IMPACT OF SINGLE-MICROPHONE DEREVERBERATION ON DNN-BASED MEETING TRANSCRIPTION SYSTEMS

Takuya Yoshioka[†,‡], *Xie Chen*[†], *Mark J. F. Gales*[†]

†Cambridge University Engineering Department, Cambridge, UK
‡NTT Communication Science Laboratories, Kyoto, Japan
{ty274,xc257,mjfg}@eng.cam.ac.uk

ABSTRACT

Over the past few decades, a range of front-end techniques have been proposed to improve the robustness of automatic speech recognition systems against environmental distortion. While these techniques are effective for small tasks consisting of carefully designed data sets, especially when used with a classical acoustic model, there has been limited evidence that they are useful for a state-of-the-art system with large scale realistic data. This paper focuses on reverberation as a type of distortion and investigates the degree to which dereverberation processing can improve the performance of various forms of acoustic models based on deep neural networks (DNNs) in a challenging meeting transcription task using a single distant microphone. Experimental results show that dereverberation improves the recognition performance regardless of the acoustic model structure and the type of the feature vectors input into the neural networks, providing additional relative improvements of 4.7% and 4.1% to our best configured speaker-independent and speaker-adaptive DNN-based systems, respectively.

Index Terms— Environmental robustness, meeting transcription, reverberation, deep neural network, single distant microphone

1. INTRODUCTION

Robustness againt environmental distortion caused by background noise and reverberation has been one of the main challenges facing automatic speech recognition. A variety of techniques have been proposed to tackle this problem. Of these techniques, this work concerns front-end approaches, which attempt to eliminate the effect of distortion from feature vectors. It is sometimes argued that front-end approaches are advantageous since they can be used with any forms of acoustic models, including those based on deep neural networks (DNNs) [1, 2], which have recently been breaking records in several speech recognition tasks. Nevertheless, these techniques have often been evaluated by using classical but less sophisticated systems and small tasks with a significant degree of mismatch between training and testing environments. In practically relevant voice-enabled services, there may be typical operating environments and, whenever possible, a large quantity of development data is collected before (and even after) they are released to minimize the environmental mismatch. Therefore, it is very important to investigate the way in which the environmental robustness techniques affect the performance of state-of-the-art systems trained on data collected in certain realistic scenarios.

Several researchers have already begun work on this topic. In [3], Rennie et al. evaluated one exemplary feature enhancement front-end [4] on a private large scale in-car speech recognition task by using an acoustic model based on fMPE [5] and CMLLR [6]. Their results showed that the feature enhancement provided a very marginal average performance gain, although it was effective for test utterances that had much lower SNRs than those of average training data. In [7], Seltzer et al. conducted experiments to evaluate the degree of noise robustness achieved by a DNN-based acoustic model. Their work achieved the best published result for the Aurora4 multi-condition task, which is an artificially created medium vocabulary task. They also showed that preprocessing feature vectors with a cepstral-domain MMSE noise reduction algorithm [8] degraded the performance of the DNN-based acoustic model[1]. These results clearly indicate that a significant portion of the gains from existing environmental robustness techniques might not be carried over to state-of-the-art systems in practical scenarios, calling for further investigation. Li and Sim [9] also explored a front-end processing approach to improving DNN-based acoustic models by using the Aurora2 digit recognition task.

In this paper, we focus on reverberation as a type of environmental distortion and investigate the effect of dereverberation processing on a single-microphone meeting transcription task. We use the AMI meeting corpus [10], which consists of a large quantity of meeting recordings and the corresponding high quality transcriptions. In particular, we employ a single distant microphone (SDM) setup, where only speech data from a single table-top microphone are available. As a result of the large distance between the microphone and the speakers, speech signals are contaminated by reverberation, thus making transcription very challenging [11]. To combat the reverberant distortion, we employ one exemplary dereverberation method proposed in [12] and experimentally investigate how it can affect the performance of DNN-based acoustic models for both speaker independent (SI) and speaker adaptive training (SAT) scenarios. Results show that employing dereverberation in the front-end provides a consistent improvement irrespective of the model structure and the type of features input into the DNNs. Since the test environments are acoustically close to those of the training data, these results mean that this front-end captures acoustic aspects that cannot be learned by the DNNs.

The remainder of this paper is organized as follows. Section 2 defines the meeting transcription task being considered. Section 3 provides a detailed description of the speech recognition systems that were built and tested. Section 4 reports and analyzes experimental results, followed by our conclusion in Section 5.

