2008 IEEE International Conference on
Robotics and Automation
Pasadena, CA, USA, May 19-23, 2008

A Robot Referee for Rock-Paper-Scissors Sound Games

Kazuhiro Nakadai, Shunichi Yamamoto, Hiroshi G. Okuno,
Hirofumi Nakajima, Yuji Hasegawa and Hiroshi Tsujino

Abstract— **This paper describes a robot referee for "rock-paper-scissors (RPS)" sound games; the robot decides the winner from a combination of rock, paper and scissors uttered by two or three people simultaneously without using any visual information. In this referee task, the robot has to cope with speech with low signal-to-noise ratio (SNR) due to a mixture of speeches, robot motor noises, and ambient noises. Our robot referee system, thus, consists of two subsystems – a real-time robot audition subsystem and a dialog subsystem focusing on RPS sound games. The robot audition subsystem can recognize simultaneous speeches by exploiting two key ideas;** *preprocessing* **consisting of sound source localization and separation with a microphone array, and** *system integration* **based on missing feature theory (MFT). Preprocessing improves the SNR of a target sound signal using geometric source separation with a multi-channel post-filter. MFT uses only** *reliable* **acoustic features in speech recognition and masks out unreliable parts caused by interfering sounds and preprocessing. MFT thus provides smooth integration between preprocessing and automatic speech recognition. The dialog subsystem is implemented as a system-initiative dialog system for multiple players based on deterministic finite automata. It first waits for a trigger command to start an RPS sound game, controls the dialog with players in the game, and finally decides the winner of the game. The referee system is constructed for Honda ASIMO with an 8-ch microphone array. In the case with two players, we attained a 70% task completion rate for the games on average.**

I. Introduction

In daily lives, simultaneous listening, or listening to several things at once, is mandatory for robots as well as human in some situations; for example, multi-party games, auctions, the stock exchange floor, and wholesale fish markets like "Tsukiji" in Japan. A robot should cope with simultaneously-uttered speech to realize rich and natural human-robot interaction. The robot will have a lot of opportunities to recognize a mixture of speeches to help people in real dialog situations as mentioned above. In addition, even in a normal multi-party dialog, conversation is usually alternate, but is sometimes interrupted by another speaker, which is called "barge-in". Therefore, a robot should recognize not only a speech from a single speaker but also simultaneous speeches uttered by multiple speakers.

Several dialog systems for a robot have been reported so far. Asoh *et al.* have developed Jijo-2 which was able to recognize and learn the surroundings through a dialog with people in an office environment [1]. Matsusaka *et al.* reported ROBISUKE supporting a multi-party dialog. It had a function to predict turn-taking by using users' eye movements and their speech [2]. Nakano *et al.* reported a multi-domain dialog system which enabled a robot to execute multiple tasks [3]. Mavridis and Roy proposed a grounded situation model to develop amodal representation and associated processes through verbal interaction with a human, and showed the effectiveness for the control of a robot arm [4]. Most of their robots made each speaker to wear a headset microphone to obtain high signal-to-noise ratio speech signals for automatic speech recognition (ASR). This solution also avoided a barge-in situation where user's utterance was contaminated by other utterances and thus was low in SNR. If a robot uses its own microphone, simultaneous speech recognition is critical in their applications. This means that research focusing on simultaneous speech recognition is necessary so that a robot can deal with real-world situations.

For this purpose, "Robot Audition" has been proposed as a new framework to recognize and understand real-world auditory scenes by using *robot's own microphones* in 2000 [5]. After several-year-studies to improve real-time auditory processing in the real world, robot audition is now considered as an essential function for understanding the surrounding auditory world such as human voices, music, and other environmental sounds. Actually, various robot audition systems were reported at robotics-related conferences [6], [7], [8], [9]. Thanks to recent rapid progress of signal processing level functions such as sound source localization, tracking and separation, we are ready to develop sophisticated applications. However, only a few preliminarily works have reported recognition of separated speech, and real-world applications of robot audition systems for human-robot interaction. Asano *et al.* developed a robot audition system with a simple dialog function in order to change TV channels and control TV volume by using voice commands in a normal office environment [8]. One weak point for their system is that one target speaker for speech recognition was assumed. We developed a robot audition system which is able to recognize simultaneous speech, but system evaluation was insufficient in terms of an application to human-robot interaction.

