

Threshold Reduction for Improving Sparse Coding Shrinkage Performance in Speech Enhancement

Neda Faraji*, S. M. Ahadi*, S. Saloomeh Shariati**

Department of Electrical Engineering

* Amirkabir University of Technology, **Iran University of Science and Technology,
Tehran, Iran

nfaraji@cic.aut.ac.ir , sma@aut.ac.ir, ssaloomeh_shariati@ee.iust.ac.ir

Abstract

In this paper, we modify the Sparse Coding Shrinkage (SCS) method with an appropriate optimal linear filter (Wiener filter) in order to improve its efficiency as a speech enhancement algorithm.

SCS transform is only applicable for sparse data and speech features do not have this property in either time or frequency domains. Therefore we have used Linear Independent Component Analysis (LICA) to transfer the corrupted speech frames to the sparse code space in which noise and speech components are separated by means of a shrinkage function. Before employing SCS, Wiener filtering was applied on the ICA components to reduce noise energy and consequently the SCS shrinkage threshold. Experimental results have been obtained using connected digit database TIDIGIT contaminated with NATO RSG-10 noise data.

1. Introduction

The primary purpose of noise compensation methods applied in the context of speech processing is to reduce the effect of any signal which is alien to and disruptive of the message and to extract original speech as pure as possible. Depending on the application, speech enhancement methods aim at speech quality improvement and or speech or speaker recognition. Some common noise compensation methods are a) Spectral Subtraction, b) least mean square (LMS), adaptive filtering, c) filter-based parametric approaches, d) Hidden Markov Model (HMM)-based speech enhancement techniques. Wavelet transform has also been employed in speech enhancement systems during recent years [1].

In this paper we focus on modifying a relatively new method that is Sparse Coding Shrinkage. It has been used in [2, 3] for image denoising and in [4] for speech enhancement. The advantages of this method with respect to other popular methods can be summarized as: 1) Most algorithms apply Fourier transform, discrete Fourier transform or Karhunen-Loeve transform which facilitate the estimation of the clean speech model parameters. However, SCS is based on a data-driven transformation that is highly conformed to the structure of clean speech data. 2) Most methods just amend the distorted amplitude of the speech signal leaving the phase unprocessed. On the other hand, experimental results have shown that phase parameter plays a relatively important role in speech quality [5]. In the SCS method, the speech frame is uniformly processed without need to separating the amplitude and phase data. Independent Component Analysis is a basic method for blind source separation. In Linear Independent Component Analysis (LICA), the goal is to transfer the

observed data to the independent source space. Assuming that all independent sources are supergaussian, ICA technique will be equivalent to sparse coding method [2].

Denoising process includes an offline training stage in which clean speech frames are employed for estimating the ICA transform matrix and required parameters of the shrinkage function. These estimated functions can then be used to extract the clean components from the noisy ones.

The organization of the paper is as follows: First we explain in detail each of the methods used for speech enhancement in section 2. Section 3 demonstrates the noise reduction capability of the proposed algorithm through the computer simulations. Finally, conclusions are given in the last section.

2. Algorithms used in the proposed enhancement method

Sparse Coding Shrinkage is carried out in 2 stages:

- 1) Transferring data to the sparse coding space that is performed using ICA in this paper.
- 2) Computing Shrinkage function for each sparse code that can be achieved through Maximum Likelihood Estimation.

In this section we have provided details about the employed algorithms.

2.1. Independent Component Analysis

Linear Independent component analysis mixing model can be formulated as [6]:

$$\mathbf{x} = \mathbf{A}\mathbf{s} \quad (1)$$

We can rewrite the Eq. (1) as:

$$\mathbf{x} = \sum_i \mathbf{a}_i s_i \quad (2)$$

where \mathbf{a}_i , \mathbf{x} and s_i are basis functions, observed vector and independent components respectively. This ICA model is a generative model, i.e. it describes how the observed data are generated by a process of mixing the components s_i . After estimating the matrix \mathbf{A} , we can compute its inverse, say \mathbf{W} , and obtain the independent components simply by:

$$\mathbf{s} = \mathbf{W}\mathbf{x} \quad , \quad \mathbf{W} = \mathbf{A}^{-1} \quad (3)$$

