THE UNIVERSITY of EDINBURGH

Edinburgh Research Explorer

Deep Beamforming Networks for Multi-channel Speech Recognition

Citation for published version:
Xiao, X, Watanabe, S, Erdogan, H, Lu, L, Hershey, J, Seltzer, ML, Chen, G, Zhang, Y, Mandel, M & Yu, D 2016, Deep Beamforming Networks for Multi-channel Speech Recognition. in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* Institute of Electrical and Electronics Engineers (IEEE), pp. 5745 - 5749, 41st IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2016, Shanghai, China, 20/03/16. https://doi.org/10.1109/ICASSP.2016.7472778

Digital Object Identifier (DOI):
10.1109/ICASSP.2016.7472778

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

OPEN ACCESS

Download date: 17. Apr. 2024

DEEP BEAMFORMING NETWORKS FOR MULTI-CHANNEL SPEECH RECOGNITION

Xiong Xiao[1], *Shinji Watanabe*[2], *Hakan Erdogan*[3], *Liang Lu*[4]
John Hershey[2], *Michael L. Seltzer*[5], *Guoguo Chen*[6], *Yu Zhang*[7], *Michael Mandel*[8], *Dong Yu*[5]
[1]Nanyang Technological University, Singapore, [2]MERL, USA,
[3]Sabanci University, Turkey, [4]University of Edinburgh, UK, [5]Microsoft Research, USA,
[6]Johns Hopkins University, USA, [7]MIT, USA, [8]Brooklyn College, CUNY, USA

ABSTRACT

Despite the significant progress in speech recognition enabled by deep neural networks, poor performance persists in some scenarios. In this work, we focus on far-field speech recognition which remains challenging due to high levels of noise and reverberation in the captured speech signals. We propose to represent the stages of acoustic processing including beamforming, feature extraction, and acoustic modeling, as three components of a single unified computational network. The parameters of a frequency-domain beamformer are first estimated by a network based on features derived from the microphone channels. These filter coefficients are then applied to the array signals to form an enhanced signal. Conventional features are then extracted from this signal and passed to a second network that performs acoustic modeling for classification. The parameters of both the beamforming and acoustic modeling networks are trained jointly using back-propagation with a common cross-entropy objective function. In experiments on the AMI meeting corpus, we observed improvements by pre-training each sub-network with a network-specific objective function before joint training of both networks. The proposed method obtained a 3.2% absolute word error rate reduction compared to a conventional pipeline of independent processing stages.

Index Terms— microphone arrays, direction of arrival, filter-and-sum beamforming, speech recognition, deep neural networks.

1. INTRODUCTION

The performance of ASR has been significantly improved in recent years [1] mainly due to three reasons: 1) the use of highly expressive acoustic models such as deep neural networks (DNN) and recurrent neural networks (RNN), e.g. long short term memory (LSTM) [2], that are able to handle large variations in speech data and directly optimized for the ASR task; 2) the use of large amount of training data that cover large variations of speech data; 3) the use of powerful GPUs that make the training of big model on big data feasible. The state-of-the-art ASR technology has achieved promising recognition accuracy in a number of speech transcription and benchmark tasks, however, far-field speech recognition remains an open challenge due to low signal to noise ratio (SNR), large volume of reverberation, and frequent overlapped speech, etc [3, 4, 5, 6].

The work reported here was carried out during the 2015 Jelinek Memorial Summer Workshop on Speech and Language Technologies at the University of Washington, Seattle, and was supported by Johns Hopkins University via NSF Grant No IIS 1005411, and gifts from Google, Microsoft Research, Amazon, Mitsubishi Electric, and MERL. Hakan Erdogan was partially supported by TUBITAK BIDEB-2219 program. Xiong Xiao was fully supported by the DSO funded project MAISON DSOCL14045, Singapore.