In this paper, we focus on a referee task for *rock-paper-scissors (RPS) sound games* which requires simultaneous speech recognition. RPS is a worldwide popular game, world championships are held every year (http://www.rpschamps.com/). In an RPS game, each player uses their hand to represent a rock, a paper, or scissors, while, in an RPS sound game, he or she says one of the three words

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

S. Yamamoto and H.G. Okuno are with Graduate School of Informatics, Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto, JAPAN {shunichi, okuno}@kuis.kyoto-u.ac.jp

978-1-4244-1647-9/08/$25.00 ©2008 IEEE.

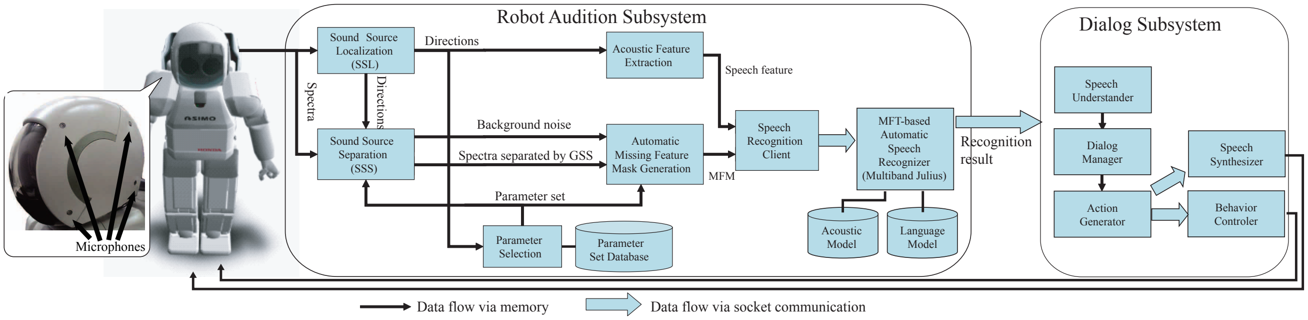


Fig. 1. A Referee system for Rock-Paper-Scissors Sound Games

instead of using gestures (see also Fig.4).

The referee system consists of two subsystems, that is, a robot audition subsystem and a dialog subsystem. The robot audition subsystem is able to recognize simultaneous speech by exploiting two key ideas; *Preprocessing* of ASR and *system integration* based on Missing Feature Theory (MFT)[10], [11]. Preprocessing of ASR such as sound source localization and separation using a robot-embedded microphone array to improve SNR before performing ASR. MFT integrates preprocessing with ASR by masking out unreliable features included in preprocessed signals and using only reliable features for recognition. The dialog subsystem is a system-initiative dialog system for multiple players based on Deterministic Finite Automata (DFA). It first waits for a trigger command to start an RPS sound game, controls the dialog with players in the game, and finally decides the winner of the game.

We implemented the referee system to Honda ASIMO with an 8-ch microphone array. The system was evaluated in terms of recognition of single and simultaneous speech when a robot noise was present, and task completion rates of RPS sound games to show the effectiveness of robot audition system more practically.

The rest of this paper is organized as follows: Section II explains system architecture of our referee system. Section III and IV describe the robot audition subsystem and the dialog subsystem, respectively. Section V evaluates our system. The last section concludes this paper.

II. Robot Referee System Architecture

An 8-ch microphone array embedded in Honda ASIMO is shown in the left side of Fig. 1. The positions of the microphones are bilaterally symmetric. This is because the longer the distance between microphones is, the better the performance of sound source separation of Geometric Source Separation (GSS). Fig. 1 depicts the architecture of the referee system. It includes two subsystems – a robot audition subsystem and a dialog subsystem. The robot audition subsystem consists of seven modules: Sound Source Localization (SSL), Sound Source Separation (SSS), Parameter Selection, Acoustic Feature Extraction, Automatic Missing Feature Mask Generation, Speech Recognition Client, and Missing Feature Theory based Automatic Speech Recognition (MFT-ASR). The six modules except for MFT-ASR are implemented as component blocks of *FlowDesigner* [12], a free data flow oriented development environment. The reason why MFT-ASR is treated separately is twofold; First, it needs a heavy CPU load in recognizing speech. Second, it uses a light-weighted data format in communication with the other modules, that is, it uses acoustic features and MFM for communication with the other modules, while the other modules use raw signal data for their communication. FlowDesigner and Multiband Julian may run separately on different CPUs, since they can communicate with each other via a network. Since the five modules communicate a large amount of data with each other, the reduction of communication traffic is critical in real-time processing. FlowDesigner provides the mechanism of sharing data on a shared memory between modules. It also provides the reusability of modules for rapid prototyping.