In mixing model, \mathbf{A} and \mathbf{s} are both unknown and should be solely estimated using observed data \mathbf{x} . One estimation method is to use probabilistic characteristics of components that are assumed independent in ICA method. It has been shown that component independency in ICA mixing model is

directly related to maximum nongaussianity of the components. One robust criterion for nongaussianity testing is negentropy that can be expressed as:

$$J(y) = H(y_{gauss}) - H(y) \quad (4)$$

In which H is entropy function and $J(y)$ represents the entropy difference between random value y and Gaussian variable y_{gauss} , which has the same covariance matrix as y .

$J(y)$ will be zero if and only if y is Gaussian, otherwise it has positive non-zero value. Using a robust estimation of negentropy and ascent gradient method, we can rewrite \mathbf{W} matrix rows as following:

$$\mathbf{w}^+ = E\{\mathbf{x}g(\mathbf{w}^T \mathbf{x})\} - E\{g'(\mathbf{w}^T \mathbf{x})\} \mathbf{w}^- \quad (5)$$

where \mathbf{x} is the observed vector and \mathbf{w}^- and \mathbf{w}^+ represent the \mathbf{W} matrix row before update and after update respectively. Here $g(u) = \tanh(au)$ and a is selected so that $1 < a < 2$. In order to increase the convergence speed, data is whitened in the first step. Hence, the searching space is limited to the space perpendicular to the previous vectors (rows).

2.1.1. Parametric estimation of the probabilistic density function of the independent components

Probability density of independent components is defined as a generalized exponential function as follows [7]:

$$p(x|\mu, \sigma, \beta) = \frac{\omega(\beta)}{\sigma} \exp\left[-c(\beta) \left|\frac{x - \mu}{\sigma}\right|^{2/1+\beta}\right]$$

$$c(\beta) = \frac{\left[\frac{\Gamma[\frac{1}{2}(1+\beta)]}{\Gamma[\frac{1}{2}(1+\beta)]}\right]^{1/\beta}}{\Gamma[\frac{1}{2}(1+\beta)]} \quad (6)$$

$$\omega(\beta) = \frac{\Gamma[\frac{1}{2}(1+\beta)]^{1/2}}{(1+\beta)\Gamma[\frac{1}{2}(1+\beta)]^{3/2}} \quad \sigma > 0$$

where μ is the mean parameter, σ represents standard deviation and Γ is Gamma function. β_i is the parameter that controls the deviation of probability density function from normal distribution. The higher β gets ($\beta \gg 1$), the more the super-Gaussian of the independent components PDF is. Assuming s_i components with zero mean ($\mu = 0$) and unit variance ($\sigma^2 = 1$), we can estimate β by Maximum A Posterior (MAP) method as follows:

$$\beta_i = \max_{\beta_i} p(\beta_i | s_i) \propto \max_{\beta_i} p(s_i | \beta_i) p(\beta_i) \quad (7)$$

Assuming $p(\beta_i)$ is uniform for $\beta_i > 1$ and making the derivative with respect to β_i equal to zero, we can estimate β_i by means of iteration methods.

2.2. Shrinkage function extraction

SCS uses Maximum A Posterior for estimating the non-Gaussian variable contaminated with Gaussian noise.

Suppose s is the random non-Gaussian variable and v is the Gaussian noise with zero mean and unit variance and y is the observed variable, so that:

$$y = s + v \quad (8)$$

We want to extract s from the observed vector y . If the probability density function of s is shown by p , then we can estimate s by:

$$\hat{s} = \operatorname{argmin}_u \left[\frac{1}{2\sigma^2} (y - u)^2 + f(u) \right] \quad (9)$$

where $f = -\log p$ and σ^2 is the noise variance. Minimizing the left side of (9) and making its derivative with respect to u equal to zero is equivalent to the following equation:

$$\frac{1}{2\sigma^2} (\hat{s} - y) + f'(\hat{s}) = 0 \quad (10)$$

For the Laplace function in the form of $p(\hat{s}) = \frac{\lambda}{2} \exp(-\lambda |\hat{s}|)$, we have $f'(\hat{s}) = \operatorname{sign}(\hat{s})$. After

