2008 IEEE/RSJ International Conference on Intelligent Robots and Systems
Acropolis Convention Center
Nice, France, Sept, 22-26, 2008

Real-Time Implementation of Blind Spatial Subtraction Array for Hands-Free Robot Spoken Dialogue System

Yu Takahashi, Hiroshi Saruwatari, and Kiyohiro Shikano

Abstract— **In this paper, we construct a hands-free robot spoken dialogue system based on the real-time blind spatial subtraction array (BSSA) and evaluate the system. BSSA is the blind source extraction method, and the source extraction in BSSA is carried out by subtracting the power spectrum of the estimated noise signal by the independent component analysis from the power spectrum of the target speech partly enhanced signal. Although BSSA can reduce noise signal efficiently, ICA consumes huge amount of computational costs. Thus it is difficult to run BSSA in real-time. In this paper, we newly propose a real-time architecture of BSSA and construct a hands-free robot spoken dialogue system based on the real-time BSSA. In the hands-free robot spoken dialogue system with the real-time BSSA, 6% improvement of the speech recognition result can be seen compared with the conventional speech enhancement methods.**

I. Introduction

A hands-free speech recognition system is essential for realizing an intuitive, unconstrained, and stress-free human-machine interface, especially in human-robot speech interaction [1]–[3]. In this system, however, it is difficult to achieve a high recognition accuracy because noise and the reverberation always deteriorate a target speech quality.

One approach to address the problem is to separate the observed signals into each source signal by blind source separation (BSS) technique. BSS is the approach to estimate the original sources using only information of the observed signal in each microphone. Basically, BSS is classified as an unsupervised filtering technique, and does not require any supervisions on directions-of-arrival (DOAs) and target-speech pause where only noise exists. Recently, various methods of BSS based on independent component analysis (ICA) [4] have been presented on acoustic-sound separation [5]–[8]. Indeed the conventional ICA could work especially in speech-speech (or point sources) mixing, but such a mixing condition is very rare and not realistic; real noises are often widespread sources. In such a sound mixing condition, we have found that ICA is proficient in noise estimation rather than in target speech estimation [9]. Based on the above-mentioned fact, we have proposed a novel blind source extraction method, i.e., blind spatial subtraction array (BSSA) which utilize ICA as noise estimator [9]. In BSSA, source extraction is achieved by subtracting the power spectrum of the estimated noise via ICA from the power spectrum of the noisy observations.

This work was partly supported the NEDO project for strategic development of advance robotics elemental technologies in Japan.

The authors are with the Graduate School of Information Science, Nara Institute of Science and Technology, Nara 630-0192, Japan (e-mail: {yuu-t, sawatari, shikano}@is.naist.jp). Tel: +81-743-72-5287, Fax: +81-743-72-5289.

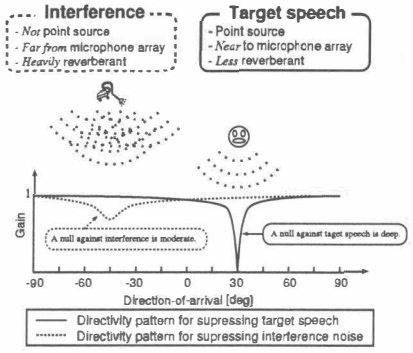

Fig. 1. Directivity pattern which is shaped by ICA.

To work in real-time is one of the indispensable factor for a hands-free speech recognition system. Indeed BSSA can reduce noises efficiently, BSSA is difficult to work in real-time because ICA part of BSSA consumes huge amount of computational complexities. Thus, it is required to develop a real-time architecture of BSSA.

In this paper, we newly propose the real-time architecture of BSSA and implement the real-time BSSA. Moreover, we introduce the implemented real-time BSSA into the spoken-oriented guidance system "Kitarobo" which has already been installed at an actual railway station, and construct a hands-free spoken dialogue system. Although many real-time robot audition systems have been proposed [3], the behavior and performance are not explicitly analyzed under heavy widespread noise condition, e.g., an actual railway-station, as far as we know. Finally, we evaluate the constructed hands-free spoken dialogue system with the real-time BSSA based on the speech recognition test, and 6% improvement of the speech recognition result can be revealed compared with the conventional speech enhancement methods.

II. Blind Spatial Subtraction Array

A. Motivation

Generally speaking, the conventional ICA could work particularly in speech-speech mixing, i.e., all sound sources can be regarded as point sources, but such a mixing condition is very rare and unrealistic; real noises are often widespread. Thus, the following scenario and problem are likely to arise (see Fig. 1).

