The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Sound Source Separation and Automatic Speech Recognition for Moving Sources

Kazuhiro Nakadai, Hirofumi Nakajima, Gökhan Ince, and Yuji Hasegawa

Abstract— **This paper addresses sound source separation and speech recognition for moving sound sources. Real-world applications such as robots should cope with both moving and stationary sound sources. However, most studies assume only stationary sound sources. We introduce three key techniques to cope with moving sources, that is, Adaptive Step-size control (AS), Optima Controlled Recursive Average (OCRA), and Separation Parameter Switching (SPS). We implemented a real-time robot audition system with these techniques for our humanoid robot with an 8ch microphone array by using HARK which is our open-source software for robot audition. Preliminary results show that the performance of recognition of moving sound sources improved drastically, and also the performance of the system is shown through two speech dialog scenarios which requires sound source separation and automatic speech recognition for moving sources.**

I. Introduction

Robot audition is an active research area which realizes natural human-robot interaction in a daily environment. The main claim in robot audition is listening to several things simultaneously using a robot's own ears [1]. However,various types of sound sources coexist with a target speech source. Thus, one of the hottest research topics in robot audition is sound source separation and speech enhancement.

Nakadai et al. reported an Active Direction-Pass Filter for binaural processing [2]. It can separate three simultaneous sound sources by using two microphones. However, the separation performance is poor in the real world due to reverberation and background noise. Valin et al. reported Geometric Sound Source Separation, which is a hybrid algorithm between beamforming and blind separation [3]. Their GSS implementation was extended to support online processing, and Yamamoto et al. integrated their GSS with ASR by using Missing Feature Theory (MFT) [4]. They finally showed speech recognition of three simultaneous speeches. Hara et al. reported an adaptive beamformer based on sound source separation which was combined with ASR [5]. They showed a voice based TV control task in an office environment. Saruwatari et al. developed SIMO-ICA which can separate a mixture of sound sources with a pair of microphones [6]. They also developed special hardware to realize small and real-time processing. This is used in a robot called Kita-chan with a speech dialog system, which is used in the Ikoma city center as a navigator. However, this system considered only stationary cases where a robot and users are fixed when they are speaking.

K. Nakadai and G. Ince are with Honda Research Insititute Japan Co., Ltd., 8-1 Honcho, Wako, Saitama 351-0114, JAPAN, and also with Graduate School of Information Science and Engineering, Tokyo Institute of Technology {nakadai,ince.gokhan}@jp.honda-ri.com

H. Nakajima, and Y. Hasegawa are with Honda Research Insititute Japan Co., Ltd., 8-1 Honcho, Wako, Saitama 351-0114, JAPAN {nakajima,yuji.hasegawa}@jp.honda-ri.com

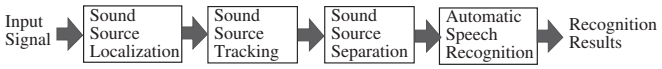


Fig. 1. A Basic Flow of Robot Audition Systems

There are a lot of issues in robot audition systems to support moving sound source recognition. The basic processing flow of a typical robot audition system is shown in Fig. 1. There are mainly three processing blocks such as *Sound Source Localization (SSL)*, *Sound Source Tracking (SST)* and *Sound Source Separation (SSS)* before an ASR block. These preprocessing blocks should consider moving sound source situations. For SSL and SST, some studies mentioned mobile functions of robot audition, that is, low-level active audition [2], [7], [8]. However, their mobile functions are still limited to localization and tracking.

For SSS, Nakadai et al. reported the first result of separation and speech recognition for a dynamic environment [9]. They developed two techniques to improve SSS: *Adaptive Step-size control (AS)* which controls step-size parameter optimally [10], and *Optima Controlled Recursive Average (OCRA)* which precisely estimates a separation matrix [9]. They introduced these techniques to *Geometric Source Separation (GSS)* which is an SSS algorithm by integrating *Blind Source Separation (BSS)* and beamforming reported in [3]. They showed that GSS improved around 5 dB in *Signal-to-Noise Ratio (SNR)*, and ASR performance improved 15 points in isolated word recognition of 200 words for moving sources after performing GSS. However, this method indirectly dealt with moving sound sources, because they focused only on the convergence of the separation matrix. In addition, they showed the effectiveness of their method only for short utterances like isolated words.

Therefore, in this paper, we propose a new method to improve SSS for moving sources more directly so that longer utterances like sentences can be recognized with ASR. There is room to improve ASR for moving sources in the ASR block in Fig. 1. The most common technique is noise adaptation of an acoustic model for ASR such as multi-condition training. Thus, we also use a noise adaptation technique with the proposed method.

