AN ACOUSTIC ECHO CANCELLER COMBINED WITH BLIND SOURCE SEPARATION

Duong Quang K. Ngoc [*], Park Chul[**], and Seung-Hyon Nam[***]

Departmemt of Electronic Engineering, Paichai University

[*]dqkngoc@pcu.ac.kr, [**]pc8105@pcu.ac.kr[**], [***]shnam@pcu.ac.kr

ABSTRACT

Acoustic echo cancellation (AEC) aims to suppress the echo picked up by the microphone in a hands-free communication system. But the suppression is significantly reduced during double-talk period in which both sides talk simultaneously since the adaptive filter of AEC is frozen. In this paper, we propose to apply blind source seperation (BSS) in front of the AEC in a car environment. We developed a real-time block online BSS for this purpose. It is demostrated that the combination of AEC and BSS can enhance the overall speech enhancement performance. Simulation results using real audio recording in a car confirm the expectation.

I. INTRODUCTION AND MOTIVATION

Acoustic echo cancellation (AEC) is an essential part of a hands-free communication system that eliminates acoustic echo picked up by a microphone. The AEC can be employed in a car where a driver can make a hands-free communication with far-end side while driving. We assume that voice from the far-end side is played back by many speakers (normally more than four) installed in a car. In a car, there exist strong consistent background noise and the loudspeaker-enclosure microphone (LEM) environment is constantly changing. All these make AEC operation difficult. Furthermore, during double-talk period, the adaptive filter of AEC is not adapted to maintain stability of the AEC and echo may not be well-suppressed.

In this paper, we propose to use blind signal separation to separate interfering signals including the voice from the far-end side to improve the performance of the AEC. BSS is known as a powerful tool for enhancing noisy speech in many real world environments. In a car, hands-free communication may suffer from interfering signals such as background noise and voices of other passengers which cannot be removed in the AEC. BSS may be helpful in reducing those interferences.

The conceptual block diagram of the AEC combined with BSS is shown in Fig. 1. BSS works on two microphone signals and separate input signals into local driver's voice and the interfering signals. The interfering signal may be far-end echo or local interfering signals. The output channel that contains the local driver's voice is then fed into the AEC. The residual echo remains in the separated output is then removed further in the AEC.

A basic acoustic echo canceller consists of the blocks of band-pass filter, de-correlation filter, voice activity detection, double talk detection, normalized least mean square (NLMS) adaptive filter, non-linear processor, and optional auto gain control. A modified open-loop correlation method is proposed for reliable double talk detection [2]. For BSS, the multichannel blind deconvolution (MBD) algorithm with spectrum normalization is used. This algorithm is known to provide fast stable convergence and improved separation [1]. For continuous real-time separation operation, the BSS algorithm is implemented in a block-online fashion.

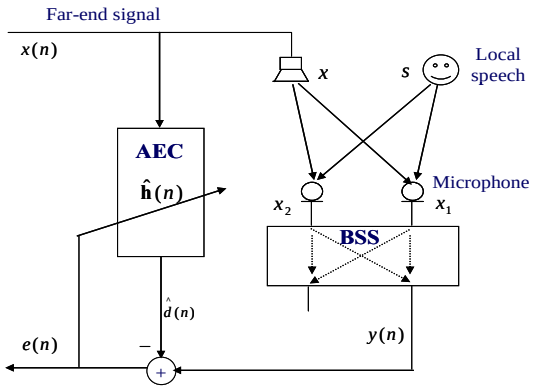

Fig. 1. Block diagram of AEC combines with BSS.

II. FREQUENCY-DOMAIN BLOCK AEC

The conceptual block diagram illustrating the operation of the AEC algorithm is shown in Fig. 2. Descriptions on major blocks are given below.

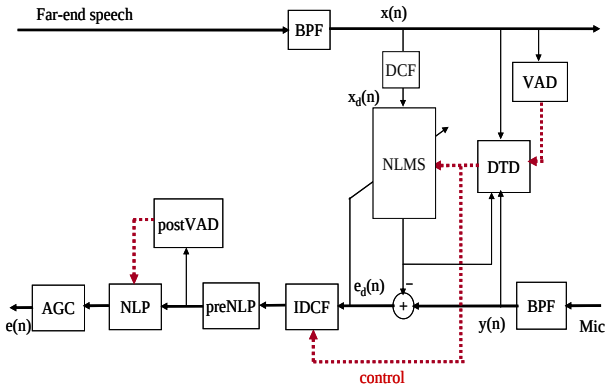

Fig. 2. Block diagram of a basic AEC implementation.

