MEETING RECOGNITION WITH ASYNCHRONOUS DISTRIBUTED MICROPHONE ARRAY USING BLOCK-WISE REFINEMENT OF MASK-BASED MVDR BEAMFORMER

Shoko Araki[1], *Nobutaka Ono*[2*], *Keisuke Kinoshita*[1], *Marc Delcroix*[1]

[1] NTT Communication Science Laboratories, NTT Corporation,
2-4, Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237, Japan
[2] Faculty of System Design, Tokyo Metropolitan University
6-6, Asahigaoka, Hino-shi, Tokyo 191-0065, Japan

ABSTRACT

This paper addresses a front-end system for speech recognition of spontaneous conversational speech signals that are recorded with asynchronous distributed microphones such as smartphones. In our previous work, we proposed combining blind synchronization and a state-of-the-art microphone array speech enhancement technique, e.g., a time-frequency mask based minimum variance distortionless response (MVDR) beamformer. This approach has provided reasonably high recognition performance even if we use asynchronous microphones. However, because the previous speech enhancement method was applied in a full-batch mode, it has been difficult to track speaker position movement in a real meeting conversation. To make it possible to handle the speaker movement, this paper describes our attempt to refine the mask-based MVDR beamformer in a block-wise manner, and reports that such a refinement reduces the word error rate from 31.4% to 28.8% for real meeting recordings.

Index Terms— Meeting recognition, distributed microphones, blind synchronization, mask-based MVDR beamformer, block-batch

1. INTRODUCTION

Recently, automatic speech recognition (ASR) of spontaneous speech spoken by a single speaker has achieved a high level of accuracy [1, 2], and the performance of the distant speech recognition (DSR) of a single speaker has also greatly improved (e.g., [3–5]). Some DSR techniques for a single speaker have already reached a level of commercial use, e.g., home assistance products. However, despite much research over the years [6–17], ASR of multi-speaker conversational speech still remains a difficult task especially when we use distant microphone(s). In such a conversation scenario, we have to consider overlapping speech, in addition to noise and reverberation. To handle these issues, there have been a number of studies that deal with acoustic interferences in multi-speaker conversations by using a standard microphone array equipped with several synchronized microphones [9, 15–22]. By employing such a microphone array, we can realize powerful beamforming and improve ASR performance [9, 15–18].

However, it is sometimes difficult to obtain synchronous multichannel recordings, because it requires all the microphones to be connected to the same analog-to-digital converter, which may be costly and impractical in many applications. In contrast, it is easy to obtain asynchronous multichannel recordings, due to the widespread availability of voice recording devices including smartphones. However, it then becomes difficult to apply beamforming approaches directly to such an asynchronous microphone array, because even a small synchronization error can severely degrade performance.

To handle the asynchronous recordings, we have proposed employing blind synchronization before multichannel blind speech enhancement and confirmed that it improved the ASR accuracy [23]. In [23], we apply a speech enhancement method in a full-batch mode, i.e., we process an entire meeting to estimate spatial correlation matrices for each speaker, and use these matrices to compute beamformer coefficients, which were constant in each session. However, our analysis of real meetings confirmed that, even for sitting meetings, there were large fluctuations in speaker positions in our case of up to 30 degrees (detailed in Sec. 2.2 with Fig. 2). Naturally, a constant beamformer may not be optimal for dealing with such variations.

In order to refine the performance by tracking the speaker position movements, in this paper we describe our attempt to employ a block-online speech enhancement approach [16]. However, the straightforward application of a block-online approach causes a permutation problem between time blocks. To avoid this block-wise permutation problem, our proposed approach first calculates the parameters for obtaining the beamformer coefficients in a full-batch mode, and then refines the parameters and the beamformer coefficients at each time block. We should note that our objective is not to realize an online algorithm, but to obtain high recognition performance with off-line processing. We will show that the proposed block-wise refinement of the beamformer coefficients successfully improves recognition performance for real meeting conversations recorded with asynchronous microphones.

The rest of this paper is organized as follows. Section 2 describes the task of this paper, and Sec. 3 details our proposed approach, which consists of blind synchronization and speech enhancement techniques. Section 4 reports the experimental results, and Sec. 5 concludes this paper.

