Making A Robot Recognize Three Simultaneous Sentences in Real-Time

Shun'ichi Yamamoto†, Kazuhiro Nakadai*, Jean-Marc Valin‡, Jean Rouat‡, François Michaud‡,
Kazunori Komatani†, Tetsuya Ogata†, and Hiroshi G. Okuno†

†Graduate School of Informatics, Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, Japan

*Honda Research Institute Japan Co., Ltd. 8-1 Honcho, Wako-shi, Saitama 351-0114, Japan nakadai@jp.honda-ri.com

‡Department of Electrical Engineering and Computer Engineering, Université de Sherbrooke, Sherbrooke, Quebec, Canada, J1K 2R1

{shunichi,komatani,ogata,okuno}@kuis.kyoto-u.ac.jp {Jean-Marc.Valin,Jean.Rouat,Francois.Michaud}@USherbrooke.ca

***Abstract*— A humanoid robot under real-world environments usually hears mixtures of sounds, and thus three capabilities are essential for robot audition; sound source localization, separation, and recognition of separated sounds. We have adopted the missing feature theory (MFT) for automatic recognition of separated speech, and developed the robot audition system. A microphone array is used along with a real-time dedicated implementation of Geometric Source Separation (GSS) and a multi-channel post-filter that gives us a further reduction of interferences from other sources. The automatic speech recognition based on MFT recognizes separated sounds by generating missing feature masks automatically from the post-filtering step. The main advantage of this approach for humanoid robots resides in the fact that the ASR with a clean acoustic model can adapt the distortion of separated sound by consulting the post-filter feature masks. In this paper, we used the improved Julius as an MFT-based automatic speech recognizer (ASR). The Julius is a real-time large vocabulary continuous speech recognition (LVCSR) system. We performed the experiment to evaluate our robot audition system. In this experiment, the system recognizes a sentence, not an isolated word. We showed the improvement in the system performance through three simultaneous speech recognition on the humanoid *SIG2*.**

***Index Terms*— automatic missing feature mask generation, missing feature theory, continuous speech recognition, robot audition**

I. Introduction

Due to increasing demands for symbiosis of humans and robots, humanoid robots are increasingly expected to possess perceptual capabilities similar to humans. In particular, auditory function is essential for social interaction, because verbal communication is important for humans. Unfortunately, current speech recognition technology, which usually assumes a single sound source is present, does not suppose to be used in real-world environments on a robot. When confronted with a mixture of sounds, three main capabilities are essential for robot audition; sound source localization, separation, and recognition of the separated sounds. While the first two are often addressed, the last one has not been studied as much.

A conventional approach used in human-robot interaction is to use microphones near the speaker's mouth to collect only the desired speech. Kismet of MIT has a pair of microphones with pinae, but a human partner still uses a microphone close to the speaker's mouth [1]. A group communication robot, Robita of Waseda University, assumes that each human participant uses a headset microphone [2].

The improvement of noise-robustness in automatic speech recognition (ASR) has been studied, in particular, in the AURORA project [3]. In order to realize noise-robust speech recognition, multi-condition training (training on a mixture of clean speech and noises) has been studied [4], [5]. This is currently the most common method for car and telephony applications. Because an acoustic model obtained by multi-condition training reflects all expected noises in specific conditions, ASR's use of the acoustic model is effective as long as the noise is stationary. This assumption holds well for background noises in a car and on a telephone. However, multi-condition training may not be effective for robots, since they usually work under dynamically changing noisy environment. Under such conditions, the missing feature theory (MFT) is often used as an alternative method [6], [7].

In this paper, we adopt MFT to realize a robot audition which has a capability of recognizing continuous speech faster. In previous work [8], we developed the method of computing a missing feature mask only from the data available to the robot in a real environment, that is automatic missing feature mask generation. However, our ASR which was constructed by using the CASA Toolkit CTK [6] has two limitations as follows:

One is that the reported ASR has dealt with only isolated word recognition. We realize large vocabulary continuous speech recognition (LVSCR) of simultaneous speech signals by using automatic missing feature mask generation. ASR usually uses an acoustic model, a language model, and a dictionary. In the isolated word recognition, a language model is simple and does not include relationships between words. In LVCSR, a language model needs relationships between words, and is implemented as a grammar or N-gram. LVCSR depends on not only an acoustic model but a language model. However, the reported CTK based ASR does not support the