[1] It should be noted that, in [7], they also proposed a noise-aware training scheme, which provided a certain degree of improvement over the DNN-based baseline system.

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

2. MEETING SPEECH DATA

The AMI meeting corpus is used for the present investigation. The corpus consists of recordings of meetings conducted in English at three different sites. Each meeting has four participants. The meetings are either scenario-based role playing discussions or natural unconstrained conversations. Many of the meeting participants are non-native speakers and hence they may have very different accents and speaking styles. Each meeting was recorded with an eight-channel circular microphone array placed on a table in the meeting room, providing eight-channel synchronized acoustic signals. We mainly focus on an SDM task in which only the first channel data are used. As a consequence of the large distance between the microphones and speakers, the speech signals are distorted by reverberation and background noise. The background noise is almost stationary and the SNR does not vary significantly over different meetings. (Owing to this property, additive noise reduction yielded no improvement over our baseline system, and thus is not investigated further.) See the corpus website (https://www.idiap.ch/dataset/ami) for further details.

We selected a set of eight meetings (ES2009a-d and IS2009a-d) for a development test and another set of eight meetings (ES2008a-d and IS2008a-d) for an evaluation test. The remaining portion of the corpus was used for training. The development and evaluation sets each included eight speakers while the training set consisted of utterances from 175 speakers. We excluded overlapping speech segments from both the training and test sets, which left 59 hours of speech for training, 2.7 hours for the development test, and 2.6 hours for the evaluation. This data partition is consistent with [13]. Speech segments and speaker identities were obtained from the provided labels.

As regards the use of the speaker information, we considered two conditions: SI and SAT. The SI condition does not perform any type of model adaptation at either the training or test stages. The speaker information is utilized only for feature statistics normalization. In the SAT condition, acoustic models are trained with a speaker adaptive approach and test data can be swept over multiple times. (In reality, it is a challenging task to accurately estimate speaker labels from a single microphone input. However, we think that the SAT experiments are also meaningful in the sense that the results obtained allow us to understand whether there is an overlapping effect between front-end processing and model adaptation.)

Although our focus is on the SDM task, this paper also describes the performance of our systems, which use all the eight microphones. As previous work on meeting transcription often used microphone arrays, this may help us to understand the relative degree of effectiveness of dereverberation as opposed to that of beamforming. However, it should be noted that the MDM task assumes that dedicated multi-channel recording devices are available. In contrast, an SDM solution would work with ubiquitous recording devices such as digital voice recorders and smartphones and thus will be preferable in practical applications.

3. SYSTEM DESCRIPTION

A speech recognition system generally consists of a front-end processor and a back-end recognizer. Our objective is to compare three different front ends, detailed in Section 3.1, in various back-end configurations for both SI and SAT conditions. Note that all three front ends act on unsegmented speech and work on-line.

3.1. Front ends

We consider two front ends for the SDM task and one for the MDM task. The first SDM front-end performs conventional feature extraction and computes 13 MFCCs (including C0) as well as the first-, second-, and third-order delta parameters. Thus, a stream of 52-dimensional feature vectors is yielded and transmitted to a back-end recognizer. In addition, 24 log mel-filter bank channel outputs and their delta parameters up to the third order are also generated.

The second SDM front end performs dereverberation prior to feature extraction to reduce the distortion caused by reverberation. We employ the STFT-domain dereverberation algorithm that was first proposed in [12] for a two-microphone one-output case and generalized later in [14]. The single-channel version is briefly described in [11] and in the following.

The SDM dereverberating front end receives an unsegmented single-channel speech signal $y(t)$, which may contain multiple speakers, background noise, and reverberation, with t being a discrete time index. Let $y_n[k]$ denote an STFT coefficient calculated from $y(t)$, where n and k are the time frame and frequency bin indices, respectively. This front end attempts to dereverberate $y_n[k]$ at each frequency bin prior to computing spectral magnitudes by using a frequency-dependent linear filter as follows:

$$x_n[k] = y_n[k] - \sum_{\tau=T_\perp}^{T_\top} g_\tau^*[k] y_{n-\tau}[k], \tag{1}$$

where $*$ stands for complex conjugation. $T_\perp$ and $T_\top$ define the time period in which the filter has an effect. $T_\perp$ is normally set at 3 while $T_\top$ has a large value ($T_\top = 50$ in our implementation) to deal with long-term reverberation. $G = (g_{T_\perp}, \cdots, g_{T_\top})$ is a set of filter coefficients to be optimized. (Here and in the following, the frequency bin index k is omitted. All frequency bins are processed independently.)