2.1. Decorrelation and inverse decorrelation filter

It is known that a fixed first order decorrelation filter provides remarkable result [3]. The first order filter can be a simple high-pass filter since speech signals have low-pass characteristics. The formula to decorrelate the far-end and the microphone signals for NLMS tap-weight update are:

$$x_d(n) = x(n) - ax(n-1) \quad (1)$$

$$y_d(n) = y(n) - ay(n-1) \quad (2)$$

where $a = 0.85$.

Two decorrelation filters are necessary to decorrelate two signals that are fed into the NLMS adaptive filter and the inverse decorrelation filter is necessary to recover true error signal as $e(n) = e_d(n) + ae_d(n-1)$. These blocks are shown in Fig. 2.

2.2. Voice activity detection

A short-term magnitude estimation of each block of input signal is computed as [3]

$$\overline{|x(n)|} = (1-\gamma(n))|x(n)| + \gamma(n)\overline{|x(n-1)|} \quad (3)$$

where

$$\gamma(n) = \begin{cases} \gamma_r, & \text{if } |x(n)| > \overline{|x(n-1)|} \\ \gamma_f, & \text{else.} \end{cases}$$

For 8 kHz sampling rate, we choose $\gamma_r = 0.95$ and $\gamma_f = 0.97$. Voice activity of a far-end signal is detected if the short-term magnitude exceeds a predefined noise threshold of 35 dB.

2.3. Double talk detection

Double talk detection is a typical problem that is deeply concerned in AEC implementation to detect the situation in which both sides talk simultaneously. During the double talk period, the residual error increases due to local speech so that the AEC stability bound decreases and the algorithm may start to diverge. This situation must be prevented and the adaptive filter coefficients must be frozen during the double talk. To detect the double talk, correlation-based methods use degree of similarity between microphone signal $y(n)$ and the output signal $\hat{d}(n)$ of adaptive filter or the loudspeaker signal $x(n)$ in term of correlation [2,3].

2.4. Frequency-domain block NLMS adaptive filter

The frequency domain block NLMS filter provides a possible solution to the computational structure and complexity problem. The algorithm consists of the following main steps [6]. The most recently acquired block of $N/2$ input samples and the previous block make up the new input block $\mathbf{X}_n$ and this new block was transformed into frequency domain by the discrete Fourier transform: $\mathbf{X}_n = \mathbf{F}\{\mathbf{x}_n\}$. The most recent L frequency domain input blocks are retained $\mathbf{X}_n = [\mathbf{X}_n \mathbf{X}_{n-1} \ldots \mathbf{X}_{n-L+1}]$.

The adaptive filter uses low-latency structure where the filter response is evenly divided into non-overlapping segments and filter output can be calculated in time-domain as

$$\hat{y}(n) = \sum_{j=0}^{L*N/2-1} h(j)x(n-j) = \sum_{i=0}^{L-1}\sum_{k=0}^{N/2-1} h_i(k)x_j(n-k) \quad (4)$$

This corresponds to calculation in frequency domain is given by

$$\hat{\mathbf{Y}}_n(k) = \sum_{i=0}^{L-1} \mathbf{X}_{n-i}(k)\mathbf{H}_i(k) \quad for \quad k = 0,...,N/2 \quad (5)$$

where L is number of segments, $\mathbf{H}_i$ is a segments of NLMS filter response in frequency domain and has been properly constrained by the rule of overlap-save method such that the underlying convolution is linear [7]. To exactly implement the linear convolution in frequency domain by overlap-save method, the filter response of each segment $\mathbf{H}_i$ must be constrained by transforming back to time domain and only the first half elements is retained, the rest set to zero.

III. BLOCK ONLINE BSS

In this paper, we used the MBD algorithm with spectral normalization in the frequency domain [1,7]. This algorithm provides fast and stable convergence while reducing the whitening problem of the separated output. It can be implemented in the frequency domain using FFT for reduced computation.