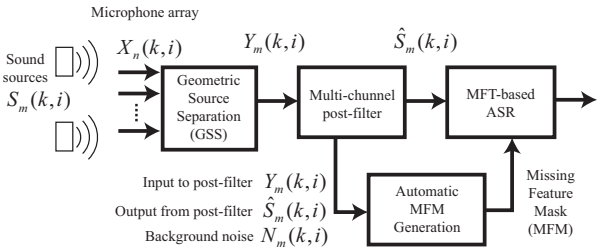


Fig. 1. Overview of the system

N-gram language model. The other is that the system was not sufficiently fast although real-time processing is essential for human robot interaction.

Therefore, we improve Julius which is a real-time LVCSR system to be able to have a function of automatic missing feature mask generation based on MFT.

The remainder of this paper is organized as follows. Section II provides an overview of the proposed recognition system. Section III details the multi-channel post-filter. Section IV detailed the basis of speech recognition using the missing feature theory. Section V explains how the missing feature mask is computed. Section VI provides an experiment for evaluating automatic missing feature mask generation, and Section VII provides a conclusion and future work.

II. System Overview

The speech recognition system, as shown in Figure 1, is composed of four parts:

1) Linear separation of the sources, implemented as a variant of the Geometric Source Separation (GSS) algorithm;
2) Multi-channel post-filtering of the separated output;
3) Computation of the missing feature mask from the post-filter output;
4) Speech recognition using the separated audio and the missing feature mask.

A sound source separation based on Independent Component Analysys (ICA) is fine, however, GSS is adopted since a processing speed of GSS is fast and GSS allows changes of number of sound sources. The microphone array used is composed of a number of omni-directional elements mounted on the robot. We assume that these sources are detected and localized by an algorithm such as [9] (our approach is not specific to any localization algorithm).

A. Source Separation

The source separation stage consists of a linear separation based on the Geometric Source Separation (GSS) approach proposed by Parra and Alvino [10]. It is modified so as to provide faster adaptation using stochastic gradient and shorter time frames estimations [11].

B. Multi-channel post-filter

The initial separation using GSS is followed by a multi-channel post-filter that is based on a generalization of beamformer post-filtering [12], [11] for multiple sources. This post-filter uses adaptive spectral estimation of background noise and interfering sources to enhance the signal produced during the initial separation. The main idea resides in the fact that, for each source of interest, the noise estimate is decomposed into stationary and transient components assumed to be due to leakage between the output channels of the initial separation stage.

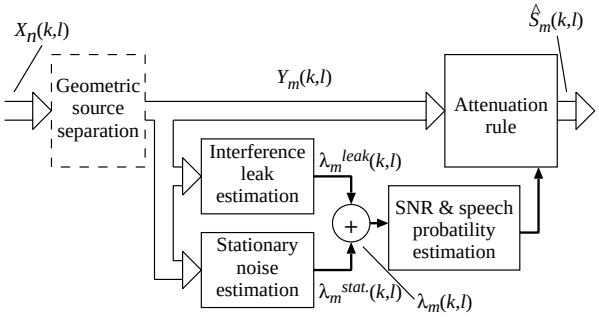


Fig. 2. Overview of the post-filter.
$X_n(k,\ell), n = 0 \ldots N-1$: Microphone inputs, $Y_m(k,\ell), m = 0 \ldots M-1$: Inputs to the post-filter, $\hat{S}_m(k,\ell) = G_m(k,\ell)Y_m(k,\ell),\ m = 0 \ldots M-1$: Post-filter outputs.

C. Missing Feature Mask Computation

The multi-channel post-filter is not only useful for reducing the amount of interference in the separated sounds. It also provides useful information concerning the amount of noise present at a certain time, at a particular frequency. Hence, we use the post-filter to estimate a missing feature mask that indicates how reliable each spectral feature is when performing recognition.

D. Recognition

For speech recognition, we used CTK, which is based on the missing feature theory. CTK does not yet support statistical language models, we use isolated word recognition only. However, in this paper, we used the improved Julius, which is detailed in Section IV-D. Though CTK does not support stochastic language models, Julius does.