- The target sound is user's speech, which can be approximately regarded as a *point source*. In addition, the user locates themselves relatively *close to the microphone array* (e.g., 1 m apart), and consequently the

978-1-4244-2058-2/08/$25.00 ©2008 IEEE.

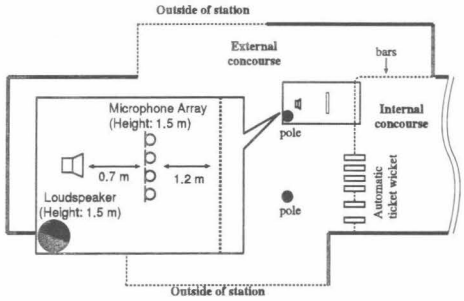

Fig. 2. Layout of an actual railway-station in our preliminary experiment.

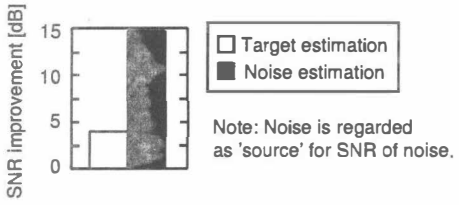

Fig. 3. Separation results in our preliminary experiment.

accompanying reflection and reverberation components are small.

- As for the noise, we are often confronted with interference sounds which are *not point sources* but widespread sources. Also the noise is usually far from the array and heavily reverberant.

From the above-mentioned scenario, it is expected that the conventional ICA can suppress the user's speech signal to pick up the noise source, but ICA is very weak in picking up target speech itself via suppression of the far-located widely-spread noise. This is due to the fact that ICA with the small number of sensors and filter taps often provides only directional nulls against the undesired source signals [8].

To confirm this fact, we have conducted preliminary source separation experiment under an actual railway-station condition. Figure 2 illustrates the layout of the railway-station in this experiment, where the reverberation time is 1000 ms. We use 46 speakers (200 sentences) as the target speech. As for noise, we used an actually recorded railway-station noise. Figure 3 shows the result for the average SNR improvement of all the target speakers, and from this result, we can confirm that ICA is proficient in noise estimation rather than in target speech estimation. This result gives us an unfortunate conclusion that ICA is *not* proficient in speech enhancement in such a diffuse noise environment. However, this also implies that we can still use ICA as an accurate noise estimator even under reverberant conditions.

Based on the above-mentioned fact, we have proposed BSSA that utilizes ICA as a noise estimator [9]. In BSSA, source extraction is achieved by subtracting the power spectrum of the estimated noise via ICA from the power spectrum of the target speech enhanced observed signal via delay-and-sum (DS). The detailed signal processing is shown below.

B. Basic Principle of BSSA [9]

The block diagram of the BSSA is shown in Fig. 4. BSSA consists of two paths; a primary path which is DS-based target speech enhancer, and a reference path which is ICA-based noise estimator. Finally, we obtain the target speech extracted signal based on spectral subtraction [10].

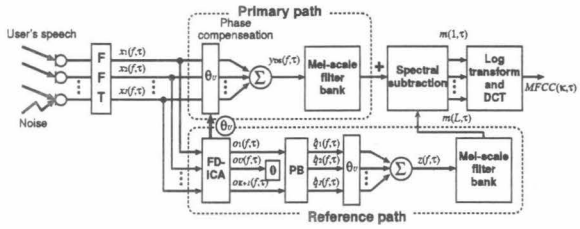

Fig. 4. The block diagram of the off-line BSSA

First, the observed signal vector in time-frequency domain is defined as

$$\boldsymbol{x}(f,\tau) = [x_1(f,\tau),\ldots,x_J(f,\tau)]^{\mathrm{T}}, \tag{1}$$

where $\boldsymbol{x}(f,\tau)$ is the observed signal vector, f is the frequency bin, τ (= 0, 1, 2, ...) is time frame index, and J is the number of microphones. In the primary path, the target speech is partly enhanced via DS; the procedure can be given as

$$y_{\mathrm{DS}}(f,\tau) = \boldsymbol{g}_{\mathrm{DS}}(f,\theta_U)^{\mathrm{T}}\boldsymbol{x}(f,\tau), \tag{2}$$

$$\boldsymbol{g}_{\mathrm{DS}}(f,\theta) = [g_1^{(\mathrm{DS})}(f,\theta),\ldots,g_J^{(\mathrm{DS})}(f,\theta)]^{\mathrm{T}}, \tag{3}$$

$$g_j^{(\mathrm{DS})}(f,\theta) = \frac{1}{J}\exp\left(-i2\pi(f/M)f_s d_j \sin\theta/c\right), \tag{4}$$

where $\boldsymbol{g}_{\mathrm{DS}}(f,\theta)$ is the coefficient vector of DS array, and θ_U is the look direction which is estimated by the unmixing matrix optimized by ICA [11]. Also, f_s is the sampling frequency and d_j ($j = 1,\cdots,J$) is the microphone position. Besides, M is the DFT size, and c is the sound velocity.