Beamforming is an indispensable front-end processing to improve the robustness of ASR systems in multi-channel far-field scenarios (e.g., [7, 8, 9]), and recent distant talk ASR benchmark such as the AMI meeting room transcription, CHiME and REVERB challenges also show the importance of beamforming in this scenario [3, 4, 10]. Although current beamforming techniques are able to improve the performance of far-field ASR, the full potential of microphone array processing has not yet been reached for several reasons. First, current mainstream beamforming techniques are developed to optimize signal level objective functions such as SNR [11] or acoustic likelihood [12], instead of directly maximizing speech recognition accuracy. Second, current techniques usually do not make use of the vast quantity of microphone array signals that can be easily collected from daily communication or by simulation.

To address the limitations of conventional beamforming methods, this paper proposes a learning-based deep beamforming network, which uses neural networks to predict the complex-valued parameters of a frequency-domain beamformer. With multi-channel inputs, the beamforming network filters the multi-channel short-time Fourier transform (STFT) of array signals to produce an enhanced signal. The proposed network enjoys lower computational complexity as compared to time domain methods using convolutional neural networks (CNN) [13]. We train the network using simulated multi-channel data from a given array geometry using all possible direction of arrival (DOA) angles, and test its generalization performance on AMI meeting corpus [3]. Furthermore, the beamforming network can be concatenated with the acoustic model neural network to form an integrated network that takes waveforms as input and produces senone posteriors. Since the gradient of the cost function can be back-propagated from the acoustic model network to the beamforming network, the beamforming processing can be optimized for the ASR task by using a large amount of multi-channel training data.

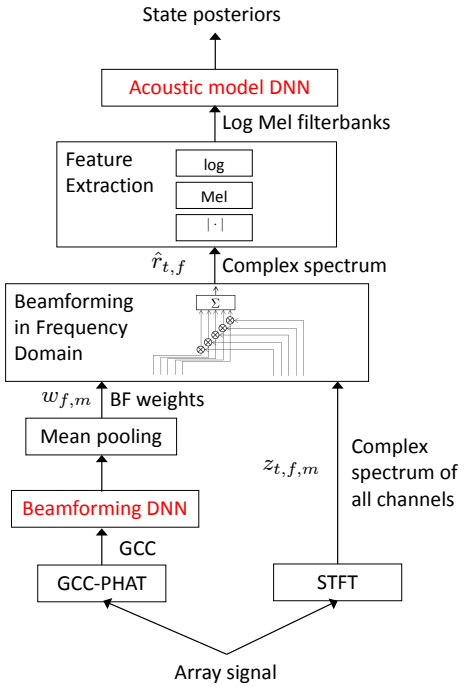
2. BEAMFORMING NETWORKS

2.1. System Overview

There may be many ways to deal with multi-channel inputs with neural networks. For example, a straightforward approach is to feed the array signals to a big network and let it predict the senone posteriors [14, 15]. However, such a network is too flexible to train parameters. Instead, our approach follows the successful conventional architecture using beamforming and ASR pipeline, and designs a computational network to reformulate the architecture with a deep network framework, where a part of computational nodes (beamforming and acoustic modeling) is learnable from training data.

The network structure we used in this paper is shown in Fig. 1. We use a neural network to predict the beamforming weights from the generalized cross correlation (GCC) [16] between microphones.

Fig. 1. Network structure of joint beamforming and acoustic model training. Blocks in red are trained from data while blocks in black are deterministic. Mean pooling means taking the mean of beamforming weights over an utterance.



The GCC encodes the time delay information between pairs of microphones and is essential for determining the steering vector of the beamformer. The predicted beamforming weights are averaged over an utterance (mean pooling) and then used to filter multi-channel STFT coefficients of the input signals to produce single-channel STFT coefficients. After that, conventional feature extraction steps are applied, including 1) computing the power spectrum of the beamformed complex spectrum; 2) Mel filtering; 3) logarithm dynamic range compression; 4) computing dynamic features, such as delta and acceleration; 5) optional utterance level mean normalization; 6) optional concatenation of 11 frames of consecutive features to incorporate contextual information. The output of the feature extraction pipeline is used for acoustic model training as usual.