2. PROBLEM DESCRIPTION

2.1. Meeting scenario

The scenario dealt with in this paper is conversation sessions of four to six speakers in a noisy room (Fig. 1). The length of each session was around 15 to 20 minutes. We recorded real spontaneous conversations using four stereo microphones on smartphones (SP) on an oval table (see Fig. 1). Here, although the stereo microphones on each smartphone were synchronized, the four SPs were asyn-

*This work was undertaken when N. Ono was with National Institute of Informatics.

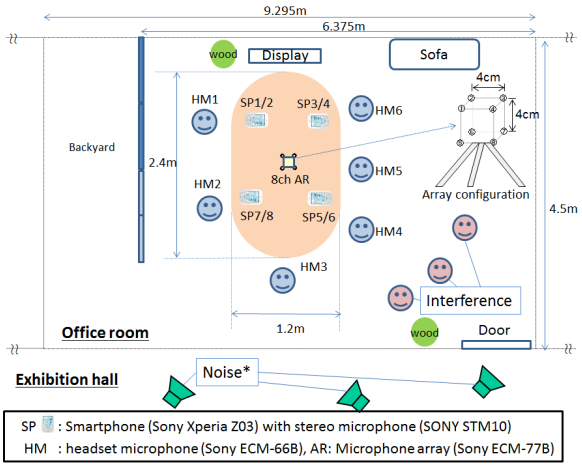

Fig. 1. Schematic diagram of the meeting room and the position of the microphones.

chronous, and therefore there were offset time and sampling frequency mismatches. These eight SP microphones are considered to be asynchronous distributed microphones in this paper.

The objective of this paper is speech recognition of each speaker utterance in real conversations recorded with the asynchronous distributed microphones on smartphones ("SP microphones"). In particular, we aim for a method that is robust to speaker movements.

2.2. Analysis of speaker movements in real meetings

To check the speaker movement during sessions, we looked at the estimated directions of arrival (DOAs) and diarization of speakers, that were estimated with 8-ch synchronous microphone array (AR in Fig. 1) observations and a probabilistic spatial dictionary-based method [24]. Figure 2 shows an example of estimated speaker movements in a session. From these estimated DOAs of speakers, we can see that the speakers moved up to 30 degrees during sessions despite them sitting on chairs without casters. This suggests that building algorithms that are able to track speaker positions may improve speech enhancement performance.

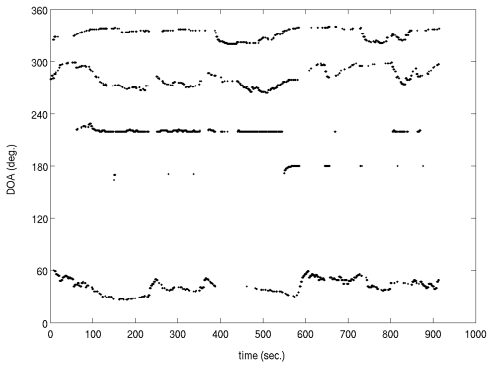

Fig. 2. Example of estimated speaker directions in a conversational session. Here, DOA=0° and 270° correspond to the display and the seat with HM5 directions (see Fig. 1), respectively.

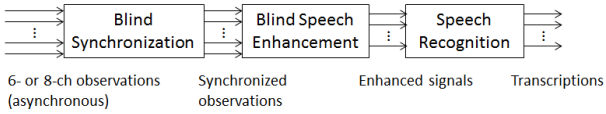

Fig. 3. Processing flow of the proposed meeting recognition system.

3. PROPOSED METHOD

Figure 3 shows the processing flow of the proposed method. We first synchronize the SP microphone recordings by using a blind synchronization technique. We then apply speech enhancement to the synchronized observations. The refinement of this enhancement step is the main contribution of this paper. Finally, the enhanced speech signals are passed to the ASR system to obtain the final transcriptions. This section describes the synchronization and enhancement steps, and the ASR step will be detailed in Sec. 4.2.

3.1. Blind synchronization

In this work, we record conversations with four smartphones as described in Sec. 2. Because they do not communicate with each other and each device records sound independently, they work as asynchronous distributed microphones. In this case, the recording can start at a different time on each device and the sampling frequency is not completely identical even though they have the same nominal sampling frequency, Therefore, if we assume that the sampling frequency of each device is time-invariant, the asynchronous recording is characterized by two parameters: the differences in recording start and sampling frequency.