III. Multi-channel post-filter

In order to enhance the output of the GSS algorithm, we derive a frequency-domain post-filter that is based on the optimal estimator originally proposed by Ephraim and Malah [13], [14]. Several approaches to microphone array post-filtering have been proposed in the past. Most of these post-filters address reduction of stationary background noise [15], [16]. Recently, a multi-channel post-filter taking into account non-stationary interferences was proposed by Cohen [12]. The novelty of our approach resides in the fact that, for a given channel output of the GSS, the transient components of the corrupting sources are assumed to be due to leakage from the other channels during the GSS process. Furthermore, for a given channel, the stationary and the transient components are combined into a single noise estimator used for noise suppression, as shown in Figure 2.

For this post-filter, we consider that all interferences (except the background noise) are localized (detected by the

localization algorithm) sources and we assume that the leakage between channels is constant. This leakage is due to reverberation, localization error, differences in microphone frequency responses, near-field effects, etc.

Section III-A describes the estimation of noise variances that are used to compute the weighting function G_m by which the output Y_m of the separation algorithm (LSS) is multiplied to generate a cleaned signal whose spectrum is denoted $\hat{S}_m$.

A. Noise estimation

The noise variance estimation $\lambda_m(k,\ell)$ is expressed as:

$$\lambda_m(k,\ell) = \lambda_m^{stat.}(k,\ell) + \lambda_m^{leak}(k,\ell) \quad (1)$$

where $\lambda_m^{stat.}(k,\ell)$ is the estimate of the stationary component of the noise for source m at frame ℓ for frequency k, and $\lambda_m^{leak}(k,\ell)$ is the estimate of source leakage.

We compute the stationary noise estimate $\lambda_m^{stat.}(k,\ell)$ using the Minima Controlled Recursive Average (MCRA) technique proposed by Cohen [17].

To estimate λ_m^{leak} we assume that the interference from other sources is reduced by a factor η (typically $-10\,\mathrm{dB} \leq \eta \leq -5\,\mathrm{dB}$) by the LSS. The leakage estimate is thus expressed as:

$$\lambda_m^{leak}(k,\ell) = \eta \sum_{i=0, i\neq m}^{M-1} Z_i(k,\ell) \quad (2)$$

where $Z_m(k,\ell)$ is the smoothed spectrum of the m^{th} source, $Y_m(k,\ell)$, and is recursively defined (with $\alpha_s = 0.7$) as:

$$Z_m(k,\ell) = \alpha_s Z_m(k,\ell-1) + (1-\alpha_s) Y_m(k,\ell) \quad (3)$$

B. Suppression rule in the presence of speech

We now derive the suppression rule under H_1, the hypothesis that speech is present. From here on, unless otherwise stated, the m index and the ℓ arguments are omitted for clarity and the equations are given for each m and for each ℓ.

The proposed noise suppression rule is based on minimum mean-square error (MMSE) estimation of the spectral amplitude in the loudness domain, $|X(k)|^{1/2}$. The choice of the loudness domain over the spectral amplitude [13] or log-spectral amplitude [14] is motivated by better results obtained using this technique, mostly when dealing with speech presence uncertainty (Section III-C).

The loudness-domain amplitude estimator is defined by:

$$\hat{A}(k) = \left(E\left[|S(k)|^{\alpha}\,|Y(k)|\right]\right)^{\frac{1}{\alpha}} = G_{H_1}(k)\,|Y(k)| \quad (4)$$

where $\alpha = 1/2$ for the loudness domain and $G_{H_1}(k)$ is the spectral gain assuming that speech is present.

The spectral gain for arbitrary α is derived from Equation 13 in [14]:

$$G_{H_1}(k) = \frac{\sqrt{\upsilon(k)}}{\gamma(k)} \left[\Gamma\left(1+\frac{\alpha}{2}\right) M\left(-\frac{\alpha}{2}; 1; -\upsilon(k)\right)\right]^{\frac{1}{\alpha}} \quad (5)$$

where $M(a;c;x)$ is the confluent hypergeometric function, $\gamma(k) \triangleq |Y(k)|^2/\lambda(k)$ and $\xi(k) \triangleq E\left[|S(k)|^2\right]/\lambda(k)$ are respectively the *a posteriori* SNR and the *a priori* SNR. We also have $\upsilon(k) \triangleq \gamma(k)\xi(k)/\left(\xi(k)+1\right)$ [13].