In the reference path, the ICA-based noise estimation is performed. First, we perform signal separation using the complex valued unmixing matrix $\boldsymbol{W}_{\mathrm{ICA}}(f)$, so that the output signals $\boldsymbol{o}(f,\tau) = [o_1(f,\tau),\ldots,o_K(f,\tau)]^{\mathrm{T}}$ become mutually independent; this procedure can be represented by

$$\boldsymbol{o}(f,\tau) = \boldsymbol{W}_{\mathrm{ICA}}(f)\boldsymbol{x}(f,\tau), \tag{5}$$

$$\boldsymbol{W}_{\mathrm{ICA}}(f) = \begin{bmatrix} W_{11}^{(\mathrm{ICA})}(f) & \cdots & W_{1J}^{(\mathrm{ICA})}(f) \\ \vdots & \ddots & \vdots \\ W_{K1}^{(\mathrm{ICA})}(f) & \cdots & W_{KJ}^{(\mathrm{ICA})}(f) \end{bmatrix}. \tag{6}$$

Also, the unmixing matrix is updated iteratively by

$$\boldsymbol{W}_{\mathrm{ICA}}^{[p+1]}(f) = \mu\left[\boldsymbol{I} - E[\boldsymbol{\varphi}(\boldsymbol{o}(f,\tau))\boldsymbol{o}^{\mathrm{H}}(f,\tau)]\right]\boldsymbol{W}_{\mathrm{ICA}}^{[p]}(f) + \boldsymbol{W}_{\mathrm{ICA}}^{[p]}(f), \tag{7}$$

where μ is the step size parameter, $[p]$ is used to express the value of the p-th step in the iterations, $\boldsymbol{I}$ is an identity matrix, and $E[\cdot]$ is the expectation operator. Besides, $\boldsymbol{M}^{\mathrm{H}}$ denotes hermitian transpose of matrix $\boldsymbol{M}$, and $\boldsymbol{\Phi}(\cdot)$ is the appropriate nonlinear vector function [7].

In the reference path, it is only required to estimate noise component. Thus, the target signal component $o_U(f,\tau)$ is removed from the output signal vector $\boldsymbol{o}(f,\tau)$. This processing can be designated as

$$\boldsymbol{q}(f,\tau) = [o_1(f,\tau),\ldots,o_{U-1}(f,\tau),0, o_{U+1}(f,\tau),\ldots,o_K(f,\tau)]^{\mathrm{T}}. \tag{8}$$

Next, we apply the projection back (PB) [6] method to remove the ambiguity of amplitude. This procedure can be represented as

$$\hat{\boldsymbol{q}}(f,\tau) = \boldsymbol{W}^{+}_{\mathrm{ICA}}(f)\boldsymbol{q}(f,\tau), \tag{9}$$

where $\boldsymbol{M}^{+}$ denotes Moore-Penrose pseudo inverse matrix of $\boldsymbol{M}$. Next, we obtain the estimated noise signal $z(f,\tau)$ by performing DS as follows:

$$z(f,\tau) = \boldsymbol{g}^{\mathrm{T}}_{\mathrm{DS}}(f)\hat{\boldsymbol{q}}(f,\tau). \tag{10}$$

Note that $z(f,\tau)$ is the function of the frame number τ, unlike the constant noise prototype estimated in the traditional spectral subtraction method [10]. Therefore, BSSA can deal with *nonstationary* noise.

Finally, source extraction is achieved by spectral subtraction as follows

$$y_{\mathrm{BSSA}}(f,\tau) = \begin{cases} \left\{|y_{\mathrm{DS}}(f,\tau)|^2 - \beta\cdot|z(f,\tau)|^2\right\}^{\frac{1}{2}}, \\ \quad (\text{ if } |y_{\mathrm{DS}}(f,\tau)|^2 - \beta\cdot|z(f,\tau)|^2 \geq 0\ ) \\ \gamma\cdot|y_{\mathrm{DS}}(f,\tau)| \quad (\text{otherwise}), \end{cases} \tag{11}$$

where $y_{\mathrm{BSSA}}(f,\tau)$ is the final output BSSA, β is the over-subtraction parameter, and γ is the flooring parameter. The appropriate setting, e.g., $\beta > 1$ and $\gamma \ll 1$, gives an efficient noise reduction.