The rest of this paper is organized as follows: Section II describes issues in dealing with moving sources for robots. Section III proposes our new methods to solve the issues. Section IV shows our robot audition system introduced the proposed method. Section V evaluates the system to show

the effectiveness of our proposed method for moving sources. The last section gives the conclusion.

II. Issues in Dealing with Moving Sources

In order to explain issues in dealing with moving sources, online GSS [3] that we are using as a base algorithm is formulated as a proxy for online SSS methods. It is promising as one of the online adaptive SSS algorithms for robot audition, because it requires a smaller calculation cost than the other SSS algorithms.

A. Formulation of online GSS

Suppose that there are *M* sources and *N* ($\geq$ *M*) microphones. A spectrum vector of *M* sources at frequency ω, $\mathbf{s}(\omega)$, is denoted as $[s_1(\omega)\ s_2(\omega)\ \cdots\ s_M(\omega)]^T$, and a spectrum vector of signals captured by the *N* microphones at frequency ω, $\mathbf{x}(\omega)$, is denoted as $[x_1(\omega)\ x_2(\omega)\ \cdots\ x_N(\omega)]^T$, where T represents a transpose operator. $\mathbf{x}(\omega)$ is, then, calculated as

$$\mathbf{x}(\omega) = \mathbf{D}(\omega)\mathbf{s}(\omega), \tag{1}$$

where $\mathbf{D}(\omega)$ is a transfer function (TF) matrix. Each component H_{nm} of the TF matrix represents the TF from the m-th source to the n-th microphone. The source separation is then formulated as

$$\mathbf{y}(\omega) = \mathbf{W}(\omega)\mathbf{x}(\omega), \tag{2}$$

where $\mathbf{W}(\omega)$ is called a *separation matrix*. The separation with the general SSS is defined as finding $\mathbf{W}(\omega)$ which satisfies the condition that output signal $\mathbf{y}(\omega)$ is the same as $\mathbf{s}(\omega)$. In order to estimate $\mathbf{W}(\omega)$, GSS introduces two cost functions, that is, separation sharpness (J_{SS}) and geometric constraints (J_{GC}) defined by

$$J_{SS}(\mathbf{W}) = \|E[\mathbf{y}\mathbf{y}^H - \mathrm{diag}[\mathbf{y}\mathbf{y}^H]]\|^2 \tag{3}$$

$$J_{GC}(\mathbf{W}) = \|\mathrm{diag}[\mathbf{W}\mathbf{D} - \mathbf{I}]\|^{\mathbf{2}} \tag{4}$$

where $\|\cdot\|^2$ indicates the Frobenius norm, $\mathrm{diag}[\cdot]$ is the diagonal operator, $E[\cdot]$ represents the expectation operator and H represents the conjugate transpose operator. The total cost function $J(\mathbf{W})$ is represented as

$$J(\mathbf{W}) = \alpha J_{SS}(\mathbf{W}) + J_{GC}(\mathbf{W}), \tag{5}$$

where α means the weight parameter between the costs of separation and geometric constraint, which is usually set to $\|\mathbf{x}^H\mathbf{x}\|^{-2}$ according to [3].

When a long sequence of $\mathbf{x}$ can be used, we can directly estimate the best $\mathbf{W}$ by minimizing $J(\mathbf{W})$ in an offline manner. However, a robot needs to work in real time, and the best $\mathbf{W}$ is always changing in the real world. Thus, the online GSS adaptively updates $\mathbf{W}$ by using

$$\mathbf{W}_{t+1} = \mathbf{W}_t - \mu_{SS}\mathbf{J'}_{SS}(\mathbf{W_t}) + \mu_{GC}\mathbf{J'}_{GC}(\mathbf{W_t}). \tag{6}$$

where $\mathbf{W}_t$ denotes $\mathbf{W}$ at the current time step t, $\mathbf{J'_{SS}}(\mathbf{W})$ and $\mathbf{J'_{GC}}(\mathbf{W})$ are complex gradients [11] of $\mathbf{J_{SS}}(\mathbf{W})$ and $\mathbf{J_{GC}}(\mathbf{W})$, which decide an update direction of $\mathbf{W}$. μ_{SS} and μ_{GC} are called step-size parameters.

B. Problems in online GSS

The online GSS has three issues in dealing with a dynamically-changing environment like moving sources. These issues are regarding estimation of separation matrix $\mathbf{W}$ as follows:

1) fast adaptation of SSS parameters to dynamic changes (the robustness of convergence of $\mathbf{W}$),
2) precise estimation of SSS parameters (the accuracy of converged $\mathbf{W}$),
3) reset of SSS parameters according to dynamic changes (discontinuity of $\mathbf{W}$ and $\mathbf{D}$ due to motion).