The *a priori* SNR $\xi(k)$ is estimated recursively as:

$$\begin{aligned}\hat{\xi}(k,l) &= \alpha_p G_{H_1}^2(k,\ell-1)\gamma(k,\ell-1) \\ &+ (1-\alpha_p)\max\left\{\gamma(k,\ell)-1, 0\right\}\end{aligned} \quad (6)$$

using the modifications proposed in [17] to take into account speech presence uncertainty.

C. Optimal gain modification under speech presence uncertainty

In order to take into account the probability of speech presence, we derive the estimator for the loudness domain:

$$\hat{A}(k) = \left(E\left[A^{\alpha}(k)|\,Y(k)\right]\right)^{\frac{1}{\alpha}} \quad (7)$$

Considering H_1, the hypothesis of speech presence for source m, and H_0, the hypothesis of speech absence, we obtain:

$$\begin{aligned}E[A^{\alpha}(k)|Y(k)] &= p(k)E\left[A^{\alpha}(k)|\,H_1, Y(k)\right] \\ &+ [1-p(k)]E[A^{\alpha}(k)|H_0, Y(k)]\end{aligned} \quad (8)$$

where $p(k)$ is the probability of speech at frequency k.

The optimally modified gain is thus given by:

$$G(k) = \left[p(k)G_{H_1}^{\alpha}(k) + (1-p(k))G_{min}^{\alpha}\right]^{\frac{1}{\alpha}} \quad (9)$$

where $G_{H_1}(k)$ is defined in (5), and G_{min} is the minimum gain allowed when speech is absent. Unlike the log-amplitude case, it is possible to set $G_{min} = 0$ without running into problems. For $\alpha = 1$, this leads to:

$$G(k) = p(k)G_{H_1}(k) \quad (10)$$

Setting $G_{min} = 0$ means that there is no arbitrary limit on attenuation. Therefore, when the signal is certain to be non-speech, the gain can tend toward zero. This is especially important when the interference is also speech since, unlike stationary noise, residual babble noise always results in musical noise.

The probability of speech presence is computed as:

$$p(k) = \left\{1 + \frac{\hat{q}(k)}{1-\hat{q}(k)}\left(1+\xi(k)\right)\exp\left(-\upsilon(k)\right)\right\}^{-1} \quad (11)$$

where $\hat{q}(k)$ is the *a priori* probability of speech presence for frequency k and is defined as:

$$\hat{q}(k) = 1 - P_{local}(k)P_{global}(k)P_{frame} \quad (12)$$

where $P_{local}(k)$, $P_{global}(k)$ and P_{frame} are defined in [17] and correspond respectively to a speech measurement on the current frame for a local frequency window, a larger frequency and for the whole frame.

IV. Continuous Speech Recognition Based on Missing Feature Theory

A. Feature for Missing Feature Theory

Since Mel-Frequency Cepstrum Coefficient (MFCC) is not appropriate for recognizing separated sounds from simultaneous speech signals, we use spectral features that are obtained to apply Inverse Discrete Cosine Transform (DCT) to MFCCs. The detailed flow of calculation is as follows:

1) 16 bit acoustic signals sampled by 16kHz are analyzed by FFT with 400 points of window and 160 frame shift to obtain spectrum..
2) Spectrogram is analyzed by Mel-scale filter bank to obtain Mel-scale spectrum of 24th order.
3) Mel-scale spectrum of 24th order is converted to log-energies.
4) The log Mel-scale spectrum is converted by DCT to the Cepstrum.
5) The Cepstrum is normalized by Cepstral Mean Subtraction.
6) The normalized Cepstrum of 24th order is converted to the spectral domain by Inverse DCT.
7) The features are differentiated in the time domain. Thus, we obtain 24 spectral features and their first-order differentiation.

The reason why we use Log-Mel-scale spectrum is to remove multiplicative noises caused by microphone distortion or transmission distortion. MFCC is obtained by removing the current components after the step of 5), but this removal does not allow inverse operations.