The dialog subsystem consists of five modules – Speech Understander, Dialog Manager, Action Generator, Speech Synthesizer, and Behavior Controller. The first three modules are implemented as one program, and the other two modules are connected with the program via a network.

III. Robot Audition Subsystem

This section describes our two key ideas for achieving robot audition – preprocessing and missing-feature-theory-based integration. Related work is also mentioned.

A. Preprocessing for Automatic Speech Recognition

A common approach to achieving noise-robust ASR is the use of an acoustic model for ASR trained with noise adaptation techniques [13]. However, it is suitable neither for dealing with unknown noise that is not included in training speech data nor for recognizing extremely noisy speech captured by a robot-embedded microphone. To improve the SNR of the input speech signals before performing ASR, SSL and SSS were introduced.

For SSL, we adopted a frequency-domain adaptive BF method, MUltiple SIgnal Classification (MUSIC)[14]. It has good performance in the real world because a sharp local peak corresponding to a sound source direction is obtained from the MUSIC spectrum in comparison with other BF methods. In our implementation, impulse responses which were measured at the intervals of 5 degrees were used to calculate a correlation matrix.

SSS consists of GSS and the multi-channel post-filter. GSS is originally proposed proposed by Parra *et al.* [15] as a kind of Blind Source Separation (BSS). It relaxes BSS's limitations such as a permutation and a scaling problem by introducing "geometric constraints" obtained from the locations of microphones and sound sources obtained from SSL. We modified the GSS so as to provide faster adaptation using stochastic gradient and shorter time frame estimation. It improved 10.3 dB in signal-to-noise ratio on average for separation of three simultaneous speech signals [16]. The multi-channel post-filter [16] is used to enhance the output of GSS. It is based on the optimal estimator originally proposed by Ephraim and Malah [17]. Their method is a kind of spectral subtraction [18], but it generates less distortion because it takes temporal and spectral continuities into account. This method is extended to enable support of multi-channel signals so that they can estimate both stationary and non-stationary noise, while most post-filters address the reduction of a type of noise, stationary background noise [19], [20]. For further reduction of spectral distortion caused by sound source separation, we add noise to multi-channel post-filtered speech, because an additive noise plays a roll to blur the distortions, that is, to avoid the fragmentation. We exploit covering a distortion in any frequency band by adding white noise, a kind of broad-band noise, to noise-suppressed speech signals.

B. Missing-Feature-Theory (MFT) Based Integration

Several robot audition systems with preprocessing and ASR have been reported so far [6], [8]. Those systems just combined preprocessing with ASR and focused on the improvement of SNR and real-time processing. Two critical issues remain; what kinds of preprocessing are required for ASR, and how does ASR use the characteristics of preprocessing besides using an acoustic model with multi-condition training. We exploited an interfacing scheme between preprocessing and ASR based on MFT.

MFT uses *missing feature masks (MFMs)* in a spectro-temporal map to improve ASR. Each MFM specifies whether a spectral value for a frequency bin at a specific time frame is reliable or not. Unreliable acoustic features caused by errors in preprocessing are masked out using MFMs, and only reliable ones are used for a likelihood calculation in the ASR decoder.

Most conventional ASR systems use *Mel-Frequency Cepstral Coefficient (MFCC)* as an acoustic feature, but noises and distortions which are concentrated in some areas in the spectro-temporal space are spread to all coefficients in MFCC. In general, Cepstrum based acoustic features like MFCC are not suitable for MFT-ASR. Acoustic Feature Extraction, therefore, calculates *Mel-Scale Log Spectrum* (MSLS) [21] as an acoustic feature.