The first one is related to the fact that the step-size parameters μ_{SS} and μ_{GC} are fixed values decided heuristically or empirically, although they should be frequency-dependent and time-variant values according to environmental changes. When these values are too large, perturbation around the optimal $\mathbf{W}$ occurs, or $\mathbf{W}$ is sometimes diverged. When these are too small, $\mathbf{W}$ behaves like a matrix with fixed values, and thus, it is difficult to adapt to dynamical changes of the optimal $\mathbf{W}$. Therefore, the optimal design of these step-size parameters is the key to realize faster adaptation of $\mathbf{W}$ to the optimal $\mathbf{W}$.

The second is caused by the calculation error of $\mathbf{J'}_{SS}(\mathbf{W})$ in Eq. (6). In implementation, online GSS used an instantaneous product of $\mathbf{y}$ for incremental processing by omitting the expectation operation in Eq. (3). This produces an error in estimation of the optimal $\mathbf{W}$, and perturbation of $\mathbf{W}$ occurs because $\mathbf{W}$ tries to converge to the erroneous optimal $\mathbf{W}$.

In the last issue, two parameters for GSS were involved, i.e., a transfer function $\mathbf{D}$ and a separation matrix $\mathbf{W}$. When a sound source and a microphone array are stationary, $\mathbf{D}$ is fixed and Eq. (4) is easy to be calculated. However, for a moving sound source, $\mathbf{D}$ continuously changes according to the motions. Thus, $\mathbf{D}$ in Eq. (4) should be changed adaptively. There are two ways to obtain $\mathbf{D}$ adaptively. One is to calculate $\mathbf{D}$ from the geometric relationship between the sound source and the microphone array. The other is to discretely change $\mathbf{D}$ by selecting the most appropriate one from the TF database. The former is an intuitive method because it provides continuous $\mathbf{D}$ according to the motion of the sound source. The calculation is easy in free acoustic space, but for a robot-embedded microphone array, the effect of a robot's head and body has to be considered. Indeed, a finite element method and a boundary element method can give accurate TFs by taking such effect into account, but their real-time calculation is difficult. The latter requires a lot of impulse response measurements to construct the TF database. Since it is based on measurement, the SSS performance is better than the former one. In this method, the timing for the change of $\mathbf{D}$ is a crucial issue, because such a change makes discontinuity of $\mathbf{D}$. Thus it might affect SSS performance badly if the changed timing is wrong. $\mathbf{W}$ is updated by using Eq. (6). When a sound source is stationary, the change of the optimal $\mathbf{W}$ is small. However, when a sound source direction is changed, $\mathbf{D}$ which is used to calculate $\mathbf{J'_{GC}}(\mathbf{W})$ in Eq. (6) is changed as mentioned

above. Thus, the optimal $\mathbf{W}$ varies according to the change of $\mathbf{D}$. In other words, the optimal $\mathbf{W}$ is almost fixed for a specific sound source direction. Just after a sound source direction moves from θ_1 to θ_2, $\mathbf{W}$ is converged to the optimal $\mathbf{W}$ for θ_1. Thus the SSS performance deteriorates. However, when the system initializes $\mathbf{W}$ to be the optimal $\mathbf{W}$ for θ_2, the SSS performance is maintained. In this case, initialization timing of $\mathbf{W}$ is the key.

C. Approaches to solve the problems in online GSS

For the first issue, our reported *Adaptive Step-size (AS)* method [10] is effective because it controls both μ_{SS} and μ_{GC} optimally. AS is well-studied in the field of echo cancellation [12]. We extended AS to support multi-channel input and complex number signals by using the multi-dimensional version of Newton's method and linear approximation formula for a complex gradient matrix[10]. With this method, these step-size parameters become large values when a separation error is high, for example, due to source position changes. These will have small values when the error is small due to the convergence of the separation matrix. Thus, step-size parameters are automatically controlled to be optimal values.