The filter G is optimized to minimize the following objective function:

$$F_{\mathrm{WPE}} = \sum_{n=1}^{N} \left(\frac{\left| y_n - \sum_{\tau=T_\perp}^{T_\top} g_\tau^* y_{n-\tau} \right|^2}{\theta_n} + \log \theta_n \right), \tag{2}$$

where N is the total number of time frames. Here $\Theta = (\theta_1, \cdots, \theta_N)$ is a set of auxiliary variables that need to be optimized jointly with G, which leads to interleaved updates of G and Θ. The update of each θ_n is performed simply by calculating $\theta_n = \left| y_n - \sum_{\tau=T_\perp}^{T_\top} g_\tau^* y_{n-\tau} \right|^2$ for a fixed G. Using notation $\boldsymbol{g} = [g_{T_\perp}, \cdots, g_{T_\top}]^T$, where the superscript T indicates a non-conjugate transpose operation, G can be updated as

$$\boldsymbol{g} = \boldsymbol{R}^{-1} \boldsymbol{r}, \tag{3}$$

where $\boldsymbol{R}$ and $\boldsymbol{r}$ are given by the following equations:

$$\boldsymbol{R} = \sum_{t=1}^{N} \frac{\boldsymbol{y}_{n-T_\perp} \boldsymbol{y}_{n-T_\perp}^H}{\theta_n}, \quad \boldsymbol{r} = \sum_{t=1}^{N} \frac{\boldsymbol{y}_{n-T_\perp} y_n^*}{\theta_n} \tag{4}$$

with the superscript H representing a conjugate transposition and $\boldsymbol{y}_n$ being defined as $\boldsymbol{y} = [y_t, \cdots, y_{t-T_\top+T_\perp}]^T$. Two or three iterations provide good estimates and can be executed at a small computational cost. In practice, to enable the filter to follow acoustic changes in a room, $\boldsymbol{R}$ and $\boldsymbol{r}$ (and also G) are re-calculated every two seconds and smoothed over time blocks by taking a running average. The

optimized filter is applied to y_n to generate dereverberated STFT coefficient x_n, from which feature vectors are computed.

Finally, for beamforming, the MDM front end employs the BeamformIt algorithm [15], which has been adopted in many previous studies. This front end has eight-channel microphone signals as an input, generates a single-channel enhanced signal, and then extracts feature vectors from the enhanced audio. The beamforming algorithm steers acoustic beams towards meeting participants by exploiting meeting information that is extracted automatically from the input signals. By doing so, both reverberation and background noise are reduced before recognition is performed. A detailed description of the algorithm can be found in [15].

3.2. Back ends

For each of the three front ends described above, we considered several different configurations of DNN-based acoustic models. Each of our acoustic models is based on one of the two forms described below.

In one form, called a DNN-HMM hybrid, a DNN has an extended feature vector at each time frame, where the extended vector consists of a standard feature vector, such as MFCCs plus their first to third-order delta parameters, spliced with neighboring feature vectors within a context window of nine frames. With the extended feature vector denoted by $\boldsymbol{o}_n$, the DNN estimates the posterior probability, $p(s|\boldsymbol{o}_n)$, of HMM state s. The state likelihood needed for Viterbi decoding is then calculated based on these posteriors as $p(\boldsymbol{o}_n|s) \propto p(s|\boldsymbol{o}_n)/p(s)$. The state prior $p(s)$ is estimated by counting the occurances of state s in the training data. Our experiments used a DNN consisting of five hidden layers with 1,000 nodes per layer unless otherwise noted.

In the other form, called a DNN tandem, a DNN is utilized for nonlinear feature transformation [16]. The DNN used in the tandem system has only 26 nodes at the fifth hidden layer, which is called the bottleneck layer. After the network has been trained, each original feature vector is forwarded through the network and the linear outputs from the bottleneck layer are computed. These bottleneck layer outputs are further converted by a global semi-tied covariance transform [17] and concatenated with the corresponding 39-dimensional HLDA feature vector (described later) to form a 65-dimensional tandem feature vector. Finally, a GMM-HMM system is built to model the tandem feature vectors.

Both forms of DNN-based acoustic models are used in our SI and SAT systems. The steps in the construction of our SI and SAT systems are detailed below.