Automatic MFM generation estimates MFMs by using information about the amount of noise present in a frequency band provided by the multi-channel post-filter. Since our acoustic feature vector consists of 48 spectral-related acoustic features, the missing feature mask is a vector of 48 corresponding features. Each element of a vector represents the reliability of each acoustic feature in binary format (1 for reliable, and 0 for unreliable).

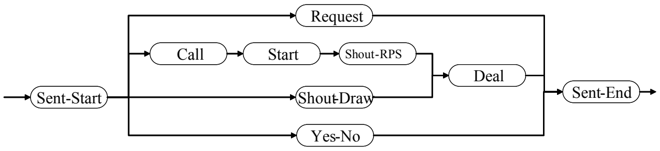


Fig. 2. Network Grammar Model

MFT-ASR outputs a sequence of phonemes from acoustic features of separated speech and the corresponding MFMs which are sent by Speech Recognition Client. For MFT-ASR, we used Multiband Julian [22], which is based on the Japanese real-time large vocabulary speech recognition engine Julian [23]. It supports various HMM types such as shared-state triphones and tied-mixture models. Network grammar is supported for a language model. It works as a standalone or client-server application. To run as a server, we modified the system to be able to communicate acoustic features and MFM via a network.

C. Parameter Selection

Parameter Selection selects an appropriate parameter set for a current state by using results of SSL. There are 13 parameters in the robot audition subsystem, which concern the performance of SSS and MFT-ASR. We optimized parameter values by using Genetic Algorithm because they are mutually-dependent. It was reported that the optimization improved 10 – 20 pts in a word correct rate of isolated word recognition for two and three simultaneous speeches [24].

IV. Dialog Subsystem

This section describes the dialog subsystem for RPS sound games. In an RPS sound game, each player says one of the three words instead of using gestures. The robot, then, decides the winner only by using information obtained from simultaneous speeches. Therefore, the referee system requires simultaneous speech recognition. The dialog subsystem is implemented as a system-initiative speech dialog system, since it assumes limited vocabulary in each dialog state. Generally, this kind of dialog system supports only one user, but our system allows multiple users to cope with simultaneous speeches.

A language model for MFT-ASR is defined as network grammar to support an RPS referee task shown in Fig. 2. "Request" means a trigger speech command such as "Let's start rock-paper-scissors" or "Restart". "Call", "Start", "Shout-RPS" are nodes to deal with in-play commands such as "rock-paper-scissors, Rock.", "rock-paper-scissors, Paper.", and "rock-paper-scissors, Scissors." "Shout-Draw" is also used for an in-play command after a drawn case occurs. "Yes-No" is a node for a yes-no reply.

Speech Understander generates four kinds of speech requests shown in Tab. Ia) from recognition results of separated speeches sent by the robot audition subsystem. Since multiple players usually speak at the same time in the RPS game, it is necessary for the dialog system to detect a set of recognition results for speeches uttered simultaneously. A speech recognition result is sent to Speech Understander

TABLE I

REQUEST, STATE, AND OUTPUT FOR DIALOG SUBSYSTEM

a) Dialog Requests

Request	Description
Start	a request when a game-start command is detected
Draw	a request when a draw occurs
Finish	a request when the winner was found
End	a request when a game-end command is detected

b) Dialog States

Status	Description
Wait	waiting for a start trigger command from a player
Play	waiting for a speech mixture of rock, paper, and/or scissors
Draw	**Play** after a draw.
Cont	waiting for a reply to a question if the players want to start the next game

c) Dialog Outputs

Output	Description
Call	Urge each player to say one of three words
Next	**Call** after a drawn case
Result	Inform the players of the winner with gestures and speech. For gestures, ASIMO turns its face to and points its hand to the winner by using a winner direction obtained from sound source localization. For speech, ASIMO answers the winner's choice like "the player who said rock is the winner."
End	Just finish the game without any gesture and speech

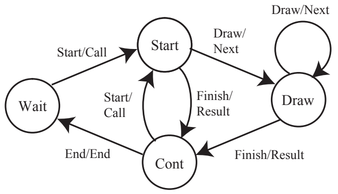


Fig. 3. Dialog State Transition

only when the speech end is detected. Thus, it detects such a recognition result set for simultaneous speeches based on speech end time information.