For the second issue, we propose to use *Optima Controlled Recursive Average (OCRA)* [9] which makes the convergence of the separation matrix smoother, and improves the separation performance of stationary states. OCRA estimates a precise correlation matrix by using an adaptively controlled window. In online systems, correlation matrix $\mathbf{R}_{xx}$ at time frame t is estimated as $\hat{\mathbf{R}}_{xx}(t)$ from a partial signal of $\mathbf{x}(t)$ by using a time window $w(\cdot)$.

$$\begin{aligned}\hat{\mathbf{R}}_{xx}(t) &= w(t) * [\mathbf{x}(t)\mathbf{x}^H(t)] \\ &= \sum_{\tau=0}^{\infty} w(\tau)[\mathbf{x}(t-\tau)\mathbf{x}^H(t-\tau)]. \end{aligned} \quad (7)$$

For better estimations, the window length must be long, however, it makes the adaptation speed slow. Therefore, it is necessary to set the optimal length for required precision. Because the required precision is proportional to the separation sharpness, we propose an optimal control method for window length defined by

$$N(t) = \left(\beta \cdot \min[E[\mathbf{y}\mathbf{y}^H - \mathrm{diag}[\mathbf{y}\mathbf{y}^H]]]\right)^{-2}, \quad (8)$$

where $N(t)$ is the window length when $w(t)$ is a rectangular window, β is an allowable error parameter which is empirically set to 0.99, and $\min[\mathbf{A}]$ represents the minimum element's value in matrix $\mathbf{A}$. To avoid an extraordinary long window, we introduced the maximum value of $N(t)$, $N_{max}(= 1,000)$ in Fig. 2. The decay parameter α for the exponential window which is equivalent to the rectangular length $N(t)$ is defined as

$$\alpha(t) = (N(t) - 1)/(N(t) + 1). \quad (9)$$

Finally, the correlation matrix is recursively estimated by using OCRA with the exponential window defined by

$$\hat{\mathbf{R}}_{xx}(t) = \alpha\hat{\mathbf{R}}_{xx}(t-1) + (1-\alpha)\mathbf{x}\mathbf{x}^H. \quad (10)$$

For the last issue, we newly propose *Separation Parameter Switching (SPS)*. SPS for transfer functions assumes the second way to use the TF database for $\mathbf{D}$, because the use of measured TF provides better performance than that of the calculated TF. As mentioned above, in this case, the switching timing of $\mathbf{D}$ is crucial. SPS is also applied to control the initialization timing of $\mathbf{W}$. We, thus, propose two and three modes for switching $\mathbf{D}$ and $\mathbf{W}$, respectively. They are described in the next section.

III. Separation Parameter Switching

We newly propose to control switching and initialization timings of $\mathbf{D}$ and $\mathbf{W}$ based on an utterance ID and sound source direction.

A. Switching Timing Control of a Transfer Function

For SPS for the transfer function $\mathbf{D}$, we propose the following two modes:

POS: Switching timing is controlled by sound source direction. When the direction of a sound source changes more than θ_{th} compared to the direction estimated at the previous time frame, the transfer function is replaced with the one which is the closest to the current sound source direction. Since our TF database was made for every 5 degrees, θ_{th} was set to be 5 degrees.

ID: Switching timing is controlled by utterance ID. In this case, the same $\mathbf{D}$ is used for an utterance even when the sound source direction changes drastically. At the beginning of the utterance, that is, when a new utterance ID is assigned, $\mathbf{D}$ corresponding to the closest to the current direction is used.

B. Initialization Timing Control of a Separation Matrix

For SPS for the separation matrix $\mathbf{W}$, we propose the following three modes:

POS: Initialization timing is controlled by sound source direction. When the direction interval between two consecutive frames is over θ_{th}, $\mathbf{W}$ is initialized. Likewise SPS for $\mathbf{D}$, θ_{th} was 5 degrees. When the system does not have converged $\mathbf{W}$ for the current sound source direction θ, the initialization defined in [3] is used as follows:

$$\mathbf{W}(\theta) = [diag[\mathbf{D}^H\mathbf{D}]]^{-1}\mathbf{D}^H. \quad (11)$$

When the system already has converged $\mathbf{W}$ for θ ($\mathbf{W}_{opt}(\theta)$), the current $\mathbf{W}$ is replaced with $\mathbf{W}_{opt}(\theta)$.

ID: The initialization timing is controlled by utterance ID. $\mathbf{W}$ is continuously updated without any initialization while the utterance ID is the same. Only when the utterance ID changes, the initialization occurs. The initialization is the same as the **POS** case, that is, the use of Eq. 11 or the selection of $\mathbf{W}_{opt}(\theta)$.