The BSS algorithm uses single-sided unmixing filters and its update rule can be described by the following simplified conceptual expression in the frequency domain.

$$\Delta\mathbf{W}^f(b) = \{\overline{\mathbf{I}} - \mathbf{\Lambda}_{\mathbf{y}}^{-1}(b)\mathbf{y}^f(b)(\mathbf{u}^f(b))^H \mathbf{\Lambda}_{\mathbf{u}}^{-1}(b)\}\mathbf{W}^f(b) \quad (6)$$

Here, b is the block index, f denotes the frequency quanity, $\mathbf{\Lambda}_{\mathbf{y}}(b)$ and $\mathbf{\Lambda}_{\mathbf{u}}(b)$ are diagonal matrices that have $\sqrt{\mathbf{P}_{y_i}(b)}$ and $\sqrt{\mathbf{P}_{u_i}(b)}$ as diagonal elements, respectively. Further, $\overline{\mathbf{I}}$ denotes the identity matrix extended in the frequency axis. The spectral power is computed by the following update rule.

$$\begin{aligned} \mathbf{P}_{y_i}(b) &= (1-\gamma)\mathbf{P}_{y_i}(b-1) + \gamma|\mathbf{y}_i^f(b)|^2 \\ \mathbf{P}_{u_i}(b) &= (1-\gamma)\mathbf{P}_{u_i}(b-1) + \gamma|\mathbf{u}_i^f(b)|^2 \end{aligned} \quad (7)$$

It should be noted that (6) is a just simplified conceptual expression. The real computation includes linear correlations and convolutions via circular convolutions and correlations in the frequency domain. Thus, farward/inverse DFT and proper aliasing eliminations are required. The algorithm can be implemented either in the overlap-save or overlap-add method.

As described in [9], the fast stable convergence of this BSS algorithm comes from the spectral normalization that equalizes the spectral tilt. The algorithm can accommodate a large step size without causing instability so that it can separate the mixed signals successfully with few number iterations of a given short data segement.

The real-time block-online implementation is based on the existing methods described in [10,11]. That is, input mixed signals are divided into online and offline blocks. One online block is made of a number of offline blocks (4 blocks in this case). These offline blocks are processed iteratively using the update rule (6). At each offline block b and at each iteration *j*, gradient of the offline update $\Delta\mathbf{W}^j(b)$ is computed. These offline updates are then averaged over *K* offline blocks at each iteration as

$$\Delta\mathbf{W}^j(b') = \frac{1}{K}\sum_{k=1}^{K}\Delta\mathbf{W}_k^j(b) \quad (8)$$

where b' denotes the online block index. After the iterative learning of b' online block, the averaged offline gradients are then added together to get online gradient.

$$\Delta \mathbf{W}(b') = \sum_{j=1}^{J} \Delta \mathbf{W}_k^j(b') \tag{9}$$

This online gradient is then used to perform online update as follows.

$$\mathbf{W}(b') = \mathbf{W}(b'-1) + \mu_{on} \Delta \mathbf{W}(b') \tag{10}$$

The updated unmixing filter is then used as an initial value for the next online update. Main advantage of the block-online algorithm is tracking capability of the changing environment such as moving speakers.

IV. SIMULATIONS

4.1 Block-online BSS algorithm

Although intensive investigation of the characteristics of the block-online BSS algorithm is now underway, we investigate the performance of the block-online BSS algorithm in a room environment with reverberation time T_{60}=340ms. Speakers are 70cm apart from two microphones, speaker 1 is placed at 105° and speaker 2 at 70°. After 10sec, speaker 2 was moved from 70° to 40°. Block size is set to 1024, filter length 512, frame rate 2 (50% overlap), number of offline blocks in an online block is *K*=4, iteration time *J*=10, $\mu = 0.02$, $\mu_{\text{on}} = 0.12$. To see the performance of the BSS algorithm, we evaluated the SIR (signal-to-interference ratio) gain as shown in Fig. 3. SIR values are fluctuating due to variations of signal power by frames. It converges after 2 online blocks. After speaker 2 moved at 10sec, the SIR of moved speaker has been dropped while that of fixed speaker increased. It takes about 2 sec to recover. It turns out that the batch algorithm provides SIR of 8.4 and 4.5 dB for each channel, respectively. Thus the block–online algorithm is shown to be effective for moving speaker.