The detailed algorithm is as follows:

1) After it receives a speech recognition result, it waits for the next speech recognition result for a stand-by time period, that is, 1 second by default.
2) When the next result arrives within 1 second, it waits for the next stand-by time period. Because the possibility that simultaneous speech occurs is getting lower, the stand-by time period becomes shorter.

Dialog Manager controls dialog state transition according to the requests. Four dialog states are defined as Tab. Ib), and state transition based on DFA is shown in Fig. 3. When Dialog Manager receives a dialog request, controls dialog states, and produces dialog outputs defined as Tab. Ic).

Action Generator generates gesture commands for **Behavior Control**, and send speech text to **Speech Synthesizer** via a network. **Behavior Control** and **Speech Synthesizer** actually control ASIMO's behaviors.

V. EVALUATION

We evaluated the robot audition system in terms of the following two points:

1) Recognition performance of simultaneous speech,
2) Performance of referee for RPS sound games.

TABLE II

WORD ERROR RATES FOR 1 – 3 SIMULTANEOUS SPEECHES (%)

# of	interval	experimental conditions					
speakers	(deg)	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**
1	-	6.5	6.5	16.5	15.5	**10.0**	18.0
2	30	100	73.5	65.5	64.5	**27.0**	14.8
	60	100	71.3	51.8	58.5	**17.8**	14.5
	90	100	67.0	55.3	59.0	**17.3**	16.0
3	30	100	94.5	98.0	93.5	**72.5**	23.0
	60	100	94.5	88.0	70.0	**34.5**	15.0
	90	100	91.5	82.5	65.5	**18.5**	18.0

A. Recognition of Simultaneous Speech

We, first, evaluated the performance of the robot audition subsystem through isolated word recognition of one, two, and three simultaneous speeches. In this experiment, a 4 m × 5 m room with 0.3–0.4 seconds reverberation time (RT_{20}) was used.

Three kinds of test data sets, that is, a single speech dataset (T1), a two-simultaneous-speech dataset (T2), and a three-simultaneous-speech dataset (T3), were recorded with the 8-ch microphone array in the room by using loudspeakers (Genelec 1029A) when ASIMO was turned on. The distance between each loudspeaker and the center of the robot was 1.5 m. In case of T1, a loudspeaker was set to the front (center) direction of ASIMO. For T2, one loudspeaker was fixed to the front direction, and the direction of the other loudspeaker was selected from 30, 60, and 90 degrees. For T3, one loudspeaker was fixed to the front direction of the robot. The locations of the left and right loudspeakers from the center loudspeaker varied from ±30 to ±90 degrees at intervals of 30 degrees. As a speech dataset, 5 male and 5 female ATR phonemically-balanced word-sets were used. Each test dataset consists of 200 combinations of words which were randomly selected from the ATR datasets. Note that a word combination can include the same words uttered by the same speakers.

By using the test data, the system performed simultaneous speech recognition with the following six conditions:

C1 The input from the left-front microphone was used without any processing and MFT using **a clean acoustic model**.

C2 Only GSS was used as preprocessing. The **clean acoustic model** was used.

C3 GSS and Post-filter were used as preprocessing, but the MFT function was not. The clean acoustic model was used.

C4 The same condition as (3) was used except for the use of the **MFT function with automatically generated MFM**.

C5 The acoustic model trained with *white-noise-added speech* (**WNA acoustic model**) was used. Except for this, the condition was the same as (4).

C6 The same condition was used except for the use of *a priori* **MFM** created by clean speech. Since this mask is *ideal*, we consider its result as the potential upper limit of our system.