B. Missing Feature Masks

Automatic generation of missing feature masks needs information about which spectral parts of a separated sound are distorted. This kind of information may be obtained by a sound source separation system. We use the post-filter gains as reference data to generate a missing feature mask automatically. Since we use a feature vector of 48 spectral-related features, the missing feature mask is a vector of 48 corresponding features. Each element of a vector represents the reliability of each feature. The value may be binary (1, reliable, or 0, unreliable) or continuous between 0 and 1. In this paper, we used a binary missing feature mask.

C. Missing Feature Theory Based ASR

Missing Feature Theory based ASR is a Hidden Markov Model (HMM) based recognizer, which is commonly used by most ASRs. The difference is only in the decoding process. In conventional ASR systems, estimation of a path with maximum likelihood is based on state transition probabilities and output probability in HMM. In the case of missing feature based recognition, estimation of the output probability is different from conventional ASR systems.

Let $o(\boldsymbol{x}|S)$ be the output probability of feature vector $\boldsymbol{x}$ in state S. The output probability is defined by

$$o(\boldsymbol{x}|S) = \sum_{l=1}^{L} P(l|S) \exp\left\{ \sum_{i=1}^{N} M(i) \log f(x_i|l,S) \right\} \quad (13)$$

where L is the dimensionality of the Gaussian mixture, $M(i)$ is missing feature mask, $f(x_i|l,S)$ is the probability density function of Gaussian distribution, and N is the dimensionality of the feature.

D. Implementation

We used Julius which is a two-pass large vocabulary continuous speech recognition decoder [18]. Various HMM types are supported such as shared-state triphones and tied-mixture models. Stochastic language models are supported. In decoding, an ordered word bi-gram is used in the first pass, and a reverse ordered word tri-gram is used in the second pass.

We used modified Julius for MFT, which was called MFT-based Julius. The only modified part is related to output probability calculation. This output probability calculation is detailed in Section IV-C. In reality, output probability is calculated in log domain.

V. Automatic Missing Feature Mask Generation

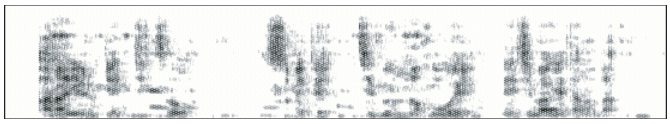
The missing feature mask is a matrix representing the reliability of each feature in the time-frequency plane. More specifically, this reliability is computed for each frame and for each Mel-frequency band. This reliability can be either a continuous value from 0 to 1, or a discrete value of 0 or 1. In this paper, discrete masks are used.

It is worth mentioning that computing the mask in the Mel-frequency bank domain means that it is not possible to use MFCC features, since the effect of the DCT cannot be applied to the missing feature mask.

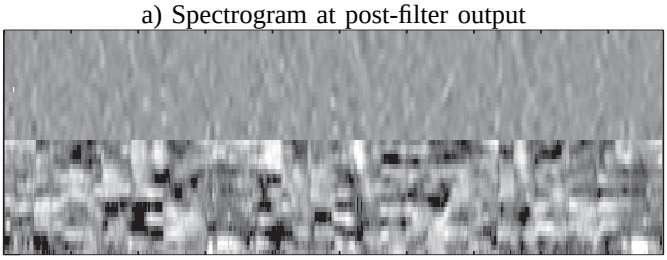
We compute the missing feature mask by comparing the input and the output of the multi-channel post-filter presented in Section III. For each Mel-frequency band, the feature is considered reliable if the ratio of the output energy over the input energy is greater than a threshold T. The reason for this choice is that it is assumed that the more noise present in a certain frequency band, the lower the post-filter gain will be for that band. The continuous missing feature mask $m(k,i)$ is thus computed as:

$$m(k,i) = \frac{S(k,i) + BN(k,i)}{Y(k,i)} \quad (14)$$

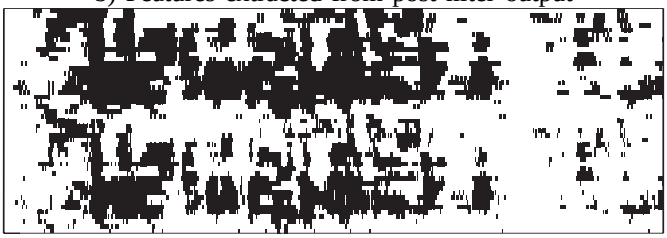
where $Y(k,i)$ and $S(k,i)$ are respectively the post-filter input and output energy for frame k, at Mel-frequency band $i = 1, \cdots, \frac{N}{2}$ and $BN(k,i)$ is the background noise estimate for that band. The main reason for including the noise estimate $BN(k,i)$ in the numerator of equation (14) is that it ensures that the missing feature mask equals 1 when no speech source



a) Spectrogram at post-filter output



b) Features extracted from post-filter output



c) Missing feature mask (white=reliable, black=unreliable)