3.2.1. SI systems

We began creating our SI systems by applying speaker and meeting-level cepstral mean and variance normalization to the 52-dimensional MFCC-based feature vectors supplied by a front end. The normalized feature vectors were projected onto a 39-dimensional feature space by heteroscedastic linear discriminant analysis (HLDA) [18]. Then, a maximum likelihood GMM-HMM acoustic model was trained to model the HLDA features. The HMMs consisted of cross-word triphone HMMs with approximately 4,000 context-dependent states and 16 Gaussians per state. The model was further refined by MPE training, resulting in a baseline GMM-HMM system. This baseline system was used to perform forced alignment to produce frame-level state labels. Given these labels, DNNs for both hybrid and tandem acoustic models were trained to predict the state labels from the extended feature vectors. Finally, a GMM-HMM system for the tandem features was trained by using an MPE criterion. As described earlier, the input layer of our DNNs consisted of a context window of nine consecutive frames, where each frame was represented by a 52-dimensional feature vector consisting of 13 MFCCs concatenated with their first to third-order delta parameters or a 96-dimensional feature vector consisting of 24 mel-filter bank outputs plus their first to third-order deltas. The DNNs were initialized with discriminative pretraining [19] and then fine-tuned by using twelve or more epochs of back propagation. The actual numbers of epochs were determined based on 10% held-out cross validation data.

At the test stage, decoding was performed by bigram lattice generation with a 40K-word language model, followed by trigram lattice rescoring and confusion network rescoring. The language model was built from a variety of sources including transcriptions of AMI, ICSI, NIST, and ISL meetings, Callhome, Switchboard, Gigaword, and extra web data.

3.2.2. SAT systems

In addition to the SI systems, a series of SAT systems were also built as follows. First, a baseline SAT system based on GMM-HMMs was trained by using global and full CMLLR transforms [6]. The baseline SAT system was used to perform forced alignment, yielding frame-level state labels. The CMLLR transforms were also used to modify the original HLDA features, based on which DNN-HMM hybrid and tandem acoustic models were trained. This resulted in SAT hybrid and tandem systems, respectively.

At the test stage, the MPE-trained GMM-HMM SI system was used to produce adaptation supervision hypotheses. Based on these hypotheses, CMLLR transforms were optimized and applied to the original test data for each speaker of each meeting, which gave us speaker and meeting-normalized feature vectors. For the systems using GMM-HMM acoustic models (i.e., the baseline and tandem systems), global and full MLLR mean adaptation was also performed. The SAT systems also employed confusion network decoding.

4. EXPERIMENTAL RESULTS

Table 1 lists the word error rates (WERs) of our SI systems. As expected, the DNN-based acoustic model substantially improved the performance, whichever of the hybrid and tandem forms was used, compared with the baseline GMM-HMM system. The hybrid system, slightly outperformed the tandem system. We also observed the use of log mel-filter bank features (shown as FBANK in the table) yielded additional gains, which is consistent with the findings of previous work [20]. We further retrained the DNN by using the labels realigned with the initial hybrid system as suggested in [2, 19]. This did not improve the performance for the task being considered.

When we compare the columns labeled SDM and DRV, we can see that dereverberation processing consistently improved the performance regardless of the back-end configuration. The relative improvements averaged over the development and evaluation test sets were approximately in the 4.0 and 5.0 range. When we compare the results for the SDM dereverberating front end and the MDM front end, we can see that single-microphone dereverberation recovered as much as 30% of the performance loss resulting from the inaccessibility of extra microphones.

The performance of the SAT systems is shown in Table 2. For the baseline SDM system, the hybrid model was superior to the tandem model. However, when dereverberation or microphone array processing was used in the front end, the tandem system outperformed the hybrid system. Therefore, the impact of dereverberation

Table 1. WERs (%) for SI systems. SDM: baseline SDM front end, DRV: SDM dereverberating front end, MDM: MDM front end. Relative reduction from SDM shown in parentheses. Best numbers for each column shown in bold.

System	Dev			Eval		
	SDM	DRV	MDM	SDM	DRV	MDM
GMM-HMM	54.7	52.4	46.8	55.6	52.7	46.0
	—	(4.2)	(14.4)	—	(5.2)	(17.3)
Tandem/MFCC	46.3	44.5	39.7	47.0	44.9	39.5
	—	(3.9)	(14.3)	—	(4.5)	(16.0)
Hybrid/MFCC	45.7	43.5	40.1	45.4	44.0	38.9
	—	(4.8)	(12.3)	—	(3.1)	(14.3)
Hybrid/FBANK	43.8	41.8	**38.6**	**43.0**	**41.3**	**36.6**
	—	(4.6)	(11.9)	—	(4.0)	(14.9)
Hybrid/FBANK +realign	**43.5**	**41.7**	38.8	43.3	41.4	36.7
	—	(4.1)	(10.8)	—	(4.4)	(15.2)

Table 2. WERs (%) for SAT systems.