The **clean acoustic model** was trained with Japanese

TABLE III

RESULT OF TWO-SPEAKER TASK (ROCK-PAPER-SCISSORS GAMES)

speaker		judgment		task completion	
Mr.A (deg.)	Mr.B (deg.)	#success/ #judgments	success rate (%)	#success/ #tasks	completion rate(%)
0	-30	11/15	73.3	6/10	60
0	-60	9/11	81.8	8/10	80
0	-90	9/12	75.0	8/10	80
0	-120	11/13	84.6	8/10	80
0	-150	6/14	42.9	2/10	20
0	-180	8/11	72.7	7/10	70
30	-30	17/18	94.4	9/10	90
60	-60	7/12	58.3	5/10	50
90	-90	12/16	75.0	6/10	60
120	-120	15/17	88.2	8/10	80
150	-150	17/18	94.4	9/10	90
average		112/157	71.3	76/110	69.1

speaker		judgment		task completion	
Mr.C (deg.)	Mr.D (deg.)	#success/ #judgments	success rate (%)	#success/ #tasks	completion rate(%)
0	-30	12/16	75.0	6/10	60
0	-60	10/11	90.9	9/10	90
0	-90	9/13	69.2	6/10	60
0	-120	17/21	90.0	6/10	60
0	-150	10/14	71.4	6/10	60
0	-180	9/11	81.8	8/10	80
30	-30	14/19	73.7	5/10	50
60	-60	11/13	84.6	8/10	80
90	-90	8/11	72.7	7/10	70
120	-120	11/12	91.7	9/10	90
150	-150	10/11	90.9	9/10	90
average		121/152	79.6	79/110	71.8
total		233/309	75.4	155/220	70.5

Newspaper Article Sentences (JNAS) corpus which includes 60-hour speech data spoken by 306 male and female speakers. Thus, the test was speaker- and word-open. The **WNA acoustic model** was trained with the JNAS speech data to which white noise was added by 40 dB of peak power. Each of these acoustic models were trained as 3-state and 4-mixture triphone HMM, because 4-mixture HMM had the best performance among 1, 2, 4, 8, and 16-mixture HMMs.

The word error rates for the front speaker in T1 – T3 are summarized in Tab. II. MFT-ASR with Automatic MFM Generation (**C4, C5**) outperformed the normal ASR (**C1–C3**). The **WNA acoustic model**(**C5**) performed the best for MFT-ASR. We think that this is because our added white noise smoothed distortion caused by preprocessing, and the system was able to regard white-noise-added preprocessed speech as white-noise-added clean speech. Since the **WNA acoustic model** does not require prior training, it is the most appropriate acoustic model for robot audition. Performance at the 30-degree interval was poor in particular for T3, because there were two interfering sources for the front speech. The fact that *A priori* mask showed quite high performance may suggest many possibilities that still remain to improve the algorithms of MFM generation.

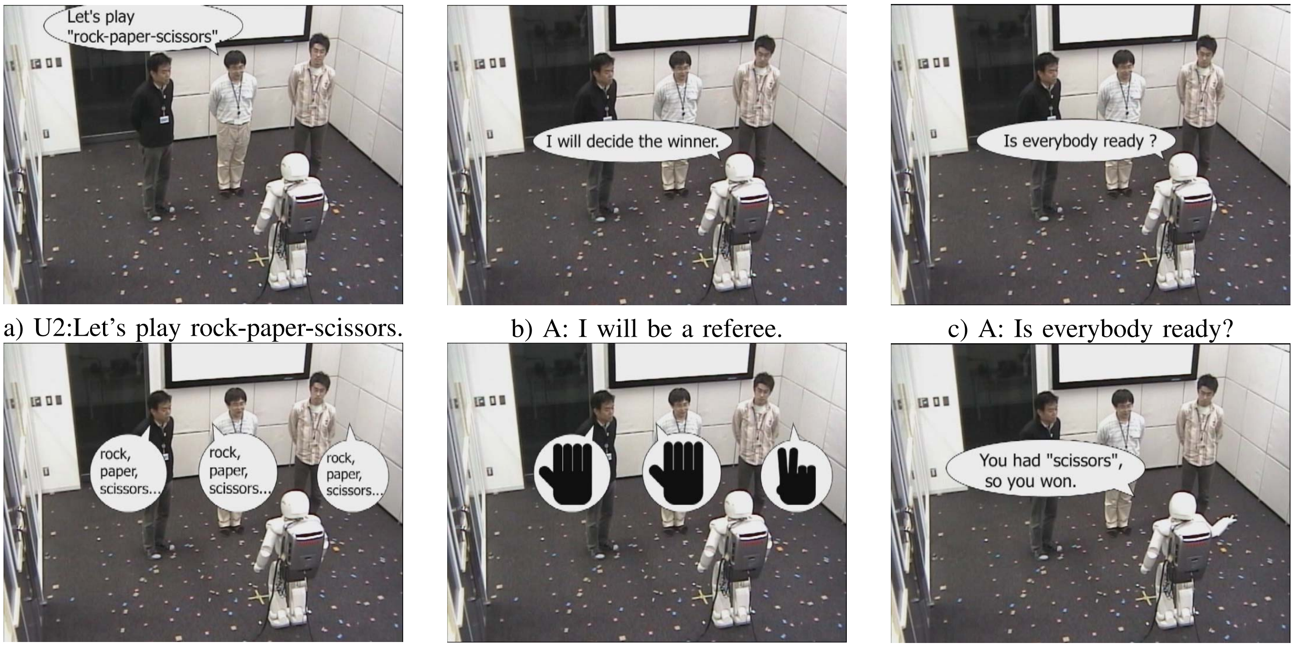
B. Performance of a Referee for Rock-Paper-Scissors Games