While traditional methods can also estimate beamforming weights from the GCC, the neural network based prediction of beamforming weights has an important advantage, i.e. the prediction of beamforming weights can now be optimized for the ASR task, as gradients can flow from the acoustic model back to the beamforming network. In the next sections, we will describe the beamforming network in detail.

2.2. Per-frequency Beamforming

Let $z_{t,f,m} \in \mathbb{C}$ be the complex-valued STFT of frequency bin f for channel m at frame t. The filter-and-sum beamformer produces a complex linear combination of the input STFTs of all channels $\{z_{t,f,m}|m = 1, \cdots, M\}$ (M: number of microphones) as the enhanced signal $\hat{r}_{t,f}$, i.e.,

$$\hat{r}_{t,f} = \sum_{m=1}^{M} w_{f,m} z_{t,f,m}. \quad (1)$$

where $w_{f,m} \in \mathbb{C}$ is a filter coefficient, which is estimated by a DNN in our proposed framework. $w_{f,m}$ is frame independent, and this assumes that the room impulse response and speaker position are fixed during $t = 1, \ldots, T$. This is usually a reasonable assumption, but the adaptive filter-and-sum beamformer ($w_{f,m} \rightarrow w_{t,f,m}$) can potentially be robust to changes in room impulse response and speaker position during an utterance. After obtaining the beamformed signal $\hat{r}_{t,f}$ in the STFT domain, we extract typical features for ASR, such as log Mel filterbanks.

2.3. Input of Beamforming Network

The objective of the beamforming network is to predict reliable beamforming weights $w_{f,m}$ from reverberant and noisy multi-channel input signals. To achieve this, it needs to have information about the time delay between input channels, or equivalently the phase difference in the frequency domain. Although such information is contained in the raw signals, it is good to represent it in a way that can be used by the beamforming network easily.

There are several representations that encode the time delay information. Motivated by the work in [17], we choose to use the GCC. In [17], a feedforward neural network is used to predict the DOA of a single source from GCC. It is reported in [17] that when the network is trained with a large amount of simulated reverberant and noisy data, it can outperform traditional DOA estimation methods in real meeting room scenarios. Prediction of beamforming weights is closely related to predicting DOA. For example, the weights of the delay and sum beamformer (DSB) are completely determined by the array geometry and DOA. If the information in the GCC allows the network to predict the DOA reliably, it may also be sufficient for predicting beamforming weights reliably. However, there may be other options for input features of the beamforming network, for example, the spatial covariance matrices of frequency bins. The spatial covariance matrix not only contains time delay information, but also speech energy information, hence allowing the beamforming network to be aware of the phone context being processed. However, we will focus on using the GCC in this work.

The GCC features have a dimension of 588 and are computed as follows. The array we considered here is a circular array with 8 microphones and 20cm diameter, i.e. the array used in the AMI corpus [3]. For every 0.2s window, the GCC values between all 28 (C_2^8) microphone pairs are computed using the GCC-PHAT algorithm [16]. The overlap between two windows is 0.1s. For each microphone pair, only the center 21 elements of GCC values that contain the delay information up to +/- 10 signal samples are retained as the rest of the elements are not useful for the task here. This is because the maximum distance between any 2 microphones in the array is 20cm, which corresponds to less than a 10 sample delay at a sampling rate of 16 kHz and sound speed of 340m/s (0.2m/340m/s*16000samples/s=9.41 samples). As the maximum possible delay is less than 10 samples, it is not necessary to retain the GCC values that encode delay information of more than 10 samples. Therefore, the total number of GCC values used as the features for the beamforming network is $28 \times 21 = 588$. For more details of GCC feature extraction, and examples of GCC features in various DOA angles and environmental conditions, please refer to [17].