System	Dev			Eval		
	SDM	DRV	MDM	SDM	DRV	MDM
GMM-HMM	48.8	45.9	40.7	50.2	47.7	41.0
	—	(5.9)	(16.6)	—	(5.0)	(18.3)
Tandem/HLDA	42.1	**39.8**	**35.3**	42.2	**40.0**	**34.7**
	—	(5.5)	(16.2)	—	(5.2)	(17.8)
Hybrid/HLDA	**41.9**	40.8	36.8	**41.3**	40.5	35.4
	—	(2.6)	(12.2)	—	(1.9)	(14.3)

can be evaluated with the least bias by comparing the DRV tandem number with that of the SDM hybrid in the table. This comparison shows that dereverberation processing resulted in a relative performance improvement of 4.1%. Overall, dereverberation processing always improved the recognition performance although its impact varied for different back-end configurations. We can also observe that the gain from the single-microphone dereverberation processing was about 30 % of that from the MDM front end, which is consistent with the SI case.

The fact that the tandem system outperformed the hybrid system only with signal enhancement (i.e., single microphone dereverberation or microphone array beamforming) may be explained as follows. Without signal enhancement, the HLDA portion of the tandem feature will be highly contaminated by the inter-frame distortion caused by reverberation [21]. Since neither CMLLR nor MLLR transforms can deal satisfactorily with such inter-frame distortion, the baseline SDM system resulted in poorer performance with the tandem configuration. On the other hand, with signal enhancement, the HLDA portion of the tandem feature will contain less inter-frame distortion, which will allow the CMLLR and MLLR adaptations to work effectively. Nevertheless, this result essentially implies that the choice of the hybrid or tandem forms depends on the task and the front end being used.

As the impact of reverberation on speech signals spans a number of consecutive frames, the DNN may learn the reverberation characteristics even better by extending the temporal coverage of the context window for the DNN input. A further experiment was conducted to investigate the way in which the DNN's reverberation modelling capability changes with the context window size. In this experiment, the context window size of the DNN input was increased to 13 and 19 frames. The number of nodes at each hidden layer was also increased to 1,500 to accommodate the increased number of input features. When the 13-frame context window was used, the delta parameters were computed up to third order. With the 19-frame context window, the triple delta parameters were not used in order to keep the input layer narrower than the hidden layers. Only the SI SDM scenario was considered.

Table 3. WERs (%) obtained with different context window configurations for SI condition and FBANK input. The Hybrid/$1,000\times5$ system in the first row is the same as the Hybrid/FBANK system in Table 1.

System	Context window	Dev		Eval	
		SDM	DRV	SDM	DRV
Hybrid $1,000\times5$	9 frames	43.8	41.8	43.0	41.3
		—	(4.6)	—	(4.0)
Hybrid $1,500\times5$	9 frames	43.5	42.0	**42.6**	41.1
		—	(3.5)	—	(3.5)
	13 frames	**42.8**	41.8	42.9	41.2
		—	(2.3)	—	(4.0)
	19 frames	43.0	41.7	42.9	41.2
		—	(3.0)	—	(4.0)
Hybrid $2,000\times5$	9 frames	43.8	**41.3**	42.9	**40.4**
		—	(5.7)	—	(5.8)

The results of this experiment are shown in Table 3. It can be seen that widening the hidden layer to 1,500 nodes resulted in further WER reduction for the baseline SDM system. As a result, the relative gain from the dereverberation was slightly reduced. Extending the context window while fixing the number of hidden nodes resulted in further error reduction for the development set but degraded the evaluation set performance. This implies that a larger context window does not necessarily allow the DNN to learn a better and robust reverberation representation. Further increasing the number of hidden nodes to 2,000 degraded the performance of the baseline SDM system, while additional WER reduction was observed when dereverberation was used in the front end. The best performance was achieved with the 13-frame context window and $1,500\times5$ hidden nodes when the dereverberation was disabled while the system using the 9-frame context window and $2,000\times5$ hidden nodes performed the best when the dereverberation was enabled. Comparing these two systems, we see that the dereverberation processing provided a relative improvement of 4.7%.