ID_POS: Switching timing is controlled both by sound source direction and by utterance ID. First, the utterance ID is checked. When it is changed, the sound source direction is checked. $\mathbf{W}$ is updated in the same way as the above two cases when the direction changes more

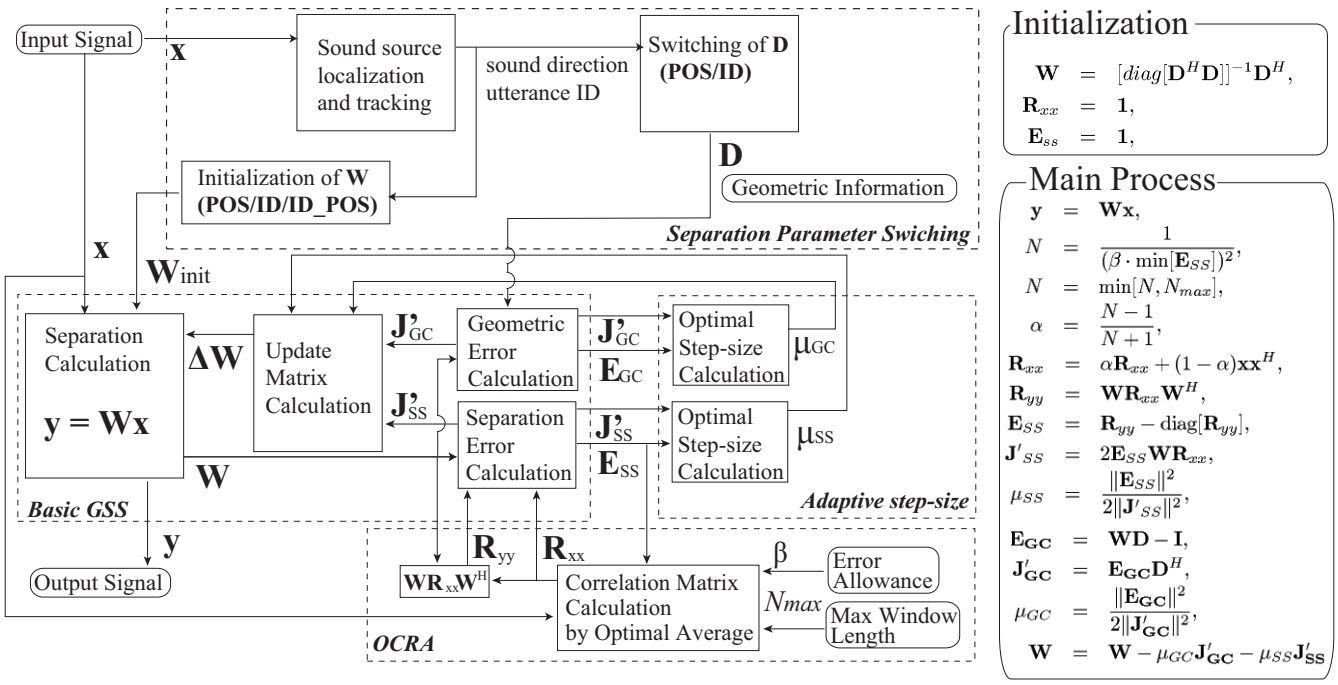


Fig. 2. Diagram of GSS with the proposed methods for moving sources

than 5 degrees. Otherwise, **W** is continuously updated without any initialization.

IV. Implementation

A. GSS for moving sound sources

Fig. 2 shows a diagram of GSS introducing AS, OCRA and SPS. The step-size and weight parameters are adaptively controlled as μ_{SS} and μ_{GC} with AS. Our GSS uses correlation matrices $\mathbf{R}_{xx}$ and $\mathbf{R}_{yy}$ estimated by OCRA instead of using the corresponding instantaneous products. The appropriate $\mathbf{D}$ is selected based on the changes of sound source direction and utterance ID according to the specified mode, **POS** or **ID**. $\mathbf{W}$ is initialized to $\mathbf{W}_{init}$ according to the mode specified in SPS for $\mathbf{W}$ such as **POS**, **ID** or **ID_POS**.

B. Real-time robot audition system

We implemented GSS depicted in Fig. 2 as a new module of HARK (Honda Research Institute Japan Audition for Robots with Kyoto University) which is our open source software for robot audition[1][13]. HARK consists of a complete set of modules for robot audition as component blocks on FlowDesigner[14][2], which works on Linux in real time. Many multi-channel sound cards are supported to build a real-time robot audition system easily. For preprocessing, sound source localization, tracking and separation are available. These preprocessing modules are able to be integrated with automatic speech recognition (ASR) based on Missing Feature Theory (MFT). For MFT, modules such as acoustic feature extraction for ASR, automatic missing feature mask generation, and ASR interface are prepared. Missing-feature-theory based ASR (MFT-ASR) is provided as a patch for Julius/Julian[15] which are Japanese open source speech recognition systems. Only MFT-ASR is implemented as a non-FlowDesigner module in HARK, but it connects with FlowDesigner by using modules from the ASR interface. Users are able to flexibly build robot audition systems by using the GUI interface. Fig. 3 shows our robot audition system using a newly-developed module for GSS with AS, OCRA and SPS.