We evaluated the performance of a referee when the number of players is two, i.e., a two simultaneous speech case. The same room was used for the previous experiments. Two pairs of male players, that is, Mr.A & B and Mr.C & D attended our experiments. For each pair, 11 sets of RPS sound games were conducted. For the first 6 sets, Mr. A or C was fixed to stand at the front direction, and the others were changed from -30 deg. to -180 deg. at 30 degree intervals. For the last 5 sets, players stood from ± 30 deg. to ± 150 deg. at 30 degree intervals. Each set consists of 10 RPS sound games. ASIMO was located at the center of the room, and every player stood 1.5 m away from ASIMO. Fig. 4 shows a sequence of snapshots for a RPS sound game with three players. In this case, a unique winner existed. Needless to say, the system was able to handle drawn cases. For evaluation, we used two measures – a judgment success rate and a task completion rate. The judgment success rate J is defined as "# of success judgments / # of judgment opportunities". The task completion rate T is defined as "# of success winner judgment / # of RPS sound games". They are different when a drawn case is included in RPS sound games. J is used for performance evaluation of speech recognition rather than that of RPS task. T is opposite.

Tab. III shows the results of this experiment. On average, a judgment success rate is 75%, and a task completion rate is 70%. J is almost the same as the performance of the two-speaker-case in Tab. II in spite of small sized vocabulary (three words). We found two reasons for this. One is that lengths of three words were not long enough for ASR. Actually, they are "gu:","choki", and "pa:" according to Japanese pronunciation. The other is that the system often failed in finding simultaneous speech parts. In RPS sound games, the system had to perform voice activity detection for simultaneous speech, while the system was able to assume that each input included simultaneous speech in Tab. II. Mr.A & B are experts of this system because they are developers, while Mr.C & D had no experience of a speech dialog system and automatic speech recognition. However, the system performance of Mr.C & D was better. This means that the performance is independent from who players are. We could not find any tendency on the difference between the results of the 11 sets. Thus, the performance is irrelevant to the locations of players. This shows that ASIMO was able to decide the winner properly even when players are out of sight, while the players have to be in sight in a referee task for a normal RPS game.

VI. CONCLUSION

We reported a referee robot for rock-paper-scissors sound games. This task requires simultaneous speech recognition which is essential to cope with real-world auditory scenes. The referee system consists of a robot audition subsystem that recognizes noisy speech contaminated by simultaneous speech, and dialog subsystem focusing on RPS sound games. The referee system was implemented with Honda ASIMO, and we attained over a 70% task completion rate even when players have no experience with this kind of system. Thus, we can say that robot audition is practically effective. In this paper, we focused on a referee task for RPS sound



a) U2:Let's play rock-paper-scissors. b) A: I will be a referee. c) A: Is everybody ready?

d) U1-U3: rock-paper-scissors... e) U1: paper, U2: paper, U3: scissors f) A: U3 won.

Fig. 4. Snapshots of rock-paper-scissors game (A: ASIMO, U1:left user, U2:center user, U3: right user)

games. However, our approach would be effective for robots to construct other dialog tasks which requires simultaneous speech recognition and noisy speech recognition such as auctions, other multi-party games, and so on.