5. CONCLUSION

In this paper, we evaluated the performance improvement that single-microphone dereverberation provides in a challenging SDM meeting transcription task. Dereverberation processing consistently improved the recognition performance of DNN-based acoustic models with different forms (i.e., hybrid or tandem), different input features, different context window lengths, and different numbers of hidden nodes in both SI and SA scenarios. This shows that dereverberation processing offers improved reverberation modelling power for a DNN-based acoustic model trained on nearly matched condition data.

6. ACKNOWLEDGEMENT

Xie Chen was funded by Toshiba Research Europe Ltd, Cambridge Research Lab.

7. REFERENCES

[1] S. Renals, N. Morgan, H. Bourlard, M. Cohen, and H. Franco, "Connectionist probability estimators in HMM speech recognition," *IEEE Trans. Speech, Audio Process.*, vol. 2, no. 1, pp. 161–174, 1994.

[2] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 1, pp. 30–42, 2012.

[3] S. Rennie, P. Dognin, and P. Fousek, "Robust speech recognition using dynamic noise adaptation," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2011, pp. 4592–4595.

[4] S. Rennie, T. Kristjansson, P. Olsen, and R. Gopinath, "Dynamic noise adaptation," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2006, pp. 1197–1200.

[5] D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau, and G. Zweig, "FMPE: Discriminatively trained features for speech recognition," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2005, pp. 961–964.

[6] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Comp. Speech, Language*, vol. 12, no. 2, pp. 75–98, 1998.

[7] M. L. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 7398–7402.

[8] D. Yu, L. Deng, J. Droppo, J. Wu, Y. Gong, and A. Acero, "Robust speech recognition using a cepstral minimum-mean-square-error-motivated noise suppressor," *IEEE Trans. Audio, Speech, Language Process.*, vol. 16, no. 5, pp. 1061–1070, 2008.

[9] B. Li and K. C. Sim, "Noise adaptive front-end normalization based on vector Taylor series for deep neural networks in robust speech recognition," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 7408–7412.

[10] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, I. McCowan, W. Post andD. Reidsma, and P. Wellner, "The AMI meeting corpus: a pre-announcement," in *Proceedings of Int. Worksh. Machine Learning for Multimodal Interaction*, 2006, pp. 28–39.

[11] T. Yoshioka, A. Sehr, M. Delcroix, K. Kinoshita, R. Maas, T. Nakatani, and W. Kellermann, "Making machines understand us in reverberant rooms: robustness against reverberation for automatic speech recognition," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 114–126, 2012.

[12] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H. Juang, "Blind speech dereverberation with multi-channel linear prediction based on short time Fourier transform representation," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2008, pp. 85–88.

[13] C. Breslin, KK Chen, M. J. F. Gales, and K. Knill, "Integrated online speaker clustering and adaptation," in *Proc. Interspeech*, 2011, pp. 1085–1088.

[14] T. Yoshioka and T. Nakatani, "Generalization of multi-channel linear prediction methods for blind MIMO impulse response shortening," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 10, pp. 2707–2720, 2012.

[15] X. Anguera, C. Wooters, and J. Hernando, "Acoustic beamforming for speaker diarization of meetings," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15, no. 7, pp. 2011–2022, 2007.

[16] F. Grezl, M. Karafiat, S. Kontar, and J. Cernocky, "Probabilistic and bottle-neck features for LVCSR of meetings," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2007, pp. IV–757–IV–760.

[17] M. J. F. Gales, "Semi-tied covariance matrices for hidden Markov models," *IEEE Trans. Speech, Audio Process.*, vol. 7, no. 3, pp. 272–281, 1999.

[18] N. Kumar and A. G. Andreou, "Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition," *Speech Commun.*, vol. 26, no. 14, pp. 283–297, 1998.

[19] F. Seide, G. Li, X. Chen, and D. Yu, "Feature engineering in context-depencent deep neural networks for conversational speech transcription," in *Proc. Workshop. Automat. Speech Recognition, Understanding*, 2011, pp. 24–29.

[20] A. Mohamed, G. Hinton, and G. Penn, "Understanding how deep belief networks perform acoustic modelling," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2012, pp. 4273–4276.

[21] M. J. F. Gales, "Model-based approaches to handling uncertainty," in *Robust Speech Recognition of Uncertain or Missing Data*, D. Kolossa and R. Haeb-Umbach, Eds., pp. 101–125. Springer, 2011.