Smoothing along Frequency in Online Neural Network Supported Acoustic Beamforming

Jens Heitkaemper, Jahn Heymann, Reinhold Haeb-Umbach

Paderborn University, Department of Communications Engineering, Paderborn, Germany
{heitkaemper, heymann, haeb}@nt.uni-paderborn.de

Abstract

We present a block-online multi-channel front end for automatic speech recognition in noisy and reverberated environments. It is an online version of our earlier proposed neural network supported acoustic beamformer, whose coefficients are calculated from noise and speech spatial covariance matrices which are estimated utilizing a neural mask estimator. However, the sparsity of speech in the STFT domain causes problems for the initial beamformer coefficients estimation in some frequency bins due to lack of speech observations. We propose two methods to mitigate this issue. The first is to lower the frequency resolution of the STFT, which comes with the additional advantage of a reduced time window, thus lowering the latency introduced by block processing. The second approach is to smooth beamforming coefficients along the frequency axis, thus exploiting their high inter-frequency correlation. With both approaches the gap between offline and block-online beamformer performance, as measured by the word error rate achieved by a downstream speech recognizer, is significantly reduced. Experiments are carried out on two copora, representing noisy (CHiME-4) and noisy reverberant (voiceHome) environments.

***Index Terms*—** Distant speech recognition, acoustic beamforming, time-frequency mask estimation

1. INTRODUCTION

With the advance of smart voice-controlled loudspeakers distant Automatic Speech Recognition (ASR) has found widespread use in the home environment. These devices employ microphone arrays to capture sound, and it has been shown that acoustic beamforming techniques provide significant recognition rate improvements compared to single-channel recognition. The state-of-the-art in microphone array beamforming for ASR is to estimate masks which indicate for each time frequency (TF) bin whether it is dominated by speech or by distortions. The masks can be either estimated by a generative model, such as a time-variant complex Gaussian Mixture Model [1], or by a Neural Network (NN) [2, 3]. With these masks the spatial covariance matrices of speech and noise are estimated, and from these matrices, in turn, the coefficients of statistically optimum beamformers, such as the Multi-Channel Wiener Filter (MWF), the Minimum Variance Distortionless Response (MVDR), or the maximum-SNR, also called Generalized Eigenvalue (GEV), beamformer are computed. However, many of these popular speech enhancement schemes do not offer the low-latency processing required for voice-controlled devices.

Several recent works presented low-latency online mask based beamformers [1, 4, 5, 6], however, with a noticeable hit in recognition rate of a subsequent speech recognizer, compared to the results achieved by a corresponding offline beamformer. In this contribution we evaluate the differences between block-online and offline neural network dependent beamforming to identify sources of error. For two of them, which have significant impact on the Word Error Rate (WER), we propose solutions, which considerably reduce the gap between block-online and offline recognition performance.

The first is concerned with the normalization of the data at the input of the neural network. A recent study [6] compared normalization methods in online mask estimation and showed that the recursive estimation of mean and variance severely degrades the beamforming results compared to estimating these statistical moments offline on a whole utterance. We propose to waive variance normalization altogether, and carry out only recursive mean normalization, which turns out to be a reasonable approach.

The second issue is related to the well-known sparsity of the speech signal in the Short time Fourier Transform (STFT) domain. Many offline mask based beamformers use frequency domain representations with high frequency resolution to achieve sparseness of speech, e.g., [3, 7, 8]. However, due to this sparsity it may take a long time for some frequency bins until sufficiently many observations of speech are observed to render the speech spatial covariance matrix estimation reliable. As a consequence beamforming on these frequencies performs poorly. We propose to reduce the STFT window size and, correspondingly, the frame advance, to solve this issue. Since the sparsity of speech, on the other hand, is beneficial for some enhancement tasks we also offer an alternative. To this end we exploit the strong correlation of speech and most noises across frequency and propose to smooth the beamformer coefficient vectors along the frequency axis.