2.4. Output of Beamforming Network

For each input GCC feature vector, the beamforming network predicts a full set of beamforming weights $w_{f,m}$ for all frequency bins and channels. The real-valued weight vector to be predicted has a

dimension of 4,112 and is computed as follows. We use an FFT length of 512 and hence there are 257 frequency bins to cover 0Hz to 8000Hz. For each frequency bin, there are 8 complex weights, one for each microphone. As a conventional neural network is not able to handle complex values directly, the real and imaginary parts of each complex weight are predicted independently. Hence, the number of real-valued weights to be predicted for each GCC vector is $257 \times 8 \times 2 = 4112$. To make the estimates more reliable, we average the beamforming weights over an utterance, an operation that is called mean pooling. As stated previously, it is also possible to use time-dependent beamforming weights to track the change of source direction and environment over time. This could be achieved by simply not using mean pooling or by smoothing the beamforming weights only in neighboring windows. However, mean pooling is used in all experiments in this paper.

2.5. Structure of Beamforming and Acoustic Model Networks

The beamforming network can be either a feedforward DNN or RNN such as an LSTM. In this study, we experimented with a feedforward DNN with 2 hidden layers, each with 1,024 sigmoid hidden nodes. As described previously, the input and output dimensions of the network are 588 and 4,112, respectively.

Two types of acoustic model networks are used. For joint cross entropy (CE) training of beamforming and acoustic model networks, we use a feedforward DNN as the acoustic model which contains 6 hidden layers, each with 2,048 sigmoid hidden nodes. The input and output dimensions are 1,320 and 3,968, respectively. To achieve better ASR performance, we also train an LSTM-based acoustic model using the features processed by the beamforming network. The reason for using a feedforward DNN as the acoustic model is mainly due to our implementation, not because of any limitation of the proposed beamforming network. We will investigate the use of LSTMs in both the beamforming and acoustic model networks in the future.

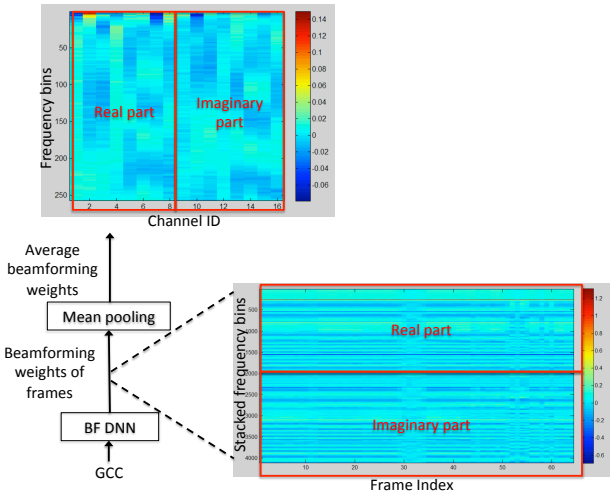
2.6. Training the Beamforming and Acoustic Model Networks

The network shown in Fig. 1 contains many hidden layers in addition to deterministic processing steps. The dynamic range of the gradients in the acoustic model and beamforming networks may be very different and their joint training may be slow and prone to falling into local minima. In practice, we first train the two networks in sequence, and then train them jointly as illustrated in following steps:

1. Train the beamforming network from simulated data by minimizing the mean square error (MSE) between predicted and optimal DSB weights.
2. Train the beamforming network from simulated data by minimizing the MSE between the predicted and clean log magnitude spectra.
3. Train the acoustic model network from ASR training data by CE criterion using the features generated by beamforming network from the second step.
4. Jointly train the beamforming and acoustic model networks from ASR training data using the CE criterion.

In the first step, as simulated data is used to train the beamforming network, the ground truth of the source DOA is known and so are the optimal DSB weights. The beamforming network can be trained to approximate the behavior of a DSB. This training step can be considered an initialization or pretraining of the beamforming network. In the second step, the beamforming network is trained such that they are optimal in predicting the clean magnitude spectrum, which is closer to the speech recognition task. In the third step, the acoustic model network is trained using the beamformed features. In the last step, the two networks are jointly trained with a large learning rate such that the networks can jump out of local minima caused by the previous steps and find a better set of weights.