VII. ACKNOWLEDGMENTS

The authors gratefully thank Dr. Jean-Marc Valin, CSIRO.

REFERENCES

[1] H. Asoh *et al.*, "Socially embedded learning of the office-conversant mobile robot *jijo-2*," in *Proc. of 15th Int'l Joint Conf. on Artificial Intelligence (IJCAI-97)*, vol. 1. AAAI, 1997, pp. 880–885.

[2] Y. Matsusaka *et al.*, "Multi-person conversation via multi-modal interface — a robot who communicates with multi-user," *Proc. of Eurospeech-1999*, 1999, pp. 1723–1726.

[3] M. Nakano *et al.*, "A two-layer model for behavior and dialogue planning in conv ersational service robots," *Proc. of IEEE/RSJ Int'l Conf. on Intelligent Robot s and Systems (IROS-2005)*, 2005, pp. 1542–1547.

[4] N. Mavridis and D. Roy, "Grounded situation models for robots: Where words and percepts meet," *Proc. of IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS 2006)*. IEEE, 2006, pp. 4690–4697.

[5] K. Nakadai *et al.*, "Active audition for humanoid," *Proc. of 17th National Conf. on Artificial Intelligence (AAAI-2000)*. AAAI, 2000, pp. 832–839.

[6] 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, October 2004.

[7] J.-M. Valin *et al.*, "Enhanced robot audition based on microphone array source separation with post-filter," *Proc. of IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS 2004)*. IEEE, 2004, pp. 2123–2128.

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

[9] S. Yamamoto *et al.*, "Improving speech recognition of simultaneous speech signals by parameter optimization with genetic algorithm," *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA 2006)*. IEEE, 2006.

[10] J. Barker *et al.*, "Robust ASR based on clean speech models: An evaluation of missing data techniques for connected digit recognition in noise," *Proc. of Eurospeech-2001*. ESCA, 2001, pp. 213–216.

[11] M. Cooke *et al.*, "Robust automatic speech recognition with missing and unreliable acoustic data," *Speech Communication*, vol. 34, no. 3, pp. 267–285, May 2000.

[12] C. Côté *et al.*, C. Raievsky, M. Lemay, and V. Tran, *Proc. of IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS 2004)*. IEEE, 2004, pp. 1820–1825.

[13] R. P. Lippmann *et al.*, "Multi-styletraining for robust isolated-word speech recognition," *Proc. of ICASSP-87*. IEEE, 1987, pp. 705–708.

[14] F. Asano *et al.*, "Sound source localization and signal separation for office robot "Jijo-2"," *Proc. of IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI-99)*, 1999, pp. 243–248.

[15] L. C. Parra and C. V. Alvino, "Geometric source separation: Mergin convolutive source separation with geometric beamforming," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 6, pp. 352–362, 2002.

[16] S. Yamamoto *et al.*, "Enhanced robot speech recognition based on microphone array source separation and missing feature theory," *Proc. of IEEE Int'l Conf. on Robotics and Automation (ICRA 2005)*. IEEE, 2005, pp. 1489–1494.

[17] 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*, vol. ASSP-32, no. 6, pp. 1109–1121, 1984.

[18] S. F. Boll, "A spectral subtraction algorithm for suppression of acoustic noise in speech," *Proc. of 1979 Int'l Conf. on Acoustics, Speech, and Signal Processing (ICASSP-79)*. IEEE, 1979, pp. 200–203.

[19] R. Zelinski, "A microphone array with adaptive post-filtering for noise reduction in reverberant rooms," *ICASSP-1988*, vol. 5, 1988, pp. 2578–2581.

[20] I. McCowan and H. Bourlard, "Microphone array post-filter for diffuse noise field," *ICASSP-2002*, vol. 1, 2002, pp. 905–908.

[21] Y. Nishimura *et al.*, "Noise-robust speech recognition using multi-band spectral features," *Proc. of 148th Acous. Soc. of America Meetings*, 1aSC7, 2004.

[22] Multiband Julius, "http://www.furui.cs.titech.ac.jp/mband_julius/."

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

[24] S. Yamamoto *et al.*, "Real-time robot audition system that recognizes simultaneous speech in the real world," *Proc. of IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS 2006)*. IEEE, 2006, pp. 5333–5338.