We evaluate our modifications on two databases. The first is the CHiME-4 dataset, which is characterized by low Signal to Noise Ratios (SNRs), different noise types, and little reverberation [9]. The second is a database composed of Wallstreet Journal (WSJ) utterances [10] which are convolved with Room Impulse Responses (RIRs) of the voiceHome corpus, to which typical household noises are added [11, 12]. This dataset represents the case of both strong additive and convolutive distortions.

This paper is structured as follows. In the next section we give an overview of the neural network supported acoustic beamformer. Then we discuss online mask estimation in Section 3, followed by block-online beamformer coefficient estimation in Sec. 4, where we discuss different frequency domain representations and propose a frequency smoothing scheme. Section 5 presents speech recognition results, and Section 6 offers some conclusions.

2. SYSTEM OVERVIEW

Fig. 1 displays the structure of the neural network supported acoustic beamformer, which we proposed in [2, 3]. In the following we briefly describe the system.

We assume a multi-channel frequency domain input $\mathbf{Y}(\ell,k) = [Y_1, Y_2, ..., Y_D]^\mathsf{T}$ consisting of D microphone signals:

$$\begin{aligned} Y_d(\ell,k) &= H_d(k)S(\ell,k) + N_d(\ell,k) \\ &= X_d(\ell,k) + N_d(\ell,k), \quad d = 1,\dots,D. \end{aligned} \tag{1}$$

Here, $S(\ell,k)$, $H_d(k)$, $X_d(\ell,k)$ and $N_d(\ell,k)$ are the STFT coefficients of the source signal, of the Room Impulse Response (RIR) from the source to the d-th microphone, and the source signal and the noise as observed by the d-th microphone, at time frame index ℓ and frequency bin index k. In the following these indices will be dropped wherever possible without sacrificing clarity.

This work has been supported by Deutsche Forschungsgemeinschaft (DFG) under contract no Ha3455/14-1.

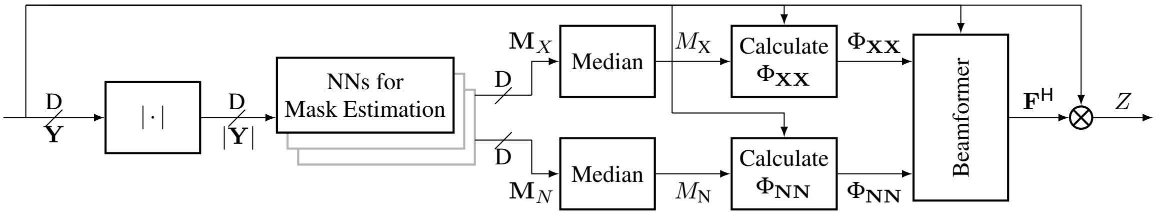

Fig. 1: System overview of mask based beamforming

A NN is applied to the magnitude spectrum of each microphone signal to predict which TF bin is dominated by clean speech and which by distortions, however the weights and biases of the D networks are tied across channels. The D estimated masks $\mathbf{M}_\nu$ are reduced to one mask by applying the median operator with $\nu \in [X, N]$. Next the the spatial covariance matrices of speech and noise, $\Phi_{\mathbf{XX}}$ and $\Phi_{\mathbf{NN}}$, are estimated from the TF bins dominated by speech and noise, respectively.