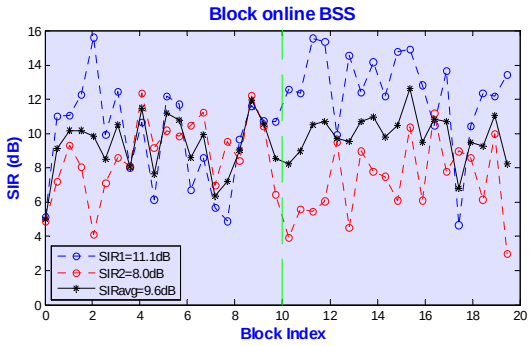

Fig. 3. SIR of the block-online BSS algorithm for a moving speaker.

4.2 AEC combined with block-online BSS

Simulation Setup: Microphone placement in a car and the processing steps are shown in Fig. 4. This simulation setup is very tough because the far-end voice is played back from the six speakers of the car. In a car, the reverberation time is approximately 50msec and sampling rate is 8kHz. Since background noise level in the car (including engine noise and wind noise) is noticeably high and that may degrade both BSS and AEC performance, Wiener filtering is necessary to eliminate the noise. In our simulation, AEC is setup with NLMS adaptive filter length 256, block size $B = N/2 = 64$, step-size $\mu = 0.05$. Block online BSS is set up with block size $L = 1024$, frame size is twice times of block size and 10 learning iterations.

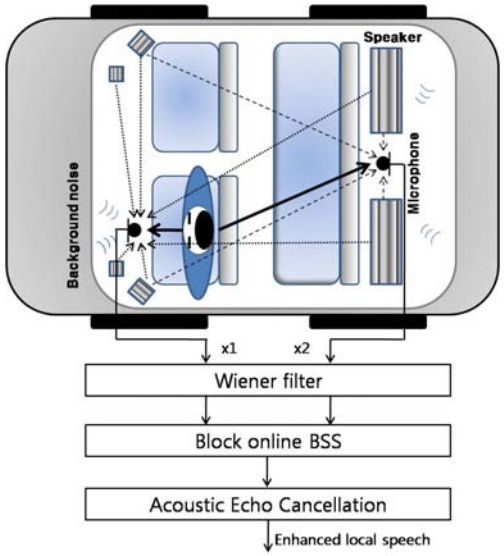

Fig.4. Simulation setup and speech enhancement process.

Simulation Results and Discussions: Figure 5 shows the recorded signal by two microphones and the resulting echo remaining after BSS and AEC. After BSS, echo is remarkably suppressed while retaining the local speech. The residual echo after BSS is suppressed further in AEC so that only local speech is transmitted to the far-end side.

To evaluate the performance of AEC and AEC combines with BSS, the error return loss enhancement (ERLE) is calculated which is defined at time index n as

$$ERLE = \frac{\overline{x_{echo}^2}}{\overline{e_{echo}^2}}$$

where $\overline{x_{echo}^2}$ and $\overline{e_{echo}^2}$ are the power of the echo signal at microphone and the remain echo at the output after AEC, respectively, averaged over a block.

The combined structure provides higher ERLE of approximately 6 dB in both single talk and double talk period when compare to AEC only as shown in Fig. 6. Notice that this improvement comes from the SIR gain of BSS. However, it is not sure at this moment if the SIR gain of BSS is directly reflected into the ERLE of AEC or not. Since the step size and regularization parameter of AEC govern the convergence speed and the ERLE, those parameters should be chosen carefully after the parameter setup of BSS.

In addition to the above mentioned advantages, we may use the simpler energy based method than the correlation based method for double-talk detection without losing any accuracy. In addition, we may exploit the information from the other separated output channel in the AEC operation.

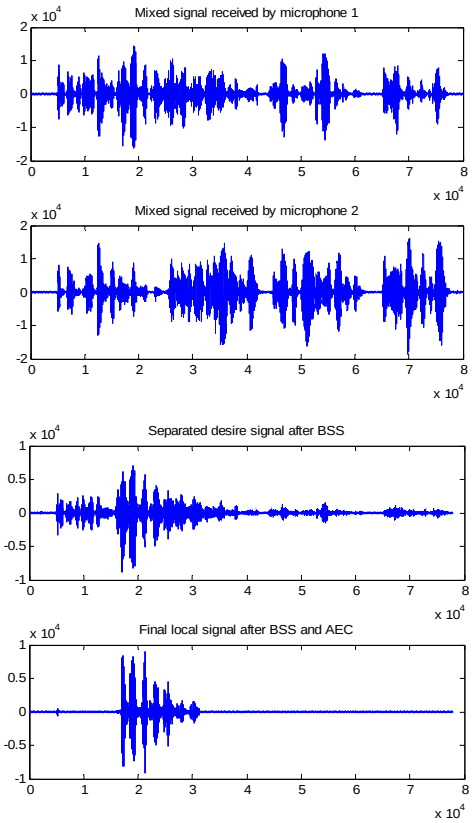

Fig.5. Microphone speech signals and echo cancellation signal after BSS and AEC.