In our case, it is expressed as follows.

$$x_0[t] = x_0\left(\frac{t}{f_s}\right), \quad (1)$$

$$x_{2S-1}[t] = x_{2S-1}\left(\frac{t}{(1+\epsilon_S)f_s} + T_S\right), \quad (2)$$

$$x_{2S}[t] = x_{2S}\left(\frac{t}{(1+\epsilon_S)f_s} + T_S\right), \quad (3)$$

where $x_0(t)$, $x_{2S-1}(t)$ and $x_{2S}(t)$ $(S = 1, \cdots, 4)$ denote *continuous* time domain observation by the reference microphone, the left and right channel observations of the S-th smartphone, and $x_0[t]$, $x_{2S-1}[t]$ and $x_{2S}[t]$ denote their *discretized* signals, respectively. Note that t is a discrete time variable in the equations above. We utilize ch. 1 of the microphone array (AR in Fig. 1) as a reference microphone $x_0(t)$ for synchronization and f_s is its sampling frequency. T_S and ϵ_S are the two parameters that represent the offset time and the sampling frequency mismatch of the microphones on S-th smartphone, respectively. Note that we used the same T_S and ϵ_S for stereo observations because they were synchronized with each other. It should also be noted that we can use the same T_S for stereo observations on a smartphone because the offset time T_S does not represent the time difference of arrival between stereo observations but the difference of the recording start time (DST). As we cannot estimate the absolute DST, we use cross-correlation between $x_0[t]$ and $x_{2S-1}[t] + x_{2S}[t]$ for estimating T_S as detailed in the next paragraph.

We synchronize the stereo observations $[x_{2S-1}[t], x_{2S}[t]]^T$ with the reference observation $x_0[t]$ by applying the blind synchronization technique proposed in [25, 26] with a small modification [23]. The offset time T_S is simply estimated by finding the peak of the

cross-correlation between $x_0[t]$ and $x_{2S-1}[t] + x_{2S}[t]$ and it is compensated. Then, the effect of the sampling frequency mismatch ϵ_s is compensated by a linear phase shift in the short-time Fourier transform (STFT) domain. This is derived by the approximation that the time-varying time difference between channels caused by the sampling frequency mismatch is constant within a time frame. The sampling frequency mismatch ϵ_S is estimated by maximizing the likelihood of the model where the compensated STFT representations follows a time-invariant multivariate Gaussian distribution. The signal processing is detailed in [25, 26].

3.2. Blind speech enhancement

In a real meeting, the precise positions of speakers and distributed microphones cannot be known in advance. Moreover, it cannot be guaranteed that the results of blind synchronization will be consistent with the positions of the speakers and microphones because the synchronization manipulation may change the inter-channel phase information. Therefore, a speech enhancement method has to work in a blind manner. As such a blind speech enhancement technique, this paper employs a time-frequency mask-based MVDR beamformer, which is the state-of-the-art enhancement approach with synchronized microphones [16, 22].

3.2.1. Mask-based MVDR beamformer

Let $\hat{\mathbf{x}}(f,\tau) = [\hat{x}_1(f,\tau), \cdots, \hat{x}_m(f,\tau), \cdots, \hat{x}_M(f,\tau)]^T$ be the STFT of the synchronized SP microphone observations, where f and τ are the indices of frequency and time frame, and $M(=2S)$ is the number of microphones. The time-frequency mask based MVDR beamformer for speaker k is given by

$$\mathbf{w}_k(f) = \frac{\mathbf{R}^{-1}_{\hat{\mathbf{x}}}(f)\mathbf{h}_k(f,\tau)}{\mathbf{h}^H_k(f,\tau)\mathbf{R}^{-1}_{\hat{\mathbf{x}}}(f)\mathbf{h}_k(f)}, \tag{4}$$

where $\mathbf{R}_{\hat{\mathbf{x}}}(f) = \sum_\tau \hat{\mathbf{x}}(f,\tau)\hat{\mathbf{x}}^H(f,\tau)$ and the steering vector $\mathbf{h}_k(f)$ of the MVDR beamformer of speaker k is estimated by using time-frequency masks. $\cdot^H$ denotes the conjugate transpose of a vector. By using these coefficients, the enhanced signal for speaker k is obtained by

$$y_k(f,\tau) = \mathbf{w}^H_k(f)\hat{\mathbf{x}}(f,\tau). \tag{5}$$

In (4), the steering vector $\mathbf{h}_k(f)$ should be estimated. To do so, first, a time-frequency mask $M_k(f,\tau)$ for extracting each speaker k is estimated by using a complex Gaussian mixture model (CGMM) [16, 22], which will be detailed in Sec.3.2.2. Then, the steering vector $\mathbf{h}_k(f)$ is calculated by using the estimated time-frequency masks $M_k(f,\tau)$,