The beamformer coefficient vector is computed from them using the well-known Minimum Variance Distortionless Response (MVDR) criterion, which minimizes the noise energy with a distortionless constraint on the target signal [13]:

$$\mathbf{F}_{\text{MVDR}} = \underset{\mathbf{F}}{\text{argmin}}\, \mathbf{F}^{\mathsf{H}} \Phi_{\mathbf{NN}} \mathbf{F} \quad \text{s.t.} \quad \mathbf{F}^H \tilde{\mathbf{H}} = 1, \tag{2}$$

where $\tilde{\mathbf{H}} = [1, \ldots \tilde{H}_D]^{\mathsf{T}}$ is the vector of relative transfer functions, i.e. the acoustic transfer functions normalized to a reference microphone (here mic. #1). The solution to this optimization problem can be written in the form [14]:

$$\mathbf{F}_{\text{MVDR}} = \frac{\Phi_{\mathbf{NN}}^{-1} \Phi_{\mathbf{XX}}}{\text{tr}\left\{\Phi_{\mathbf{NN}}^{-1} \Phi_{\mathbf{XX}}\right\}} \mathbf{u}, \tag{3}$$

where $\mathbf{u}$ is a unit vector pointing to the reference microphone, and $\text{tr}\{\cdot\}$ is the trace operator.

Finally the beamformer is applied to the input signal

$$Z(\ell, k) = \mathbf{F}^{\mathsf{H}}(k) \mathbf{Y}(\ell, k) \tag{4}$$

to obtain Z, which is an estimate of the source signal as received at the reference microphone. Here, $(\cdot)^{\mathsf{H}}$ denotes the Hermitian transpose.

3. ONLINE NEURAL MASK ESTIMATION

The original topology of the neural mask estimator [3] is slightly changed to enable online processing. The initial bidirectional LSTM layer with 256 nodes is replaced by a unidirectional causal LSTM layer of twice the number of nodes. The following three feedforward layers remain, however, unmodified.

To further account for online processing, the normalization of the input to the network is modified. In the offline scenario a normalization to zero mean and unit variance is carried out, where these statistical moments are estimated on a whole utterance. Changing this to a recursive mean and variance normalization as proposed in earlier works [6] leads to a significant performance drop. The likely cause is the poor variance estimate, since it is well-known that variance estimates are much noisier than mean estimates.

We therefore removed variance normalization completely and only carried out a recursive mean normalization. This will obviously only work if the training and test data have approximately the same power. In case that there is a significant mismatch in the power of the signals, a scale factor needs to be estimated in advance (and then kept fixed). For the databases tested in this contribution, this, however, was not necessary.

4. BLOCK-ONLINE SPATIAL COVARIANCE AND BEAMFORMER COEFFICIENT ESTIMATION

To accommodate for low latency, both $\Phi_{\mathbf{XX}}$ and $\Phi_{\mathbf{NN}}$ are estimated recursively in a block-online fashion:

$$\Phi_{\nu\nu}(nN) = \alpha_\nu \Phi_{\nu\nu}((n-1)N) + (1-\alpha_\nu)\hat{\Phi}_{\nu\nu}(nN) \tag{5}$$

$$\hat{\Phi}_{\nu\nu}(nN) = \sum_{\ell=0}^{N-1} M_\nu(nN+\ell)\mathbf{Y}(nN+\ell)\mathbf{Y}^{\mathsf{H}}(nN+\ell) \tag{6}$$

Here M_ν is the mask estimator output, $\nu \in [\mathbf{X}, \mathbf{N}]$, N the block size and α_ν a forgetting factor. Given these matrices the beamformer coefficients are updated according to (3) every N frames. Note that these computations are carried out for each frequency bin independently.

Due to the sparseness of speech in the STFT domain, it may take some time after a speech onset until speech energy is observed in a certain frequency bin. Figure 2 shows the histogram of the time it takes until the first speech dominated frame is observed. Time measurement starts when the first speech dominated frame is observed anywhere on the frequency axis. The histogram is the average over all frequencies and all utterances in the CHiME-4 simulated development set. Here, the STFT size was chosen to be equal to the frame width, and the STFT frame advance is one quarter of the STFT size.