substitution of $f'(s)$ in (10), we find $\hat{s} = |y| - \lambda\sigma^2$ for $|y| > \lambda\sigma^2$ and the equation has no answer for $|y| < \lambda\sigma^2$. Assuming that the clean signal components are sparse, we can conclude that, for $|y| < \lambda\sigma^2$ the observed component y is only resulted from noise and therefore $\hat{s} = 0$. In general, nonlinear relationship between \hat{s} and y components can be stated:

$$\hat{s} = g(y) = \operatorname{sign}(y) \times \max(0, |y| - \lambda\sigma^2) \quad (11)$$

where g represents the shrinkage function. In ICA space, the probability density of the clean speech, $p(s)$, can be obtained as explained in section 2-1-1. Assuming generalized exponential distribution for s , it is unlikely to find \hat{s} with respect to y from (10) in the closed form. The shrinkage function of a high sparse variable ($\beta \gg 1$) can be approximated as [1]:

$$\hat{s} = \operatorname{sign}(y) \times \max\left(0, \frac{|y| - bd}{2} + \sqrt{\left(\frac{|y| + bd}{2}\right)^2 - 4\sigma^2(a+3)}\right) \quad (12)$$

where d is the standard deviation of the independent components and σ^2 is the noise components variance in ICA space and $a = 1/d/(E^2\{s|s|^2 - 1\})$, $b = \sqrt{a(a+1)}/2$. Two examples of shrinkage function are illustrated in Fig.1. Since independent components extracted from speech have high sparsity, we can use the shrinkage function stated in (12).

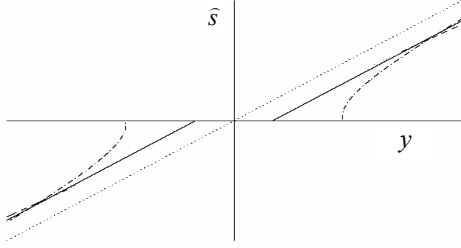


Figure 1: Two example of shrinkage function. Solid line: the shrinkage function of Laplace distribution (11), Dotted line: the shrinkage function of supergaussian distribution (12).

2.3. The Proposed Method

In previous works, SCS is used alone for speech enhancement. It seems that shrinkage threshold reduction can reduce the approximations used in solving (10) and therefore may result in improvement in denoising process. Shrinkage threshold is directly related to the noise energy. This relationship is apparent in the shrinkage function of Laplace distribution, i.e. (11). Shrinkage threshold reduction can be carried out by applying an appropriate preprocessing. One preprocessing method that can keep the Gaussianity of the noise is a linear transform and optimal linear transform that causes maximum noise reduction can be a Wiener filter. Wiener filter applied in ICA space can be easily found [8]:

$$\mathbf{F} = \mathbf{I}(\mathbf{I} - \mathbf{W}\mathbf{R}_n\mathbf{W}^T)^{-1} \quad (13)$$

where \mathbf{I} is the unit matrix and is equivalent to the covariance matrix of independent components. Note that independent components attained from ICA model have zero mean and unit variance. Besides, \mathbf{W} is the unmixing matrix of ICA model and \mathbf{R}_n is the noise covariance matrix in the time domain. Additive noise \mathbf{v} can be found by applying \mathbf{F} function on ICA space components.

$$\begin{aligned} \mathbf{F}(\hat{\mathbf{s}} + \mathbf{v}) - \hat{\mathbf{s}} = \hat{\mathbf{v}} &\rightarrow (\mathbf{F} - \mathbf{I})\hat{\mathbf{s}} + \mathbf{F}\mathbf{v} = \hat{\mathbf{v}} \\ \mathbf{R}_{\hat{\mathbf{v}}} = (\mathbf{F} - \mathbf{I})(\mathbf{F} - \mathbf{I})^T + \mathbf{F}\mathbf{R}_v\mathbf{F}^T \end{aligned} \quad (14)$$

\mathbf{v} has Gaussian distribution and according to the central limit theorem, and employing only linear function, $\hat{\mathbf{v}}$ can also be assumed Gaussian. The σ in (12) can be substituted with diagonal components of the covariance matrix $\mathbf{R}_{\hat{\mathbf{v}}}$ (noise variance) and the resultant shrinkage function can be applied to each noisy component.