Fig. 3. Missing feature mask computation. Figure b) shows static features in the lower half, and delta features in the upper half. Figure c) shows missing feature mask corresponding with Figure b).

is present. From the continuous mask $m(k,i)$, we derive a binary mask $M(k,i)$ as:

$$M(k,i) = \begin{cases} 1, & m(k,i) > T \\ 0, & otherwise \end{cases} \tag{15}$$

where $M(k,i)$ consists of the static mask ($i = 1, \cdots, \frac{N}{2}$) and the dynamic mask ($i = \frac{N}{2}+1, \cdots, N$), and T is an arbitrary threshold (we use $T = 0.3$). An example computation of the mask is shown in Figure 3.

The missing feature mask for delta-features is computed using the mask for the static features. The dynamic mask is computed as:

$$M(k,i) = M(k-2,j)M(k-1,j)M(k+1,j)M(k+2,j) \tag{16}$$

where $j = i - \frac{N}{2}$.

VI. Experiment

The proposed system is evaluated on the *SIG2* humanoid robot, on which an array of eight microphones is installed (Figure 4). In order to test the system, three voices are recorded simultaneously from loudspeakers placed two meters away from the robot. The room size is $5\,m \times 4\,m$, with a reverberation time of 0.3 – 0.4 sec. The room and *SIG2* are shown in Figure 5 Three loudspeakers are placed at every 30 degrees, 60 degrees, and 90 degrees. In this experiment the directions of arrival (DOAs) of the sound sources are given to GSS in advance. We use combinations of three different sentences selected from a set of 50 phonemically-balanced Japanese sentences. Examples of these phonemically-balanced Japanese sentences are shown in Table I.

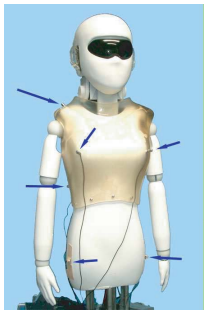


Fig. 4. SIG 2 robot with eight microphones (two are occluded).

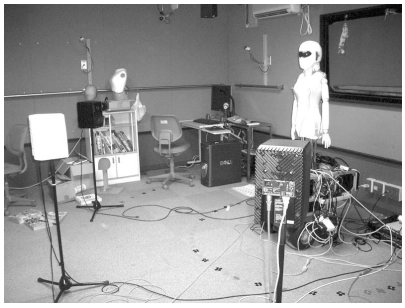


Fig. 5. The room where simultaneous speech signals are recorded on *SIG2*.

TABLE I

Examples of ATR phonemically-balanced sentences

Japanese	あらゆる/げんじつ/を/すべて/じぶん/の/ほう/へ/ねじまげ/た/の/だ/。
phoneme	arayuru/geNjitsu/o/subete/jibuNno/ho:/e/nejimage/ta/no/da
English	He distorted all the facts.
Japanese	いっ/しゅうかん/ばかり/ニューヨーク/を/しゅざい/し/た/。
phoneme	iq/shu:kaN/bakari/nyu:yo:ku/o/shuzai/shi/ta
English	I have been covering New York for about a week.

A. Acoustic Model for Speech Recognition

Even though many direction- and speaker-dependent acoustic models have been used in the past, we use only one triphone-based acoustic model for this system. The acoustic model is based on Hidden Markov Models (HMM) and is trained on clean speech. The acoustic model uses 3 states and 8 Gaussians per mixture. ASJ Continuous Speech Corpus (Japanese Newspaper Article Sentences) is used as a training data. The training data includes utterance sets from 306 speakers (153 male and female each.) Each utterance set consists of excerpts from the Mainichi Newspaper and ATR 503 phonemically-balanced sentences. The training data contains utterances of about 45,000 sentences as a whole with all speakers reading about 150 sentences each.