$$\mathbf{h}_k(f) = \mathbf{R}_{-k}(f)\mathbf{e}_k(f), \tag{6}$$

where $\mathbf{e}_k(f)$ is an eigenvector corresponding to the largest generalized eigenvalue of the matrix pencil $(\mathbf{R}_k(f), \mathbf{R}_{-k}(f))$,

$$\mathbf{R}_{-k}(f) = \frac{1}{\sum_\tau 1 - M_k(f,\tau)} \sum_\tau (1 - M_k(f,\tau))\hat{\mathbf{x}}(f,\tau)\hat{\mathbf{x}}^H(f,\tau),$$

$$\mathbf{R}_k(f) = \frac{1}{\sum_\tau M_k(f,\tau)} \sum_\tau M_k(f,\tau)\hat{\mathbf{x}}(f,\tau)\hat{\mathbf{x}}^H(f,\tau).$$

Table 1. Conversation datasets

	Office-Exh.	Office	Sound-proof
T_{60}	500 msec.	350 msec.	120 msec.
SNR	3-15 dB	15-20 dB	20-25 dB
#Spkr/ses.	4-6	4	4
Train.	40 sessions	14 sessions	30 sessions
Dev.	8 sessions		
Eval.	8 sessions		

3.2.2. Mask estimation in full-batch or block-online modes

In CGMM-based mask estimation, we assume that the observation vector $\hat{\mathbf{x}}$ follows a CGMM [27]:

$$p(\hat{\mathbf{x}}(f,\tau);\theta) = \sum_{k=1}^{N+1} \alpha_{fk}\mathcal{N}_c(\hat{\mathbf{x}}(f,\tau); 0, \phi_{\tau fk}\mathbf{B}_{fk}), \tag{7}$$

where α_{fk} is a mixture weight ($\sum_k^{N+1} \alpha_{fk} = 1$), $\mathcal{N}_c(\mathbf{x};\mu,\mathbf{\Sigma})$ is a complex Gaussian distribution with a mean vector μ and a covariance matrix $\mathbf{\Sigma}$, $\phi_{\tau fk}$ is the power of the speech source of speaker k, and $\mathbf{B}_{fk}$ is the spatial correlation matrix of speaker k. Here $k = \{1, \cdots, N\}$ corresponds to the source classes, and $k = N+1$ corresponds to a noise class. In this paper, the number of speakers N in each meeting session was given.

After estimating the model parameter set $\theta = \{\phi_{\tau fk}, \mathbf{B}_{fk}, \alpha_{fk}\}$ by using, e.g., a maximum likelihood estimation method, the masks are given by the posterior probability of each class

$$M_k(f,\tau) = \frac{\alpha_{fk}\mathcal{N}_c(\hat{\mathbf{x}}(f,\tau);\theta_k)}{\sum_{k'=1}^{N+1}\alpha_{fk'}\mathcal{N}_c(\hat{\mathbf{x}}(f,\tau);\theta_{k'})}. \tag{8}$$

Here, we detail the update rule for parameter $\{\mathbf{B}_{fk}\}$, which characterizes this paper. The parameter update rules for $\{\phi_{\tau fk}, \alpha_{fk}\}$ can be found in [16].

In **a full-batch mode**, which we employed in our previous work [23], the update rule for $\{\mathbf{B}_{fk}\}$ is the same as that in [16],

$$\mathbf{B}^{full}_{fk} = \frac{\sum_{\tau=1}^{T} \frac{M_k(f,\tau)}{\phi_{\tau fk}}\hat{\mathbf{x}}(f,\tau)\hat{\mathbf{x}}^H(f,\tau)}{\sum_{\tau=1}^{T} M_k(f,\tau)}, \tag{9}$$

where T is the number of time frames in the entire recording of each session.