3. Experimental Results

In the experimental setup, we first train the ICA model with 22 files extracted from TIDIGIT database. For this purpose, we divide the speech files to the frames of 40 samples with one sample frame shift (sampling frequency is 8 kHz). After applying hamming window on the frames and eliminating the mean value, ICA training algorithm using negentropy criteria was performed. \mathbf{W} matrix has been estimated in the training process and independent components were calculated by $\mathbf{s} = \mathbf{W}\mathbf{x}$. β corresponds to the probability

density function of independent components and a , d and b pertaining to the shrinkage function have been also estimated. The estimated β was in the range of 7-12 that verified the supergaussianity of the independent components. Finally, we found 40 shrinkage functions for 40 independent components. In order to test the algorithm, 30 files have been selected randomly and specific noises were added manually from NATO RSG-10 noise data. We used two parameters for evaluating the enhancement capability of our approach which are global SNR and segmental SNR. Table 1 and Table 2 include the obtained values of global and segmental SNRs respectively after applying Wiener filter, Shrinkage function and jointly using them aimed to improving the speech signal quality. Segmental SNR results represent the difference between segmental SNR of the input noisy speech and enhanced speech signal. The results show that our proposed method (with the presumption of having only Gaussian noise) not only improved the SNR of the speech signal in the presence of white noise but also could be efficient for the speech signal contaminated with the colored noise. Except for the Volvo noise (where the Wiener filter has reduced the noise level considerably) it can be seen that our proposed method has made a significant improvement in all SNRs.

4. Conclusion

In this paper we proposed a new method by jointly using Sparse Coding Shrinkage and Wiener filtering. In general, each speech enhancement algorithm independently reduces the effect of a specific noise. Thus, combinational algorithms can be more efficient in various noise environments. For example, in the presence of Volvo noise, Table 1 shows that the performance of Wiener filter in reducing the noise level is considerable, in comparison with the SCS effect. Therefore, jointly use of these methods may lead to better results in the presence of various types of noises. On the other hand, Wiener filter improves the SCS method performance by reducing the shrinkage threshold of the shrinkage function. The performance of the proposed method should be further studied before its application in speech recognition systems.

5. Acknowledgements

This work was in part supported by Iran Telecommunications Research Center (ITRC).

6. References

- [1] I. Potamitis, N. Fakotakis, G. Kokkinakis, "Speech enhancement using the sparse code shrinkage technique," in *Proc. ICASSP*, vol.1, pp.621-624, May 2001.
- [2] A. Hyvarinen, P. Hoyer, E. Oja, "Sparse code shrinkage for image denoising," in *Proc. IJCNN*, vol.2, pp.859-864, May 1998.
- [3] A. Hyvarinen, P. Hoyer, E. Oja, "Sparse Code Shrinkage: Denoising of Nongaussian Data by Maximum Likelihood Estimation," *Neural Computation*, Vol. 11, No. 7, pp. 1739-1768, Oct 1999.
- [4] J. Liu, C. Zhao, X. Zou, Wei Zhang, "An Approach of Speech Enhancement by Sparse Code Shrinkage," in *Proc. ICNN & B*, vol.3, pp.1952-1956, Oct 2005.

Table 1: Global SNRs of enhanced signals in different enhancement algorithms.