III. Real-Time Implementation of BSSA

A. Overview

DS, spectral subtraction, and separation filtering in BSSA are possible to work in real-time. However, it is toilsome to optimize (update) the separation filter in real-time because the optimization of the unmixing matrix by ICA consumes amount of computational costs. Therefore, we will introduce a strategy in that the separation filter optimized by using the past time period data is applied to the current data. Figure 5 illustrates a configuration of a real-time implementation for BSSA. Signal processing in this implementation is performed via the following manner.

1) Inputted signals are converted into time-frequency domain series by using a frame-by-frame fast Fourier transform. (FFT).
2) ICA is conducted using the past 1.5-s-duration data for estimating separation filter while the current 1.5 s. The optimized separation filter is applied to the next (*not current*) 1.5 s samples. This staggered relation is due to the fact that the filter update in ICA requires substantial computational complexities and cannot provide the optimal separation filter for the current 1.5 s data.
3) Inputted data is processed in two paths. In the primary path, target speech is partly enhanced by DS. In the reference path, ICA-based noise estimation is conducted. Again, note that the separation filter for ICA is optimized by using the past time period data.
4) Finally, we obtain the target-speech-enhanced signal by subtracting the power spectrum of the estimated noise signal in the reference path from the power spectrum of the primary path's output.

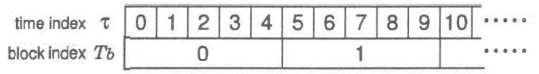

Fig. 6. Relation between time index and block index ($l_{\mathrm{sample}} = 5$ case).

Although the separation filter update in the ICA part is not real-time processing but involves totally a latency of 3.0 seconds, the entire system still seems to run in real-time because DS, spectral subtraction and separation filtering can work in the current segment with no delay. In the system, the performance degradation due to the latency problem in ICA is mitigated by oversubtraction in the spectral subtraction. Detailed *real-time* signal processing is shown below.

B. ICA part in real-time algorithm

In the ICA part of this algorithm, a sequential time-series input is divided into fixed-length blocks, and ICA is performed in each block. The number of samples in one block, l_{sample}, is defined as

$$l_{\mathrm{sample}} = \left\lfloor \frac{l_{\mathrm{sec}}\cdot f_{\mathrm{s}}}{T_{\mathrm{shift}}} \right\rfloor, \tag{12}$$

where l_{sec} is block length in seconds (we use 1.5 s in this paper), T_{shift} is frame shift size for short-time Fourier transform, and $\lfloor\cdot\rfloor$ is the floor function. Thus, a set of time frame index belonging to a block b (= 0, 1, 2, ...), T_b, can be given as

$$T_b = \{\, \tau \mid b\cdot l_{\mathrm{sample}} \leq \tau < (b+1)\cdot l_{\mathrm{sample}} \}. \tag{13}$$

Figure 6 shows the relation between a time frame index and a block index, where, e.g., $l_{\mathrm{sample}} = 5$.

The unmixing matrix for a block b, $\boldsymbol{W}^{\mathrm{ICA}}_{(b)}(f)$, is optimized by the following iterative update equation:

$$\left[\boldsymbol{W}^{\mathrm{ICA}}_{(b)}(f)\right]^{[p+1]} = \mu[\boldsymbol{I} - \langle\boldsymbol{\varphi}(\hat{\boldsymbol{o}}(f,\tau))\hat{\boldsymbol{o}}^{\mathrm{H}}(f,\tau)\rangle_{\tau\in T_b}]\left[\boldsymbol{W}^{\mathrm{ICA}}_{(b)}(f)\right]^{[p]} + \left[\boldsymbol{W}^{\mathrm{ICA}}_{(b)}(f)\right]^{[p]}, \tag{14}$$

where $\langle\cdot\rangle_{\tau\in T_b}$ is the time-averaging operator which is localized within block T_b, and $\hat{\boldsymbol{o}}(f,\tau) = [\hat{o}_1(f,\tau),\ldots,\hat{o}_K(f,\tau)]^{\mathrm{T}}$ is the temporal separated signal vector given as

$$\hat{\boldsymbol{o}}(f,\tau) = \boldsymbol{W}^{\mathrm{ICA}}_{(b)}(f)\boldsymbol{x}(f,\tau) \quad (\tau\in T_b). \tag{15}$$

Here, if the average power of the specific block b is very small, the unmixing matrix should not be updated because the low-power block which does not contains any dominant signals leads to an unstable convergence of the unmixing matrix. Thus, we do not update the unmixing matrix in such a block b if the average power of the block b is very small. This can be represented by

$$\boldsymbol{W}^{\mathrm{ICA}}_{(b)}(f) = \boldsymbol{W}^{\mathrm{ICA}}_{(b-1)}(f) \quad (\text{If } \langle|x(f,\tau)|^2\rangle_{\tau\in T_b} < th_{\mathrm{pow}}), \tag{16}$$

where th_{pow} is the threshold for the average power.