On the other hand, as our proposed **block-wise refinement**, we refine the parameter $\{\mathbf{B}_{fk}\}$ and the steering vectors $\mathbf{h}_k(f)$ every F frames, and update the MVDR beamformer coefficients (4) every F frames. That is, F is the block size. The proposed update rule at block b is

$$\mathbf{B}^b_{fk} = \eta\frac{\sum_{\tau=1}^{T_b} \frac{M_k(f,\tau)}{\phi_{\tau fk}}\hat{\mathbf{x}}(f,\tau)\hat{\mathbf{x}}^H(f,\tau)}{\sum_{\tau=1}^{T_b} M_k(f,\tau)} + (1-\eta)\mathbf{B}^{full}_{fk}, \tag{10}$$

where $T_b = F \cdot b$ is the time frame index at the end of the b-th block, and η is an adaptation parameter. If we use a "true" block-online mode ($\eta = 1$), we face a permutation problem in between time blocks, and we found that this problem is not easily solved for real meeting recordings. On the other hand, by employing $\{\mathbf{B}^{full}_{fk}\}$, the proposed update rule can mitigate the permutation problem in between time blocks, and can refine the parameters for every block.

It should be noted that, even with the block-wise refinement, parameters for estimating the mask (8) should be calculated at all the time-frequency slots to obtain $\mathbf{R}_{-k}(f)$ and $\mathbf{R}_k(f)$ for calculating (6).

Table 2. Recognition results (WER %) for eval. set.

	Mics.	Sync.	Enh.	F=50 (0.8sec)	100 (3.2sec)	150 (4.8sec)	300 (9.6sec)	500 (16sec)	1000 (32sec)
(a)	SP (ch.1)	-	off	42.2					
(b)	SP	off	on (full)	40.2					
(c)	SP	off	on (block ($\eta = 0.5$))	N/A	37.3	38.4	N/A	N/A	N/A
(d)	SP	off	on (block ($\eta = 0.8$))	N/A	38.9	38.4	N/A	N/A	N/A
(e)	SP	on	on (full)	31.4					
(f)	SP	on	on (block ($\eta = 0.5$))	30.5	29.7	28.9	29.2	28.9	29.2
(g)	SP	on	on (block ($\eta = 0.8$))	29.4	29.1	28.9	**28.8**	28.9	29.4
(h)	Headsets	-	-	18.8					

4. EXPERIMENTS

We evaluated performance in terms of the word error rate (WER). We recorded several Japanese conversations involving four to six participants in an office room adjacent to an exhibition hall. To mimic the exhibition scenario, babble noise was played through loudspeakers in the exhibition hall (see Fig. 1). The door was either open or closed depending on the session.

For recordings, we used four asynchronous smartphones (SPs) as described in Sec. 2. The degree of asynchronicity of the four SPs is summarized in [23]. We also used headset microphones (HM) and an 8-ch microphone array (AR) (see Fig. 1) for performance comparison.

The recordings were divided into training, development, and evaluation sets as shown in Table 1 "Office-Exh.". We also employed other meeting datasets recorded in a quiet office room ("Office") and a sound-proof room ("Sound-proof") [15], which were used only for training the ASR system. The length of each session was around 15 to 20 minutes.

4.1. Synchronization and enhancement setups

For the blind synchronization, the frame length was 256 ms and the frame shift was half of the frame length. In time-frequency mask based MVDR beamformer, the frame length and frame shift were 64 and 32 msec., respectively. For ASR evaluation, we have to determine which enhanced signal corresponds to which speaker. For this purpose, we used the correlation between the headset observations and enhanced signals.

We compare the performance in a full-batch mode with the performance in a block-online mode by using various block sizes $F = [50, 100, 150, 300, 500, 1000]$.

4.2. ASR setups

The DNN structure of our acoustic model was a fully connected feed-forward neural network with seven hidden layers (2048 units each) and 4100 output HMM states. The acoustic model (AM) was prepared using the following three-step procedure. As speech features, we employed 40 log mel filterbank coefficients with their delta and acceleration, and five left and five right context windows. We first trained a seed acoustic model using about 600 h of Japanese lecture speech data from the Corpus of Spontaneous Japanese (CSJ) [28], which was recorded with headset microphones. We then adapted this seed AM to the meeting speech recognition task by retraining the AM using all the headset recordings from our training dataset of conversation speech shown in Table 1. Finally, we adapted that AM to distant recordings by retraining it using the conversation speech training data recorded with a distant microphone from the microphone array (AR). Note that we did not retrain the DNN with the enhanced speech.

We used a Kneser-Ney smoothed word trigram language model (LM) [29], which was trained with transcripts of Japanese lecture speech data from the CSJ and the training set of the meeting recordings, in addition to the topic-related WWW data. These three text sets were mixed with weights that minimized the perplexity for the meeting development set. The vocabulary size was 157k.