noise \ SNR(dB)		20	15	10	5	0	-5	Ave.
Babble	SCS	20.92	16.39	11.98	7.74	3.73	-.26	10.08
	Wiener	20.2	15.47	10.9	6.7	3.17	.78	9.54
	SCS+Wiener	20.9	16.4	12	7.8	4.1	1.59	10.47
Factory	SCS	21	16.59	12.24	7.95	3.8	-.3	10.21
	Wiener	20.68	16.2	11.8	7.6	4	1.3	10.26
	SCS+Wiener	21.2	17.1	13.1	8.9	4.9	2	11.20
F16	SCS	21.22	16.79	12.42	8.11	3.99	-.8	10.29
	Wiener	20.2	16.5	12.2	8.2	4.53	1.7	10.56
	SCS+Wiener	21.5	17.5	13.6	9.6	5.6	2.49	11.72
White	SCS	22.37	18	13.55	9	4.65	.2	11.30
	Wiener	20.7	16.14	12	8.25	5	2.38	10.75
	SCS+Wiener	22.4	18	13.8	9.9	6.2	3.2	12.25
Volvo	SCS	21.28	17.22	12.95	8.68	4.46	.16	10.79
	Wiener	25.8	24.7	23.4	20.95	17.8	16.4	21.51
	SCS+Wiener	24	22.9	21.4	18.8	15.8	13	19.42
Destroyer Engine	SCS	22.51	18.24	13.8	9.3	4.86	.4	11.52
	Wiener	20.32	15.9	11.94	8.36	5.03	2.2	10.63
	SCS+Wiener	22	18	13.9	10	6.3	2.2	12.07
Ave.	SCS	21.55	17.21	12.82	8.46	4.25	-0.10	10.70
	Wiener	21.32	17.49	13.71	10.01	6.59	4.13	12.21
	SCS+Wiener	22.00	18.32	14.63	9.24	7.15	4.08	12.85

Table 2: Segmental SNR improvement for different enhancement algorithms (the results are differential SNR between input signal and enhanced signal).

noise \ SNR(dB)		20	15	10	5	0	-5	Ave.
Babble	SCS	1.19	2.10	3.5	4.8	5.6	5.9	3.85
	Wiener	.17	.46	.99	1.93	3.77	7.1	2.40
	SCS+Wiener	1.18	2.32	4.46	7.46	10.63	14.43	6.75
Factory	SCS	1.18	2.14	3.32	4.37	5.1	5.6	3.62
	Wiener	.56	1	1.8	2.8	4.56	7.6	3.05
	SCS+Wiener	1.42	2.9	5.2	8.1	11	14.9	7.25
F16	SCS	1.4	2.27	3.33	4.33	5.1	5.6	3.67
	Wiener	.7	1.34	2.19	3.31	5	7.9	3.41
	SCS+Wiener	1.77	3.3	5.6	8.4	11.4	15	7.58
White	SCS	2.54	3.33	4.01	4.58	5.06	5.45	4.16
	Wiener	.8	1.4	2.43	3.97	6.12	9	3.95
	SCS+Wiener	3.12	4.47	6.25	8.65	11.79	15.8	8.35
Volvo	SCS	.16	1.13	1.82	2.61	3.6	4.56	2.31
	Wiener	5	8.2	11	13	15	17	11.53
	SCS+Wiener	3.5	7	10.3	13.3	16.5	19.8	11.73
Destroyer Engine	SCS	2.97	3.9	4.56	5.05	5.43	5.74	4.61
	Wiener	.4	1.13	2.34	4	6	8.8	3.78
	SCS+Wiener	3.12	4.5	6.3	8.9	12	15.8	8.44
Ave.	SCS	1.57	2.48	3.42	4.29	4.98	5.48	3.70
	Wiener	1.27	2.26	3.46	4.84	6.74	9.57	4.69
	SCS+Wiener	2.35	4.08	6.35	9.14	12.22	15.96	8.35

[5] P. Vary, "Noise suppression by spectral magnitude estimation," *Signal Processing*, Vol. 8, pp. 387-00, 1985.
 [6] A. Hyvarinen, J. Karhunen, E. Oja, *Independent Component Analysis*, John Wiley & Sons, New Yourk etc. 2001.

[7] T.-W. Lee, M.-S. Lewicki" *The Generalized Gaussian Mixture Model Using ICA*," in *Proc. Int Workshop. on Independent Component Analysis*, pp.239-244, 2000.
 [8] T.-W. Lee, M.-S. Lewicki" *The Generalized Gaussian Mixture Model Using ICA*," in *Proc. Int Workshop. on Independent Component Analysis*, pp.239-244, 2000.