Moreover, the initial value of the unmixing matrix in the optimization at each block is represented by

$$\left[\boldsymbol{W}^{\mathrm{ICA}}_{(b)}(f)\right]^{[0]} = \begin{cases} \boldsymbol{W}_{\mathrm{initial}}(f) & (\text{if } b \bmod b_{\mathrm{reset}} = 0), \\ \boldsymbol{W}^{\mathrm{ICA}}_{(b-1)}(f) & (\text{otherwise}), \end{cases} \tag{17}$$

where b_{reset} is the reset period of the unmixing matrix, and $\boldsymbol{W}_{\mathrm{initial}}(f)$ is the initial value of the unmixing matrix

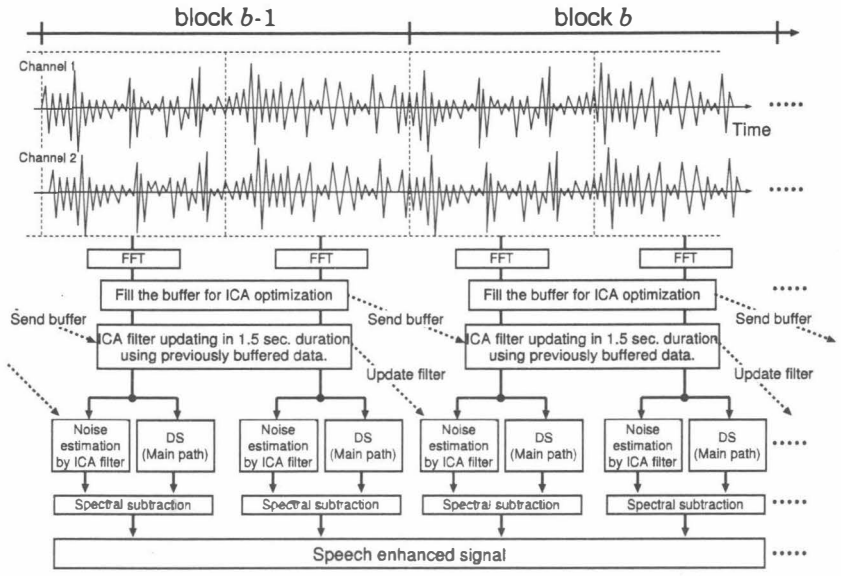

Fig. 5. Signal flow in real-time implementation of proposed method.

given in advance. This initial value is ordinarily generated using the observed signal via some methods, e.g., principle component analysis or beamforming. Thus, the optimized unmixing matrix is reset into the given initial value every b_{reset} blocks.

Furthermore, we can estimate DOAs from the unmixing matrix $\boldsymbol{W}_{(b)}^{\text{ICA}}(f)$ [11]. This procedure is represented by

$$\theta_{u,b} = \sin^{-1}\left(\frac{\left[[\boldsymbol{W}_{(b)}^{\text{ICA}}(f)]^{-1}\right]_{ju} / \left[[\boldsymbol{W}_{(b)}^{\text{ICA}}(f)]^{-1}\right]_{j'u}}{2\pi f_s c^{-1}(d_j - d_{j'})}\right), \quad (18)$$

where $\theta_{u,b}$ is the DOA of the u-th sound source in the block b. Then, we choose the U-th source signal which is the nearest the front of the microphone array, and designate the DOA of the chosen source signal as $\theta_{U,b}$ in this paper. This is because almost all users often stand in front of the microphone array in a spoken-oriented human-machine interface.

C. Noise reduction part in real-time algorithm

Noise reduction is carried out according to the following three steps;

1) First, we perform DS beamforming to enhance the target signal (primary path).
2) Next, we estimated noise signal based on ICA (reference path).
3) Finally, we obtain the target speech enhanced signal by subtracting the power spectrum of the estimated noise from the power spectrum of the primary path's output.

In the primary path, DS is performed to enhance the target speech signal. This procedure can be represented by

$$y_{(b)}^{\text{DS}}(f,\tau) = \boldsymbol{g}_{\text{DS}}^{\text{T}}(f,\theta_{U,b-2})\boldsymbol{x}(f,\tau) \ \ (\tau \in T_b), \quad (19)$$

where $y_{(b)}^{\text{DS}}(f,\tau)$ is the primary path's output in a block b.