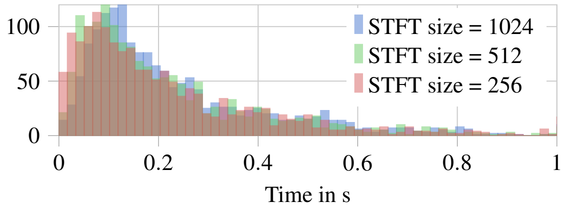

Fig. 2: Histograms of length of the time interval between the first speech dominated frame anywhere on the frequency axis and the first speech dominated frame in a given frequency, averaged over frequencies and utterances of the CHiME-4 simulated development set. Histograms are given for different STFT sizes.

From the histogram we conclude that frequencies without initial speech observations are a common issue. As a consequence the initial estimates of the spatial covariance matrix of speech are very unreliable, thus leading to poor beamformer coefficient estimates on these frequencies.

4.1. STFT Size

The histograms shown in Fig. 2 point to a possible solution: reducing the STFT size and thus the frequency resolution, alleviates the problem of missing speech observations to some extent. While the median of the histogram for STFT size of 1024 (= 64 ms) is at 0.174 s, it is reduced to 0.151 s and 0.145 s for sizes of 512 and 256. In our setup we use a fixed ratio of four between the STFT size and the frame advance. Thus, reducing the STFT size also results in more frames per second and thus more observations overall.

Further, since we choose the STFT window to be equal to the STFT size (i.e., no zero padding), the latency introduced by the block processing of the STFT is also reduced.

4.2. Frequency Smoothing

In [15] it has been observed that sparseness of speech is most prominent for a STFT size of 64 ms. The sparseness property, i.e., the fact that the speech energy is concentrated in a few frequency bins, is exploited in many signal processing algorithms, such as in mask-based blind source separation, e.g. [16], or in Minimum Statistics based noise tracking [17], to name just two examples.

However, the reduction of the frequency resolution impacts this sparseness property negatively. Furthermore there may be other constraints which prohibit to modify a given STFT size. We therefore propose an alternative to lowering the STFT size. The idea is to exploit the correlation of speech on the frequency axis, i.e. to estimate parameters at a certain frequency by using information from neighboring frequencies. We experimented with smoothing different variables across frequency, such as the spatial covariance matrix of speech or its principal eigenvector, but finally settled on smoothing the beamforming vectors themselves with a decaying window, since this gave the best results.

Fig. 3 displays the magnitude of a variable describing the normalized cross correlation:

$$\rho_{\mathbf{F}}(\Delta k) = \frac{\overline{\mathbf{F}^{\mathsf{H}}(k+\Delta k)\mathbf{F}(k)} - \overline{\mathbf{F}^{\mathsf{H}}(k+\Delta k)} \cdot \overline{\mathbf{F}(k)}}{\sqrt{\left\|\mathbf{F}(k+\Delta k) - \overline{\mathbf{F}(k+\Delta k)}\right\|^2 \cdot \left\|\mathbf{F}(k) - \overline{\mathbf{F}(k)}\right\|^2}} \tag{7}$$

between the beamforming vectors at neighboring frequencies, for a selected number of frequencies. Here $\overline{(\cdot)}$ denotes time averaging. A relatively high correlation can be observed, and it is only mildly dependent on the frequency.

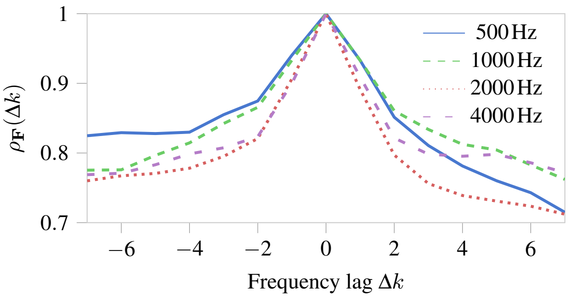

Fig. 3: $\rho_{\mathbf{F}}$ of beamforming coefficients over frequency lag Δk for multiple frequencies describing the normalized cross correlation for MVDR beamformer coefficients, which had been computed offline over a whole utterance with STFT size of 1024.