The robot audition system was constructed for the Honda humanoid robot shown in Fig. 4. An 8 ch microphone array was embedded in the head. For each microphone, a newly-developed microphone module based on a MEMS microphone shown in Fig. 5 was used.

V. Evaluation

We evaluated our robot audition system with the proposed methods through ASR as follows:

Ex.1 performance of ASR for short utterances (words),
Ex.2 performance of ASR for long utterances (sentences),
Ex.3 two speech dialog scenarios

In every experiment, an acoustic model for ASR was trained with the Japanese Newspaper Article Sentences (JNAS) corpus consisting of over 60 hours of speech data. Since noise adaptation techniques are effective, we used multi-condition training, that is, both speech data separated by using GSS and clean speech data for training data.

In **Ex.1**, isolated word recognition for a stationary speech source, a moving speech source and a mixture of stationary and moving speech sources was performed. The stationary speaker stands at 60° left of the robot in a 4.0 m × 7.0 m room with 0.3–0.4 s of RT_{20}. The moving speaker moved around the robot from 0° to -90°. We asked two persons to utter 236 isolated words included in the robot's word database, that is, real speech data. We checked the effect

[1]"HARK" has a meaning of "listen" in old English. Available at http://winnie.kuis.kyoto-u.ac.jp/HARK/.

[2]http://flowdesigner.sourceforge.net/

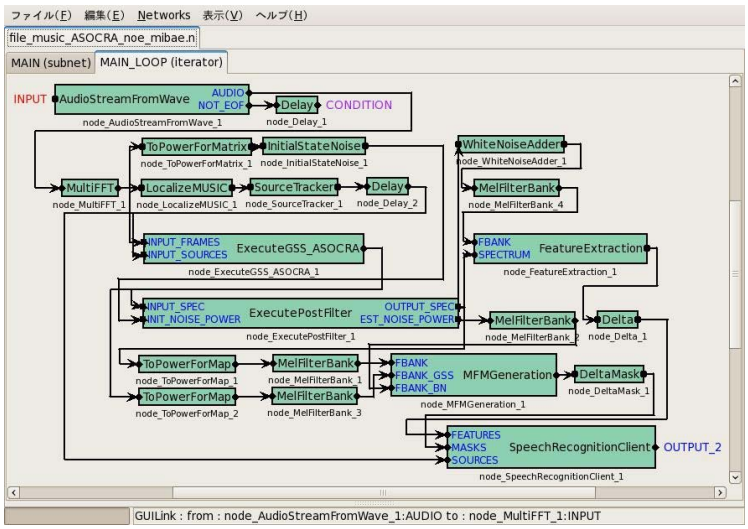


Fig. 3. HARK-based real-time robot audition system using the proposed GSS module (shown as "ExecuteGSS_ASOCRA" in the center)

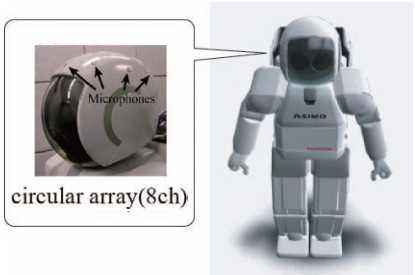


Fig. 4. An 8 ch microphone array embedded in robot's head. The microphones are circularly mounted on the head.

Fig. 5. A MEMS microphone module with an AAA battery. The center part of the module is microphone unit (Knowles SPM0406HE3H).

of AS and OCRA. SPS modes were fixed to be **POS** and **ID** for $\mathbf{D}$ and $\mathbf{W}$, respectively.

In **Ex.2**, Word Accuracy (WA) for sentence speeches is measured. WA is defined by

$$WA = \frac{N - S - D - I}{N} \tag{12}$$

where N is the number of words, I the number of insertion errors, D the number of deletion errors, S the number of substitution errors. We asked ten persons to utter 50 sentences selected from the ASJ phonetically-balanced Japanese sentence corpus. In this case, each person first stood in front of the robot, and then were asked the following two patterns: 1) to utter these sentences without motion, and 2) to walk around the robot within $\pm 20^\circ$ while they were speaking. In both cases, the robot was turned on, i.e., it generated ego-noise mainly from the back of the robot. Every combination for SPS was evaluated by using online GSS with AS and OCRA. For comparison, we also checked the case when calculation based TFs was used, which is mentioned as the first way to estimate $\mathbf{D}$ adaptively in Sec. II-B.

In **Ex.3**, we prepared two speech dialog scenarios. One is the case where a user greeted a robot while he was moving. The other is the case where a user asked a question while a robot's head is in rotation. In both cases, for the robot, a users' speech was regarded as moving sources.