We used manual annotation for voice activity detection (VAD) for ASR evaluation.

4.3. Results

Table 2 summarizes the speech recognition results in terms of WER (%) for the evaluation set. The WERs with and without synchronization ("Sync.") and enhancement ("Enh.") are shown. "Enh.= on (full)" and "Enh.= on (block)" stand for enhancement with a full-batch mode (9) and block-wise refinement (10), respectively. In this table, time offsets were compensated for in all cases, even in Sync.=off cases.

When using no synchronization or enhancement (Table 2(a)), and applying enhancement without synchronizing the sampling frequency mismatch (Table 2 (b)(c)(d)), the WERs were around 40 %. On the other hand, using both synchronization and enhancement techniques (Table 2 (e)(f)(g)), we can reduce the WERs. When we compare (e) with (f) and (g), we find that the proposed block-wise refinement successfully reduced the WERs by 8 % relative WER reduction. This confirms the effect of the proposed block-wise refinement. The frame size of $F \geq 150$ seems to work well for our dataset.

5. CONCLUSION

This paper addressed a front-end system for ASR of spontaneous conversational speech signals that were recorded with asynchronous distributed microphones. The basic concept is to combine blind synchronization and blind speech enhancement methods. To improve speech enhancement performance for real meetings that include the fluctuation of speaker positions, we proposed a method for the block-wise refinement of beamformer coefficients. Future work will include the implementation and evaluation of an online extension of the proposed approach without employing the first full-batch step.

6. ACKNOWLEDGEMENT

This work was partially supported by a Grant-in-Aid for Scientific Research (A) (KAKENHI Grant Number 16H01735) from Japan Society for the Promotion of Science (JSPS).

7. REFERENCES

[1] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig, "The Microsoft 2016 conversational speech recognition system," in *Proc. of ICASSP2017*, 2017, pp. 5255–5259.

[2] G. Saon, G. Kurata, T. Sercu, K. Audhkhasi, S. Thomas, D. Dimitriadis, B. Ramabhadran X. Cui, M. Picheny, L. L. Lim, B. Roomi, and P. Hall, "English conversational telephone speech recognition by humans and machines," in *Proc. of Interspeech2017*, 2017.

[3] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'CHiME' speech separation and recognition challenge: dataset, task and baselines," in *Proc. of ASRU2015*, 2015, pp. 504–511.

[4] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third ' CHiME ' speech separation and recognition challenge: Analysis and outcomes," *Computer Speech and Language*, 2016.

[5] E. Vincent, S. Watanabe, J. Barker, and R. Marxer, "The 4th CHiME speech separation and recognition challenge," 2016, http://spandh.dcs.shef.ac.uk/chime_workshop/presentations/CHiME_2016_Vincent_overview.pdf.

[6] A. Waibel, M. Bett, and M. Finke, "Meeting browser: tracking and summarizing meetings," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, 1998, pp. 281–286.

[7] A. Janin, J. Ang, S. Bhagat, R. Dhillon, J. Edwards, J. Macías-Guarasa, N. Morgan, B. Peskin, E Shriberg, A. Stolcke, C. Wooters, and B. Wrede, "The ICSI meeting project: resources and research," in *Proc. ICASSP'04 Meeting Recognition Workshop*, 2004.

[8] P. Wellner, M. Flynn, and M. Guillemot, "Browsing recorded meetings with Ferret," in *Proc. ICMI-MLMI*, 2004, pp. 12–21.

[9] F. Asano, K. Yamamoto, J. Ogata, M. Yamada, and M. Nakamura, "Detection and separation of speech events in meeting recordings using a microphone array," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2007, 2007, Article ID 27616, doi:10.1155/2007/27616.

[10] S. Renals, T. Hain, and H. Bourlard, "Recognition and understanding of meetings the AMI and AMIDA projects," in *Proc. ASRU'07*, 2007, pp. 238–247.

[11] G. Tur, A. Stolcke, L. Voss, S. Peters, D. Hakkani-Tur, J. Dowding, B. Favre, R. Fernandez, M. Frampton, M. Frandsen, C. Frederickson, M. Graciarena, D. Kintzing, K. Leveque, S. Mason, J. Niekrasz, M. Purver, K. Riedhammer, E. Shriberg, J. Tien, D. Vergyri, and F. Yang, "The CALO meeting assistant system," *IEEE Trans. Audio, Speech and Language Processing*, vol. 18, no. 6, pp. 1601–1611, Aug. 2010.