This correlation can be exploited by smoothing the beamformer coefficient vectors along the frequency axis. However, beamformer coefficients estimated on frequencies with sufficient observations of speech dominated frames are more reliable than those computed on frequencies with fewer observations. This can be accounted for by employing the speech masks in the smoothing:

$$\tilde{\mathbf{F}}(\ell,k) = \frac{\sum_{i=-\lfloor\frac{\kappa}{2}\rfloor}^{\lfloor\frac{\kappa}{2}\rfloor} \tilde{M}_X(\ell,k+i)\mathbf{F}(\ell,k+i)}{\sum_{i=-\lfloor\frac{\kappa}{2}\rfloor}^{\lfloor\frac{\kappa}{2}\rfloor} \tilde{M}_X(\ell,k+i)}, \tag{8}$$

where $\tilde{M}_X = (\ell,k) = \sum_{\iota=0}^{\ell} M_X(\iota,k)$ represents the cumulative sum over all seen $M_X(i,k)$ up to frame ℓ.

5. EXPERIMENTAL RESULTS

5.1. Datasets

The proposed modifications were tested by carrying out speech recognition experiments on two databases.

The CHiME-4 task features real and simulated 6-channel audio data of prompts taken from the WSJ0 5k Corpus [10] with 4 different types of real-world background noise (pedestrian, cafe, street, bus) [9]. The amount of reverberation is negligible. The evaluation set consists of both simulated test data (simu) and real recordings of speech in noisy environments (real).

As a second dataset we used WSJ utterances (test eval92 5k) which are convolved with voiceHome RIRs and voiceHome background noise [16], similar to the setup proposed by Bertin et al. [11, 12]. The reverberation times are in the range of 395 - 585 ms. It is worth mentioning that the RIRs had been recorded in three different houses, such that training, cross-validation and test can use disjoint RIRs to ensure generalization. The voiceHome background noise is very dynamic and typically found in households, e.g., vacuum cleaner, dish washing or interviews on television. Compared to the original setup of [11] we reduced the SNR and set it to values randomly drawn between 0 dB and 10 dB to simulate a more challenging environment.

The training targets of the neural mask estimator were adjusted to account for the two types of distortion, noise and reverberation, present on this task: while the target for the speech mask estimation is chosen to be the direct (line-of-sight) signal and early reflections, the target for the noise masks consists of the noise and the speech convolved with the tail of the RIRs (i.e., the late reverberation), see [18] for details.

For both datasets, the user position can be considered stationary for the duration of an utterance. Thus beamforming, where the coefficients are computed offline on the whole utterance, can be considered the best solution if low latency is not an issue.

5.2. Backend

As ASR back end we use a Wide Residual Network as proposed in [19] with logarithmic mel filterbank features and two Long-Short-Term-Memory (LSTM) layers serving as accoustic model. It consists of several convolutional layers with residual connections, followed by two BLSTM layers and two linear layers. The hyper-parameters were adapted from [19]. The neural acoustic models are trained on the respective training set of the considered corpora. Note that the acoustic model itself operates offline since we focus on the online processing in the front-end but can be replaced by an online version to achieve a fully online operating system. Decoding was carried out with the KALDI toolkit [20] using a trigram language model without rescoring.

5.3. Input Normalization

Table 1 compares different input normalization methods w.r.t. the achieved word error rates of the ASR decoder on the CHiME-4 dataset. The first entry shows the performance achieved using the offline mask estimator using a bi-directional LSTM layer as the recurrent layer in the network, and carrying out mean and variance estimation on the whole utterance (offline utt.).

Replacing the BLSTM layer with a LSTM layer of twice the size hardly affects the recognition results. The entry named ”fixed” refers to using the statistical moments estimated during training, as inspired by [21], while ”rec. mean + var” is the recursive mean and variance estimation per utterance used in [6]. It results in a significant hit in recognition accuracy. The proposed recursive estimation of only the mean results in error rates which come close to the offline utterance-wise normalization. For all following results with online mask estimation the recursive mean normalization is used.