In the reference path, first, the signal separation is performed. This can be designated as

$$\boldsymbol{o}_{(b)}(f,\tau) = \boldsymbol{W}_{(b-2)}^{\text{ICA}}(f)\boldsymbol{x}(f,\tau) \ \ (\tau \in T_b), \quad (20)$$

where $\boldsymbol{o}_{(b)}(f,\tau) = [o_{1,b}(f,\tau),\ldots,o_{K,b}(f,\tau)]^{\text{T}}$ is the separated signal vector in a block b. Next, we obtain the estimated noise signal in a block b, $z_{(b)}(f,\tau)$, as

$$z_{(b)}(f,\tau) = \boldsymbol{g}_{\text{DS}}^{\text{T}}(f,\theta_{U,b-2})\left[\boldsymbol{W}_{(b-2)}^{\text{ICA}}(f)\right]^{+}\boldsymbol{q}_{(b)}(f,\tau) \ (\tau \in T_b), \quad (21)$$

$$\boldsymbol{q}_{(b)}(f,\tau) = [o_{1,b}(f,\tau),\ldots,o_{U-1,b}(f,\tau),0, o_{U+1,b}(f,\tau),\ldots,o_{K,b}(f,\tau)]^{\text{T}}, \quad (22)$$

where $\boldsymbol{q}_{(b)}(f,\tau)$ is the vector in which the target speech component is removed.

Finally, we obtain the target speech enhanced signal $y_{(b)}^{\text{BSSA}}(f,\tau)$ by spectral subtraction. This can be given as

$$y_{(b)}^{\text{BSSA}}(f,\tau) = \begin{cases} \left\{|y_{(b)}^{\text{DS}}(f,\tau)|^2 - \beta\cdot|z_{(b)}(f,\tau)|^2\right\}^{\frac{1}{2}}, \\ \quad (\text{if } |y_{(b)}^{\text{DS}}(f,\tau)|^2 - \beta\cdot|z_{(b)}(f,\tau)|^2 \geq 0) \\ \gamma\cdot|y_{(b)}^{\text{DS}}(f,\tau)| \quad (\text{otherwise}). \end{cases} \quad (23)$$

In (19) and (21), note that we have only to use the estimated DOA and the optimized unmixing matrix in the previous block $b-2$. This is due to data buffering and optimization process for ICA. ICA optimization requires a certain length of data, e.g., 1.5 s. data. Thus, we must buffer a certain length of input data for ICA optimization. Consequently, ICA optimization just starts after the buffering. Moreover, ICA optimization cannot finish in no time at all because ICA optimization consumes huge amount of computations. Thus ICA optimization is performed while one block. As a result, in a current block b, we are only admitted to utilize the separation filter optimized in the block $b-2$ (see Fig. 7). By the same manner, we can only apply the estimated DOA of the block $b-2$ to a current block b.

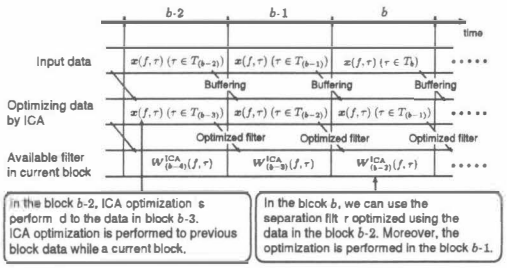

Fig. 7. Configuration of updating separation filter.

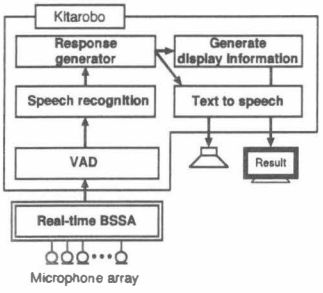

Fig. 8. Overview of hands-free robot spoken dialogue system with real-time BSSA.

IV. Hands-Free Robot Spoken Dialogue System with Real-Time BSSA

A. Overview

We introduce the real-time BSSA into the robot spoken dialogue system "Kitarobo" [12] which has already been installed in an actual railway station. In this paper, we replace the input device of Kitarobo, i.e., a close-talking microphone, with the real-time BSSA to construct the hands-free robot spoken dialogue system. Figures 8 and 9 show an overview and appearance of the hands-free robot spoken dialogue system with the real-time BSSA. Unlike the conventional Kitarobo, the input device is substituted with the real-time BSSA. Details of Kitarobo is described in the following subsection.

B. Robot Spoken Dialogue System "Kitarobo"

The spoken-oriented guidance robot "Kitarobo" is working in an actual railway station since end of March 2006. The system is installed besides the ticket gate and is adjacent to each other. Everybody can use the systems while the station is open. Since the station faces to a road, an automobile engine sound and sound of a bus horn are also inputted to the system. Kitarobo provides guidance information to visitors regarding issues on the station or around the station without resting. The input device of the original Kitarobo is a close-talking microphone. Thus the original Kitarobo is not a hands-free system and is weak against the surrounding noises. Reference [12] helps you to understand further details of Kitarobo.