Fig. 2. Illustration of predicted beamforming weights and the mean pooling step.



3. EXPERIMENTS

3.1. Settings

We generated 90 hours of simulated reverberant and noisy training data by convolving the 7,861 clean training utterances of the WSJ-CAM0 [18] corpus with room impulse responses (RIRs) simulated by the image method [19]. The T60 reverberation time is randomly sampled from 0.1s to 1.0s. Additive noise from the REVERB Challenge corpus [5] is added to the training data at an SNR randomly sampled from 0dB to 30dB.

The acoustic models are trained from the AMI corpus [3] multiple distant microphone (MDM) scenario. There are 75 hours of data in the training set and about 8 hours of data in the eval set. A trigram language model trained from the word label of the 75 hours training data is used for decoding.

For all beamforming (BF) experiments, the BF networks are used to generate enhanced speech in either waveform or filterbank features format, which is used to train the acoustic model from scratch.

3.2. Predicted Beamforming Weights

Fig. 2 shows an example of beamforming weights for an utterance predicted by the BF network. On the right of the figure is the 4,112-dimensional weight vector for each of the 0.2s-long windows in the utterance. It can be observed that the predicted weights are smooth across frames most of the time. The discontinuity may be from non-speech windows. The top of the figure shows the average beamforming weights reshaped into a 257×16 matrix. The left 8 columns show the real parts of the weights of the 8 channels, while the right

Table 1. WER (%) obtained by using beamforming networks on AMI meeting transcription task. "CMNspk" and "CMNutt" represents speaker and utterance based mean normalization respectively.

Row No.	Method	Training of BF networks	Feature		Acoustic Models		
			Resynthesize wave?	Feature Type	GMM	DNN (sMBR)	LSTM (sMBR)
1	IHM	-	-	MFCC (LDA+MLLT+fMLLR)	35.4	25.5	-
2	SDM1	-	-		68.0	53.8	-
3	DSB	-	Yes		60.8	47.9	-
4	BF networks	1. MSE in BF parameter space + simulated data (90 hours)	Yes		60.2	47.2	-
5		2. Refine with MSE in log magnitude spectrum space + simulated data (3 hours)	Yes		58.4	45.7	-
6			Yes	fbank (CMNspk)	-	46.1	-
7			Yes	fbank (CMNutt)	-	45.3	-
8			No	fbank (CMNutt)	-	45.7	-
9		3. Further refine with CE + AMI training data (75 hours)	No	fbank (CMNutt)	-	44.7	42.2
10	DSB	-	Yes	fbank (CMNutt)	-	-	44.8

8 columns show imaginary parts. We can observe stable patterns in the weight matrix.

3.3. ASR Results

The performance of the beamforming networks in terms of WER is shown in Table 1. The DNN systems were built using the Kaldi speech recognition toolkit [20], while the LSTM models were trained using CNTK [21]. For comparison, the results of the individual headset microphone (IHM), single distant microphone (SDM), and traditional DSB beamforming implemented in the BeamformIt toolkit [22] are also shown. The DSB is used as the baseline here (row 3). It is applied to entire meeting sessions without a voice activity detector. The DSB reduces the WER from 53.8% of SDM1 to 47.9% by using 8 channels. This result shows the effectiveness of beamforming in improving the performance of far-field ASR.

The BF network that are trained by only the first step in section 2.6 (row 4) obtain comparable results to the DSB. This is reasonable as in the first step of training, the BF network is trained to approximate the DSB. It is worth noting that the BF network is applied to each segment (as defined by the AMI corpus, a few seconds long on average) independently, while DSB is applied to entire audio files with the delays updated every few hundred milliseconds. So there is a minor difference between the two methods.

If the BF network is trained up to the second step (row 5), the WER is reduced to 45.7% when the DNN acoustic model is used. This is a significant improvement compared to training step 1 (row 4) and the DSB baseline (row 3). Until now the BF network has not used the AMI corpus for training. This shows that the BF network is able to generalize well to unseen room types and speakers if the array geometry of the test data is the same as that of the simulated training data.

