little noise during the reconstruction with the IIR filter.

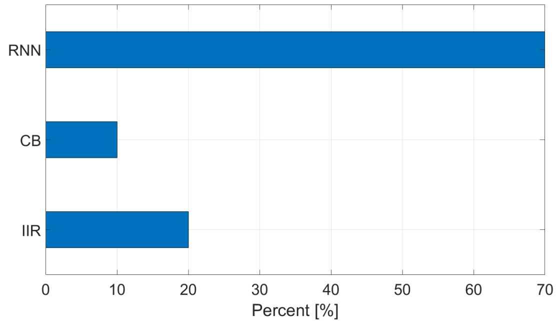

Figure 7: Reconstructed signals with different methods.

Conclusion

In this contribution two methods for estimation of spectral envelopes are presented and analyzed. The codebook method uses the *k-Means* algorithm to train the prototype envelopes. The codebook approach is compared to a deep recurrent neural network. Different models are trained to estimate the clean envelope from the noisy envelope. Furthermore, the temporal correlation of speech is exploited, in which the past envelopes are taken into account. This is realized through the temporal unfolding of the Deep-RNN. The model with the *Leaky ReLU* activation function, two layers with 32 neurons each and 4 time steps cut off best, both in comparison with the codebook and the IIR smoothing. Additionally the subjective tests have shown a slight overall improvement of speech reconstruction compared to conventional noise reduction methods. Also by direct comparison of the different speech reconstructions the Deep-RNN performed best.

References

[1] N. Wiener: The Extrapolation, Interpolation, and Smoothing of Stationary Time Series with Engineering Applications. MIT Press, 1949

[2] M. Kajala, and M. Hamalainen: Filter-and-Sum Beamformer with Adjustable Filter Characteristics. 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing, pp. 2917-2920 vol.5, 2001

[3] P. Hannon, M. Krini, and I. Schalk-Schupp: Advanced Speech Anhancement with Partial Speech Reconstruction. 21st Euopean Signal Processing Conference, pp. 1-5, 2013

[4] M. Krini, and G. Schmidt: Model-based Speech Enhancement. In E. Hänsler, G. Schmidt (eds.), Speech and Audio Processing in Adverse Environments, Springer, pp. 89-134, 2008

[5] J. Deller, J. Hansen, and J. Proakis: Discrete-Time Processing of Speech Signals. IEEE Press, 2000

[6] I. Cohen: Noise Spectrum Estimation in Adverse Environments: Improved Minima Controlled Recursive Averaging. IEEE Transactions on Speech and Audio Precessing, 2003

[7] C. Veaux, J. Yamagishi, and K. MacDonald: CSTR VCTK Corpus. University of Edinburgh, 2012

[8] S. P. LLoyd: Least Squares Quantization in PCM. IEEE Transactions on Information Theory, pp. 129-137 vol. 28, 1982

[9] A. L. Maas, A. Y. Hannun, and A. Y. Ng: Rectifier Nonlinearities Improve Neural Network Acoustic Models. ICML, 2013

[10] D. Clevert, T. Unterthiner, and S. Hochreiter: Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). CoRR, abs/1511.07289, 2015

[11] A. Géron: Hands-On Machine Learning with Scikit-learn and Tensorflow. O'Reilly Media, 2017

[12] P. Jain, and H. Hermansky: Improved Mean and Variance Normalization for Robust Speech Recognition. ICASSP, pp. 4012-4015 vol. 6, 2001

[13] Y. Gal, and Z. Ghahramani: A Theoretically Grounded Application of Dropout in Recurrent Neural Networks. arXiv e-prints, arXiv:1512.05287, 2015

[14] J. Lei Ba, J. R. Kiros, and G. E. Hinton: Layer Normalization. arXiv e-prints, arXiv:1607.06450, 2016