B. Language Model for Speech Recognition

We used stochastic language models which is trained on Japanese newspaper article sentences and ATR phonemically-balanced sentences (ATRPBS). Two language models are used for the experiment. One is trained on 50 ATR phonemically-balanced sentences, which has a vocabulary of about 400 words. Another consists of the ATRPBS language model and Mainichi Newspaper language model, and the Mainichi Newspaper language model is provided by Continuous Speech Recognition Consortium (CSRC). The composed language model has a vocabulary of about 20,000 words.

C. Results

We present the word corrects of recognition results obtained in various conditions. Our system recognized the speech signals from each direction by using each language model. The methods of recognizing three simultaneous speech signals are as follows:

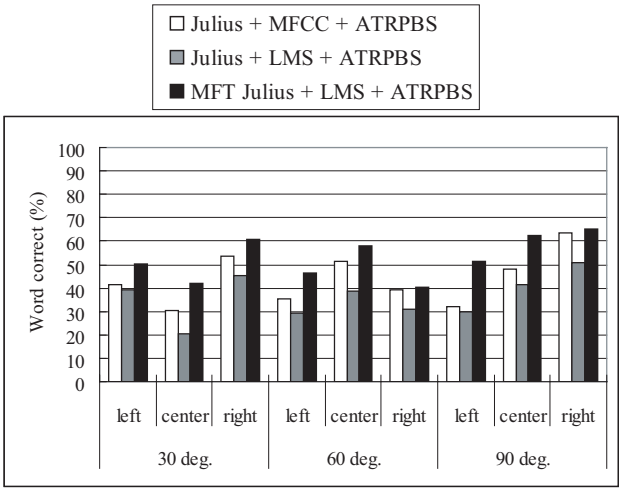


Fig. 6. Word corrects with using a language model of ATR phonemically-balanced sentence.

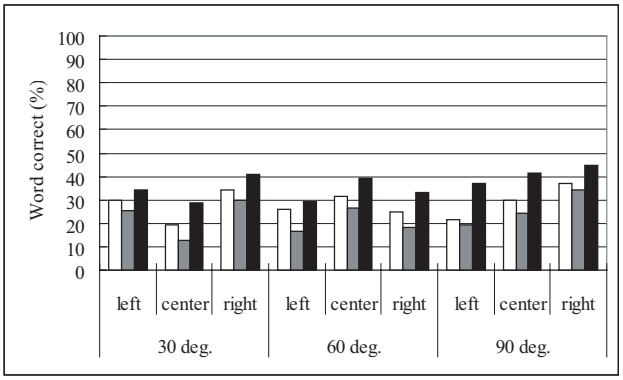


Fig. 7. Word corrects with using a language model of Mainichi Newspaper and ATR phonemically-balanced sentence.

1) Separated sounds are recognized by normal ASR, that is Julius with using MFCC.
2) Separated sounds are recognized by normal ASR, that is Julius with Log-Mel-Scale spectrum feature.
3) Separated sounds are recognized by MFT-based Julius with Log-Mel-Scale spectrum feature.

The word corrects are shown in Figure 6. This figure contains the word corrects in the case that a separated sound in each direction are recognized.

In all cases, the results of our presented method outperforms the result of normal ASR. In case of ATRPBS language model, an average of word corrects increases up to 53.0%. In case of Mainichi Newspaper and ATRPBS language model, an average of word corrects increases up to 36.5%. This demonstrates that the MFT-based approach is more efficient for speech recognition in robots than the usual approach based on MFCC.

VII. Conclusion and Future Work

In this paper, we presented the method of recognizing simultaneous speech signals by integrating sound source separation with microphone array and missing feature theory based automatic speech recognition. For our robot audition system to recognize not only a word but also a sentence faster, we used MFT-based Julius. The experiments for evaluation showed that our method worked better than normal ASR with using MFCC.

As for processing speed, MFT-based Julius takes 373 seconds to decode separated sounds of 315 seconds on Pentium 4 2.53 GHz Linux PC. In the case of normal Julius, it takes 314 seconds. Though MFT-based Julius is 84% as fast as normal Julius, it is much faster than CTK. It is not fixed times of computation that these methods require, however the methods approximately require times of computation in proportion to the length of processing speech. This platform is also easily available, so the processing speed is practical.