V. Experiment and Result

A. Simulating railway-station noise

The main task of Kitarobo is a station guidance, and always working in an actual railway-station. Thus, it is

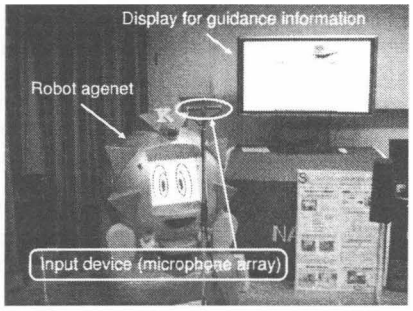

Fig. 9. Appearance of our hands-free robot spoken dialogue system with real-time BSSA.

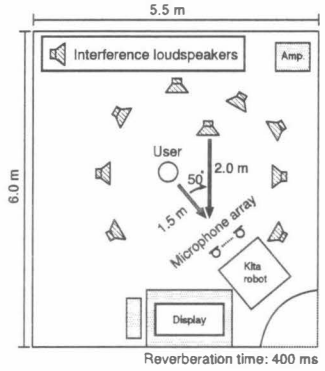

Fig. 10. Layout of reverberant room in our experiment.

difficult to conduct various BSSA experiments in an arbitrary time. Therefore, we have a necessity to construct the noise environment simulator of railway-station for experiments. To solve the problem, we have constructed the experimental room for hands-free spoken dialogue system with the real-time BSSA. The experimental room contains Kitarobo with the real-time BSSA and railway-station noise simulator. The noise simulation is performed in the following;

1) Record noises in an actual railway station. In the experiment, eight-channel directional microphones are used to record the multi-channel railway-station noise.
2) Playback the multi-channel recorded railway-station noise by eight surrounded loudspeaker (see Fig. 10).

This noise consists of various kinds of interference noises, namely, background noise, sounds of trains, ticket-vending machines, automatic ticket wickets, foot steps, cars, and wind. In addition, this noise is highly nonstationary.

B. Experimental Setup

To evaluate the hands-free spoken dialogue system with the real-time BSSA, the speech recognition test was conducted. Figure 10 depicts a layout of a reverberant room in our experiment where the reverberation time is about 400 ms. The following real-recored 16 kHz-sampled signals were used in the experiments. The target signal is user's speech which is talked in front of a microphone array and 1.5 m apart from the array. As for noise, two noises were added simultaneously. First noise is the real-recored noise in an actual railway-station noise (it simulates railway-station

noise) emitted from surrounded 8 loudspeakers. Second noise is an interference speech located at 50 degrees in the right direction of the microphone array, and its distance is 2.0 m.

We use 5 speakers (250 words) as target user, and Julius [13] ver. 4.0 RC2 as speech decoder. A eight-element array with the interelement spacing of 2 cm is used. The array consists of directional microphone SHURE MX-184. DFT size is 512 points, window length for ICA is 256 points, and window shift size is 128 points in the experiment. The proposed real-time BSSA is run on Intel Xeon X5355 with 2.66 GHz and requires 64 Mbytes RAM. However, we have succeeded at the real-time implementation of ICA on general purpose DSP [14]. The computational complexity of the proposed real-time BSSA is almost the same as the real-time ICA, i.e., DS and spectral subtraction are only added compared with the real-time ICA. Thus, it can be expected that the real-time BSSA is also implemented on general purpose DSP. Besides, RME Hammerfall DSP Multiface is used for 8-channel AD/DA.

The algorithm delay only depends on the following; (a) DS filtering, (b) noise estimation by the separation filter, and (c) hardware limitation for reading size of the input signal. Although ICA optimization is parallelly performed, the optimization result cannot be applied to current block. Thus, ICA optimization does not yields the algorithm delay. In DS and the separation filter, for reducing the effect of the circular convolution, the main pulse of the filter is located at the center of the filter. Thus, the resultant signal of the filtering is delayed, and its delay is the half of the filter length. Note that the noise estimation is performed in parallel with DS. Therefore, the total delay of DS filtering and noise estimation is also the half of the filter length. Moreover, the hardware limitation for reading size of the input signal exists. In the experiment, the signal can be read with 512 points. Consequently, the algorithm delay of the final output can be given by

$$\text{Delay [points]} = \text{Read size} + \text{Filter Size}/2. \quad (24)$$

Now, since we use 512-point filter, the algorithm delay of the final output of the real-time BSSA is 768 points. This corresponds to 48 ms delay with 16 kHz sampling.