So far the acoustic model uses MFCC features extracted from enhanced waveforms and is adapted using fMLLR. The joint training of AM and BF networks requires that the AM use features derived from the complex spectrum produced by the BF network, rather than from the resynthesized waveform. Hence, before the joint training, it is necessary to determine the performance difference between using MFCC features computed from enhanced waveforms and filterbank features computed directly from enhanced complex spectra. We first use filterbanks extracted from enhanced waveforms with speaker-level mean normalization (row 6). Comparing row 6 to row 5, we see a 0.4% increase in WER when switching from speaker adapted MFCCs to filterbanks. Then, we switch from speaker-based mean normalization to utterance-based mean normalization (row 7) and obtain 0.8% reduction in WER. Finally, we compare two filterbank features, one is extracted from enhanced waveforms (row 7), and the other is computed directly from enhanced complex spectra (row 8). The results show that the resynthesized waveforms perform slightly better. This could be due to the overlap and sum (OLS) operation used in waveform resynthesis. The OLS operation may have a smoothing effect that reduced processing variations.

The joint CE training of BF and AM networks using AMI data is shown in row 9. After the joint training, the AM network is further trained with sMBR training [23], while the BF network is frozen. This is because our current implementation does not support sMBR training of BF network yet. Results show that the CE fine tuning of BF network (row 9) produces a further WER reduction of 1.0% compared to the MSE training (row 8). This may be due to the fact that the BF network is now fine-tuned on the AMI data itself. It is worth noting that the BF network become more specific to the AMI data after the fine tuning and their performance may degrade for other corpora using the same array geometry. This is especially true for AMI as there are only few DOA angles present in the data from the 4-5 speakers, while in the simulated data, we used 360 DOA angles. Hence, the BF network trained on simulated is expected to work well for all DOA angles, while the BF network fine-tuned on the AMI data may be good for DOA angles that exist in the AMI training data, but worse for other DOA angles.

Finally, we also use the DSB and best BF network to generate filterbank features for an LSTM-based acoustic model. The results are shown in rows 10 and 9, respectively. It is observed that the LSTM improves the performance in both cases, and that the improvement of the BF network over the DSB is largely preserved.

4. CONCLUSIONS

We have investigated the feasibility of implementing beamforming with neural networks, specifically, a feedforward network. We have experimentally shown that BF networks are able to predict the real and imaginary parts of beamforming weights reliably from the GCC values. The predicted beamforming weights work well on unseen AMI test data for far-field ASR. These results validate the possibility of using neural networks for implementing beamforming. As a result, beamforming processing can now be trained together with the acoustic model to optimize for ASR tasks. In the future, we will investigate other ways of implementing beamforming with neural networks, such as using spatial covariance matrices instead of the GCC and more advanced network architectures like LSTMs. We will also study the feasibility of universal network-based beamforming that is independent of array geometry and the number of channels.

5. REFERENCES

[1] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, and Brian Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[2] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton, "Speech recognition with deep recurrent neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 6645–6649.

[3] Steve Renals, Thomas Hain, and Hervé Bourlard, "Recognition and understanding of meetings the ami and amida projects," in *IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU*, Kyoto, 12 2007, IDIAP-RR 07-46.

[4] Emmanuel Vincent, Jon Barker, Shinji Watanabe, Jonathan Le Roux, Francesco Nesta, and Marco Matassoni, "The second chimespeech separation and recognition challenge: Datasets, tasks and baselines," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 126–130.

[5] Keisuke Kinoshita, Marc Delcroix, Takuya Yoshioka, Tomohiro Nakatani, Armin Sehr, Walter Kellermann, and Roland Maas, "The reverb challenge: Acommon evaluation framework for dereverberation and recognition of reverberant speech," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2013, pp. 1–4.