A. Results

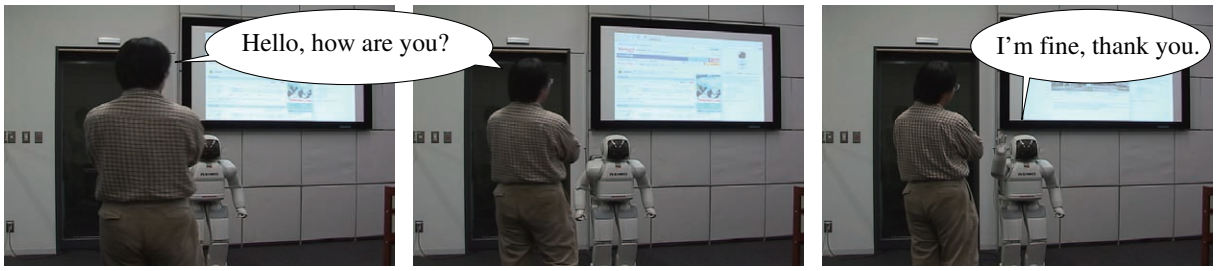
Tab. I shows the speech recognition results in **Ex.1**. For both stationary and moving single speech sources, the three methods have the same performance in speech recognition. This shows that even online GSS has the capability to deal with a moving source in less noisy cases, because W converged fast enough. However, in noisy cases where simultaneous speeches occur, online GSS is of less use. GSS with AS and GSS with AS and OCRA, thus, have better performance. Recognition of the separated speech for the stationary source also improved. We guess that this is caused by the leakage from the moving source, that is, a dynamically-changing noise. GSS with AS and GSS with AS and OCRA were able to deal with such a noise, while online GSS was not. We also found a further improvement in source separation by OCRA although the effect of OCRA was relatively small in **Ex.1**.

Tab. II shows the speech recognition results in **Ex.2**. First, sound source separation based on the TF database is obviously better than the calculation based one. For SPS, switching of $\mathbf{D}$ does not affect ASR performance. This is because the motion of each speaker was small, and such a small change was dealt with by AS and OCRA. However, the initialization of $\mathbf{W}$ slightly affected ASR performance. **ID** is better than others. This means that $\mathbf{W}$ should be continuously updated without initialization in the same utterance, but when the utterance changes $\mathbf{W}$ should be initialized. This experiment suggests that the switching of $\mathbf{D}$ might be more effective when the motion of the speaker is more active, and the initialization of $\mathbf{W}$ should be done only when a new utterance was observed.

Figs. 6 and 7 show the snapshots of the scenarios. In Fig. 6, the robot correctly recognized the moving speech source, and greeted the user. Fig. 7, the robot also recognized a user's question while its head was in rotation, and answered the question properly.

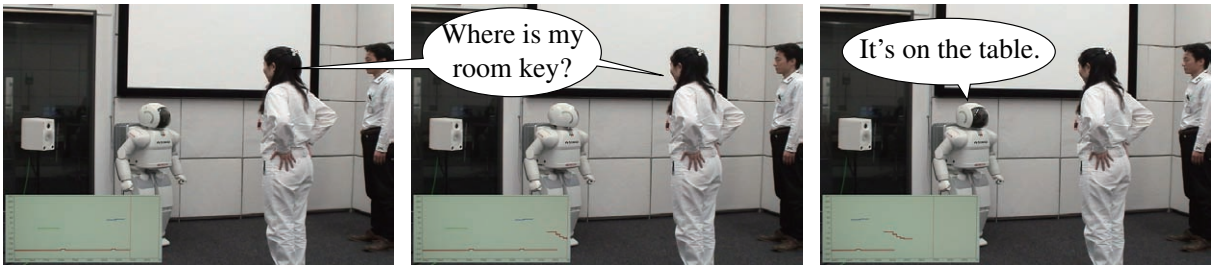
VI. Conclusion

We proposed three techniques, Adaptive Step-size control (AS), Optima Controlled Recursive Average (OCRA) and Separation Parameter Switching (SPS) to deal with moving sources, and investigated the effectiveness of these techniques. We showed that each method affects the performance of automatic speech recognition. In particular, AS was the most effective for dealing with moving sound sources. The effect of OCRA was small, but it still improved performance. For SPS, we evaluated all combinations of SPS modes,



a) A user says a greeting to a robot. b) His greeting was made while in motion. c) The robot correctly responded.

Fig. 6. Speech recognition of a moving speaker



a) A female starts asking a question while a robot is in motion.

b) Her question is regarded as a moving speech source.

c) The robot correctly answered the question.