C. Experimental result

We compared DS, the conventional ICA, and the proposed real-time BSSA on the basis of the speech recognition test. Figures 11 shows speech recognition result. From this result, we can see that both the word correct and word accuracy of the proposed BSSA are obviously superior to those of DS and the conventional ICA. In particular, 8% (in word correct) or 6% (in word accuracy) improvement of the speech recognition result can be confirmed. Thus, it is a promising evidence that the response accuracy of the spoken dialogue system will be increased with the real-time BSSA.

The demonstration movie of the constructed hands-free spoken dialogue system with the real-time BSSA is available in the following URL.

Demo: http://spalab.naist.jp/database/Demo/rtbssa/

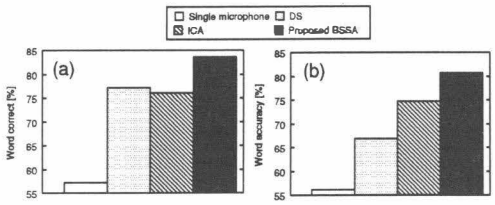

Fig. 11. Result of speech recognition test in (a) word correct, and (b) word accuracy.

VI. Conclusion

In this paper, we propose the real-time architecture of BSSA which is our previously proposed blind source extraction method. Also, we introduce the real-time BSSA into spoken-oriented guidance system "Kitarobo", and construct the hands-free robot spoken dialogue system. Finally, we evaluate the constructed system based on the speech recognition test. As a result, we can see that the speech recognition performance of the hands-free spoken dialogue system with the real-time BSSA is outperforms those of DS-, and ICA-based hands-free spoken dialogue based systems. Thus, it is expected that the response accuracy of the hands-free spoken dialogue system with the real-time BSSA is increased.

In the future, we will introduce more efficient post-filtering, e.g., Winer filter, into the proposed real-time BSSA instead of spectral subtraction. In addition, we will try to integrate the proposed real-time BSSA and speech-recognition-based approach, e.g. missing feature theory [3].

References

[1] K. Nakadai, D. Matsuura, H. Okuno, and H. Kitano, "Applying scattering theory to robot audition system: robust sound source localization and extraction," *Proc. IROS2003*, pp.1147–1152, 2003.

[2] R. Prasad, H. Saruwatari, and K. Shikano, "Robots that can hear, understand and talk," *Advanced Robotics*, vol.18, pp.533–564, 2004.

[3] H. G. Okuno and S. Yamamoto, "Computing for Computational Auditory Scene Analysis," *Journal of The Japanese Society for Artificial Intelligence* , Vol.22, No.6 (Nov. 2007) pp. 846-854.

[4] P. Comon, "Independent component analysis, a new concept?," *Signal Processing*, vol.36, pp.287–314, 1994.

[5] P. Smaragdis, "Blind separation of convoluted mixtures in the frequency domain," *Neurocomputing*, vol.22, pp.21–34, 1998.

[6] S. Ikeda and N. Murata, "A method of ICA in the frequency domain," *Proc. Intern. Workshop on ICA and BSS*, pp.365–371, 1999.

[7] H. Saruwatari, et al., "Blind source separation combining independent component analysis and beamforming," *EURASIP J. Applied Signal Proc.*, vol.2003, no.11, pp.1135–1146, 2003.

[8] S. Araki, et al., "The fundamental limitation of frequency domain blind source separation for convolutive mixtures of speech," *IEEE Transactions on Speech and Audio Processing*, Vol.11, No.2, pp.109-116, 2003.

[9] Y. Takahashi et al., "Blind spatial subtraction array with independent component analysis for hands-free speech recognition," *Proc. of IWAENC*, 2006.

[10] S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoustics, Speech, Signal Proc.*, vol.ASSP-27, no.2, pp.113–120, 1979.

[11] H. Sawada et al., "A robust and precise method for solving the permutation problem of frequency-domain blind source separation," *IEEE Trans. Speech and Audio Processing*, vol.12, pp.530–538, 2004.

[12] H. Kawanami, et al., "Development and Operational Result of Real Environment Speech-oriented Guidance Systems Kita-robo and Kita-Chan," *Oriental COCOSDA 2007*, pp.132–136, 2007.

[13] A. Lee, et al., "Julius – An open source real-time large vocabulary recognition engine," *Proc. EUROSPEECH*, pp.1691–1694, 2001.

[14] Yoshimitsu Mori, et al., "Blind Separation of Acoustic Signals Combining SIMO-Model-Based Independent Component Analysis and Binary Masking," *EURASIP Journal on Applied Signal Processing*, Article ID 34970, 17 pages, 2006.