[6] Mary Harper, "The automatic speech recognition in reverberant environments (ASpIRE) challenge," in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015, (accepted).

[7] Dirk Van Compernolle, Weiye Ma, Fei Xie, and Marc Van Diest, "Speech recognition in noisy environments with the aid of microphone arrays," *Speech Communication*, vol. 9, no. 5, pp. 433–442, 1990.

[8] Maurizio Omologo, Piergiorgio Svaizer, and Marco Matassoni, "Environmental conditions and acoustic transduction in hands-free speech recognition," *Speech Communication*, vol. 25, no. 1, pp. 75–95, 1998.

[9] Matthias Wölfel and John McDonough, *Distant Speech Recognition*, Wiley Online Library, 2009.

[10] Marc Delcroix, Takuya Yoshioka, Atsunori Ogawa, Yotaro Kubo, Masakiyo Fujimoto, Nobutaka Ito, Keisuke Kinoshita, Miquel Espi, Shoko Araki, Takaaki Hori, et al., "Strategies for distant speech recognition in reverberant environments," *EURASIP Journal on Advances in Signal Processing*, vol. 2015, no. 1, pp. 1–15, 2015.

[11] Jacob Benesty, Jingdong Chen, and Yiteng Huang, *Microphone array signal processing*, vol. 1, Springer Science & Business Media, 2008.

[12] Michael L Seltzer, Bhiksha Raj, and Richard M Stern, "Likelihood-maximizing beamforming for robust hands-free speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 5, pp. 489–498, 2004.

[13] Yedid Hoshen, Ron J Weiss, and Kevin W Wilson, "Speech acoustic modeling from raw multichannel waveforms," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015.

[14] Yulan Liu, Pengyuan Zhang, and Thomas Hain, "Using neural network front-ends on far field multiple microphones based speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 5542–5546.

[15] Pawel Swietojanski, Arnab Ghoshal, and Steve Renals, "Convolutional neural networks for distant speech recognition," *IEEE Signal Processing Letters*, vol. 21, no. 9, pp. 1120–1124, 2014.

[16] C. H. Knapp and G. C. Carter, "The generalized correlation method for estimation of time delay," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 24, August 1976.

[17] Xiong Xiao, Shengkui Zhao, Xionghu Zhong, Douglas L Jones, Eng Siong Chng, and Haizhou Li, "A learning-based approach to direction of arrival estimation in noisy and reverberant environments," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 2814–2818.

[18] T. Robinson, J. Fransen, D. Pye, J. Foote, and S. Renals, "WSJ-CAM0: a british english speech corpus for large vocabulary continuous speech recognition," in *Proceeding of ICASSP*, 1995, pp. 81–84.

[19] J.B. Allen and D.A. Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943?950, April 1979.

[20] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. Dec. 2011, IEEE Signal Processing Society, IEEE Catalog No.: CFP11SRW-USB.

[21] Dong Yu, Adam Eversole, Michael Seltzer, Kaisheng Yao, Brian Guenter, Oleksii Kuchaiev, Yu Zhang, Frank Seide, Guoguo Chen, Huaming Wang, Jasha Droppo, Amit Agarwal, Chris Basoglu, Marko Padmilac, Alexey Kamenev, Vladimir Ivanov, Scott Cyphers, Hari Parthasarathi, Bhaskar Mitra, Zhiheng Huang, Geoffrey Zweig, Chris Rossbach, Jon Currey, Jie Gao, Avner May, Baolin Peng, Andreas Stolcke, Malcolm Slaney, and Xuedong Huang, "An introduction to computational networks and the computational network toolkit," Tech. Rep., 2014.

[22] Xavier Anguera, Chuck Wooters, and Javier Hernando, "Acoustic beamforming for speaker diarization of meetings," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2011–2022, 2007.

[23] Karel Veselỳ, Arnab Ghoshal, Lukás Burget, and Daniel Povey, "Sequence-discriminative training of deep neural networks.," in *INTERSPEECH*, 2013, pp. 2345–2349.