The future work includes improvement of the performance in recognizing simultaneous speech signals, and dealing with moving speakers. By using the improved Julius for MFT, our robot audition system had a capability of also recognizing sentences. Since recognition performance however is not sufficient in order to interact with humans, improvement is necessary. In real environments, speakers are not always stopping though we have dealt with only fixed speakers, or the robot is sometimes moving. When the robot is moving, it observes moving speakers relatively. That is why we should deal with moving speakers.

References

[1] C. Breazeal. Emotive qualities in robot speech. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-2001)*, pages 1389–1394. IEEE, 2001.

[2] Y. Matsusaka, T. Tojo, S. Kuota, K. Furukawa, D. Tamiya, K. Hayata, Y. Nakano, and T. Kobayashi. Multi-person conversation via multi-modal interface — a robot who communicates with multi-user. In *Proc. of Eurospeech-1999*, pages 1723–1726, 1999.

[3] D. Pearce. Developing the ETSI AURORA advanced distributed speech recognition front-end & what next. In *Proc. of Eurospeech-2001*. ESCA, 2001.

[4] R. P. Lippmann, E. A. Martin, and D. B. Paul. Multi-styletraining for robust isolated-word speech recognition. In *Proc. of ICASSP-87*, pages 705–708. IEEE, 1987.

[5] M. Blanchet, J. Boudy, and P. Lockwood. Environmentadaptation for speech recognition in noise. In *Proc. of EUSIPCO-92*, volume VI, pages 391–394, 1992.

[6] J. Barker, M. Cooke, and P. Green. Robust asr based on clean speech models: An evaluation of missing data techniques for connected digit recognition in noise. In *Proc. of Eurospeech-2001*, pages 213–216. ESCA, 2001.

[7] P. Renevey, R. Vetter, and J. Kraus. Robust speech recognition using missing feature theory and vector quantization. In *Proc. of 7th European Conference on Speech Communication Technology (Eurospeech-2001)*, volume 2, pages 1107–1110. ESCA, 2001.

[8] S. Yamamoto, J.-M. Valin, K. Nakadai, T. Ogata, and H. G. Okuno. Enhanced robot speech recognition based on microphone array source separation and missing feature theory. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2005)*. IEEE, 2005.

[9] J.-M. Valin, F. Michaud, B. Hadjou, and J. Rouat. Localization of simultaneous moving sound sources for mobile robot using a frequency-domain steered beamformer approach. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2004)*, pages 1033–1038. IEEE, 2004.

[10] L. C. Parra and C. V. Alvino. Geometric source separation: Mergin convolutive source separation with geometric beamforming. *IEEE Transactions on Speech and Audio Processing*, 10(6):352–362, 2002.

[11] J.-M. Valin, J. Rouat, and F. Michaud. Enhanced robot audition based on microphone array source separation with post-filter. In *Proceedings of IEEE International Conference on Robots and Systems (IROS 2004)*. IEEE, 2004.

[12] I. Cohen and B. Berdugo. Microphone array post-filtering for non-stationary noise suppression. In *ICASSP-2002*, pages 901–904, 2002.

[13] Y. Ephraim and D. Malah. Speech enhancement using minimum mean-square error short-time spectral amplitude estimator. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-32(6):1109–1121, 1984.

[14] Y. Ephraim and D. Malah. Speech enhancement using minimum mean-square error log-spectral amplitude estimator. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-33(2):443–445, 1985.

[15] R. Zelinski. A microphone array with adaptive post-filtering for noise reduction in reverberant rooms. In *ICASSP-1988*, volume 5, pages 2578–2581, 1988.

[16] I.A. McCowan and H. Bourlard. Microphone array post-filter for diffuse noise field. In *ICASSP-2002*, volume 1, pages 905–908, 2002.

[17] I. Cohen and B. Berdugo. Speech enhancement for non-stationary noise environments. *Signal Processing*, 81(2):2403–2418, 2001.

[18] T. Kawahara and A. Lee. Free software toolkit for japanese large vocabulary continuous speech recognition. In *International Conference on Spoken Language Processing (ICSLP)*, volume 4, pages 476–479, 2000.