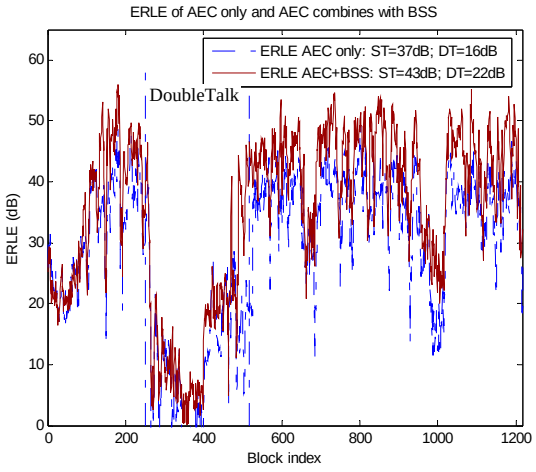

Fig.6. Comparision of ERLE: AEC only (dash-dot) and BSS+AEC (solid).

V. CONCLUSIONS

In this paper, we have proposed a new method to enhance acoustic echo cancellation in a car environment. The proposed method is to combine BSS with AEC. A basic AEC and block online BSS implementation used have been described briefly. Draft simulation results in a car reveals that putting BSS before AEC can enhance echo cancellation performance during both single talk and double talk period in a hands-free communication system. To get better understanding of the AEC system combined with BSS, we need further investigation with different setup for microphones and speakers.

ACKNOWLEDGEMENT

This work was supported by the Korea Research Foundation Grant funded by Korean Government (MOEHRD) (KRF-2005-041-D00603) and Basic Research Promotion Fund (KRF-2006-211-D00114).

REFERENCES

[1] S. H. Nam and S. Beack, "A frequency-domain nomalized multichannel deconvolutive algorithm for acoustical signals," *Proc. ICA (LNCS 3195)*, Springer, pp. 522-529, 2004.

[2] D.Q.K. Ngoc and S.H Nam, "Implementation of a basic echo canceller", *Proc. of the 19th Joint Signal Processing Conf.*, Sept., 2006.

[3] E. Hansler, G. Schmidt, *Acoustic Echo and Noise Control-A Practical Approach*, Wiley, 2004.

[4] S. Haykin, *Adaptive filter theory*, 4th Ed., Prentice-Hall, 2002.

[5] A. Mader, H. Puder, G.U. Schmidt, "Step-size control for acoustic echo cancellation filters – an overview", *Signal Processing*, vol.80, no.9, pp.1697-1719, 2000.

[6] Microchip, *Acoustic Echo Cancellation Library*, DS7013A.

[7] S. Beack, S.H. Nam, and M. Hahn, "A new speech enhancement algorithm for car environment noise cancellation with MBD and Kalman filter", *IEICE trans. Fundamental*, vol.E88-A, No.3, March 2005.

[8] T. Liu, S. Gazor, "De-correlated algorithms for faster adaptation", *Proc. of IEEE Int. Conf. on Signal Processing*, 2002.

[9] S. H. Nam and I. Jee, "A new formulation of multichannel blind deconvolution: Its properties and modification for speech separation," *The Jour. of Acoustical Society of Korea*, vol.25, no. 4E, pp.148-153, 2006.

[10] R. Aichner, H. Buchner, F. Yan, and W. Kellermann, "Real-time convolutive blind source separation based on a broadband approach," *Proc. ICA (LNCS 3195)*, Springer, 2004, pp. 840-548.

[11] R. Mukai, H. Sawada, S. Araki, and S. Makino, "Real-time blind source separation for moving speech signals," in *Speech Enhancement*, *Eds*. J. Benesty, S. Makino, and J. Chen, Springer, 2005.