Fig. 7. Speech recognition while a robot is in motion. The left bottom rectangle shows sound streams detected in the system. The vertical and horizontal axes show azimuth and time, respectively. Her speech was shown as a red curve, that is, a moving source, while other stationary streams were lines.

TABLE I

Ex.1: WORD CORRECT RATE OF ISOLATED WORD RECOGNITION (%)

		online GSS	w/ AS	w/ AS&OCRA
single	stationary	95.8	95.8	95.8
	moving	90.7	90.5	90.5
double (simultaneous)	stationary	58.3	72.3	73.1
	moving	60.2	72.9	74.4

TABLE II

Ex.2: WORD ACCURACY OF SENTENCE RECOGNITION (%)

D	W	stationary	moving
ID	ID	90.8	67.9
ID	POS	94.1	66.7
ID	ID_POS	93.9	66.0
POS	ID	90.8	67.8
POS	POS	94.2	66.8
POS	ID_POS	93.9	63.2
CALC	ID	58.2	59.1

and we obtained a suggestion that the switching of transfer functions is more effective when a user moves actively, and a separation matrix should be initialized only when a new utterance is observed. In addition, we developed a real-time robot audition system with the proposed techniques, and showed the effectiveness through two speech dialog scenarios. Our future work includes more detailed evaluation to obtain more concrete results in various dynamically-changing environments such as sentences, crossing speakers and situations with moving robots.

REFERENCES

[1] K. Nakadai, T. Lourens, H. G. Okuno, and H. Kitano, "Active audition for humanoid," in *Proc. of 17th National Conf. on Artificial Intelligence (AAAI-2000)*. AAAI, 2000, pp. 832–839.

[2] K. Nakadai *et al.*, "Improvement of recognition of simultaneous speech signals using av integration and scattering theory for humanoid robots," *Speech Communication*, vol. 44, no. 1-4, pp. 97–112, 2004.

[3] J.-M. Valin *et al.*, "Robust recognition of simultaneous speech by a mobile robot," *IEEE Trans. on Robotics*, vol. 23, no. 4, pp. 742–752, 2007.

[4] S. Yamamoto *et al.*, "Design and implementation of a robot audition system for automatic speech recognition of simultaneous speech," in *Proc. of the 2007 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU-2007)*. IEEE, Dec. 2007, pp. 111–116.

[5] I. Hara *et al.*, "Robust speech interface based on audio and video information fusion for humanoid HRP-2," in *Proc. of IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS 2004)*. IEEE, 2004, pp. 2404–2410.

[6] H. Saruwatari *et al.*, "Two-stage blind source separation based on ica and binary masking for real-time robot audition system," in *Proc. of IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS 2005)*. IEEE, 2005, pp. 209–214.

[7] Y. Sasaki *et al.*, "Daily sound recognition using pitch-cluster-maps for mobile robot audition," in *Proc. IROS 2009*, 2009, pp. 2724–2729.

[8] H.-D. Kim *et al.*, "Human tracking system integrating sound and face localization using em algorithm in real environments," *Advanced Robotics*, vol. 23, no. 6, pp. 629–653, 2007.

[9] K. Nakadai *et al.*, "Sound source separation of moving speakers for robot audition," in *Proc. of IEEE Int'l Conf. on Acoustics, Speech and Signal Processing (ICASSP 2009)*, 2009, pp. 3685–3688.

[10] H. Nakajima *et al.*, "Adaptive step-size parameter control for real-world blind source separation," in *Proc. of the 2008 IEEE Int'l Conf. on Acoustics, Speech and Signal Processing (ICASSP 2008)*. IEEE, Apr. 2008, pp. 149–152.

[11] D. Brandwood, "A complex gradient operator and its application in adaptive array theory," *IEE Proc.*, vol. 130, no. 1, pp. 251–276, 1983.

[12] S. Yamamoto and S. Kitayama, "An adaptive echo canceller with variable step gain method," *Trans. of the IECE of Japan*, vol. E65, no. 1, pp. 1–8, 1982.

[13] K. Nakadai *et al.*, "An open source software system for robot audition hark and its evaluation," in *Proc. of 2008 IEEE/RAS Int'l Conf. on Humanoid Robots (HUMANOIDS 2008)*, 2008, pp. 561–566.

[14] C. Côté *et al.*, "Reusability tools for programming mobile robots," in *Proc. of IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS 2004)*. IEEE, 2004, pp. 1820–1825.

[15] T. Kawahara and A. Lee, "Free software toolkit for Japanese large vocabulary continuous speech recognition," in *Int'l Conf. on Spoken Language Processing (ICSLP)*, vol. 4, 2000, pp. 476–479.