[12] T. Hain, L. Burget, J. Dines, P. N. Garner, F. Grezl, A. E. Hannani, M. Huijbregts, M. Karafiat, M. Lincoln, and V. Wan, "Transcribing meetings with the AMIDA systems," *IEEE Trans. Audio, Speech and Language Processing*, vol. 20, no. 2, pp. 486–498, 2012.

[13] "AMI: Augmented Multi-party Interaction," Available online: http://www.amiproject.org/ami-scientific-portal.

[14] "Rich Transcription Evaluation Project," Available online: http://www.itl.nist.gov/iad/mig/tests/rt/.

[15] T. Hori, S. Araki, T. Yoshioka, M. Fujimoto, S. Watanabe, T. Oba, A. Ogawa, K. Otsuka, D. Mikami, K. Kinoshita, T. Nakatani, A. Nakamura, and J. Yamato, "Low-latency real-time meeting recognition and understanding using distant microphones and omni-directional camera," *IEEE Trans. Audio, Speech and Language Processing*, vol. 20, no. 2, pp. 499–513, 2012.

[16] S. Araki, M. Okada, T. Higuchi, A. Ogawa, and T. Nakatani, "Spatial correlation model based observation vector clustering and MVDR beamforming for meeting recognition," in *Proc. of ICASSP2016*, 2016, pp. 385–389.

[17] S. Araki, N. Ito, M. Delcroix, A. Ogawa, K. Kinoshita, T. Higuchi, T. Yoshioka, D. Tran, S. Karita, and T. Nakatani, "Online meeting recognition in noisy environments with time-frequency mask based MVDR beamforming," in *Proc. of HSCMA2017*, 2017.

[18] X. Anguera, "BeamformIt (the fast and robust acoustic beamformer)," http://www.xavieranguera.com/beamformit/.

[19] S. Renals and P. Swietojanski, "Neural networks for distant speech recognition," in *Proc. of HSCMA2014*, 2014.

[20] P. Swietojanski, A. Ghoshal, and S. Renals, "Convolutional neural networks for distant speech recognition," *IEEE Signal Processing Letters*, vol. 21, no. 9, pp. 1120–1124, Sept. 2014.

[21] Y. Liu, P. Zhang, and T. Hain, "Using neural network front-ends on far field multiple microphones based speech recognitions," in *Proc. of ICASSP2014*, 2014, pp. 5579–5583.

[22] T. Higuchi, N. Ito, S. Araki, T. Yoshioka, M. Delcroix, and T. Nakatani, "Online MVDR beamformer based on complex Gaussian mixture model with spatial prior for noise robust ASR," *IEEE Trans. Audio, Speech and Language Processing*, vol. 25, no. 4, pp. 780–793, 2017.

[23] S. Araki, N. Ono, K. Kinoshita, and M. Delcroix, "Meeting recognition with asynchronous distributed microphone array," in *Proc. of ASRU2017*, 2017, (to appear).

[24] N. Ito, S. Araki, M. Delcroix, and T. Nakatani, "Probabilistic spatial dictionary based online adaptive beamforming for meeting recognition in noisy and reverberant environments," in *Proc. of ICASSP2017*, 2017, pp. 681–685.

[25] S. Miyabe, N. Ono, and S. Makino, "Blind compensation of inter-channel sampling frequency mismatch with maximum likelihood estimation in STFT domain," in *ICASSP2013*, 2013, pp. 674–678.

[26] S. Miyabe, N. Ono, and S. Makino, "Blind compensation of interchannel sampling frequency mismatch for ad hoc microphone array based on maximum likelihood estimation," *Signal Processing*, vol. 107, pp. 185–196, 2015.

[27] N. Ito, S. Araki, T. Nakatani, and T. Yoshioka, "Relaxed disjointness based clustering for joint blind source separation and dereverberation," in *Proc. of IWAENC2014*, 2014, pp. 269–273.

[28] S. Furui, K. Maezawa, and H. Isahara, "A Japanese national project on spontaneous speech corpus and processing technology," in *Proc of ISCA ASR*, 2000, pp. 244–248.

[29] R. Kneser and H. Ney, "Improved backing-off for m-gram language modeling," in *Proc. of ICASSP'95*, 1995, pp. 181–184.