Table 1: WER on CHiME-4 for different mask estimators with offline beamforming.

Recurrent Layer	Normalization	WER [%]	
		simu	real
BLSTM	offline utt.	8.39	10.01
LSTM	offline utt.	8.41	10.17
	fixed	8.50	11.51
	rec. mean + var	10.04	14.50
	rec. mean	8.40	11.18

5.4. STFT Size

Next we investigate the impact of the STFT size on the beamformer. For every STFT size a new mask estimator was trained, with the above described topology, except for the input and output layers which were adjusted accordingly.

To clarify the difficulties in online beamforming vector estimation for high frequency resolution, Figure 4 depicts the Cosine Distance (CD)

$$CD(\ell, k) = 1 - \frac{\mathbf{F}_{\text{off}}(k) \cdot \mathbf{F}_{\text{on}}^{\mathsf{H}}(\ell, k)}{||\mathbf{F}_{\text{off}}(k)|| \cdot ||\mathbf{F}_{\text{on}}(\ell, k)||} \tag{9}$$

between block-online and offline beamforming vectors for STFT-sizes of 1024 and 256, averaged over the CHiME-4 development dataset. For this and all experiments with block-online beamforming the block size N, see eq. (5), was set to correspond to 80 ms and the forgetting factor α_ν to 0.95.

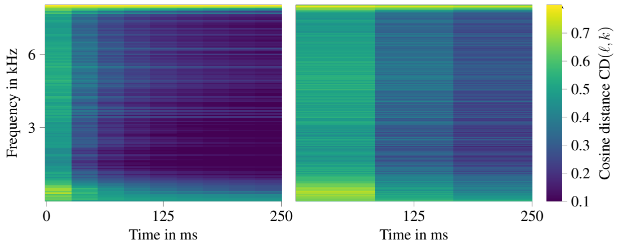

Fig. 4: Cosine distance between offline and block-online beamformer for STFT-size 256 (left) and 1024 (right) over time since the start of the first speech observation in the utterance and over frequency.

The figure clearly shows that the beamforming vector computed with lower STFT-size much quicker approaches the offline vector. One can also notice that for some frequencies it takes a long time until a vector similar to the offline case can be estimated, which is, however, less pronounced for the small STFT size than for the large.

Table 2: WER in % on CHiME-4 for different STFT sizes.

Method	STFT size	WER	
		simu	real
channel #5	–	13.36	18.60
Offline	1024	8.39	10.01
	256	8.05	10.22
Online	1024	9.37	13.32
	512	8.70	12.66
	256	8.54	11.71
	128	9.57	12.14

Table 2 shows the impact of the STFT size on the presented speech enhancement system. While offline beamforming is hardly affected, the results clearly demonstrate the benefit of reduced STFT size on block-online beamforming. Only if the STFT size and thus the frequency resolution is too low (here at size of 128) the error rate increases again.

5.5. Frequency Smoothing

Additonally, we experimented with the proposed frequency smoothing of the beamforming vector. Table 3 presents ASR results on the CHiME-4 corpus. In the online case the ASR results for STFT size 1024 with smoothing almost reach the results obtained with the STFT size of 256 while the offline result scarcely changes. The contributions of the neighboring frames within the smoothing window were weighted as described in eq. (8) and its size was set to $\kappa = 5$ neighboring frequency bins. We also experimented with increasing the size with frequency, which, however, did not improve performance. The best block-online results are achieved by the beamformer calculated on STFT size of 256 with a WER of 11.71 % on CHiME-4 real test data.

Table 3: WER on CHiME-4 for different beamformers.

Method	STFT size	Smoothing	WER [%]	
			simu	real
Offline	1024	–	8.39	10.01
	1024	✓	8.39	10.11
Online	1024	–	9.37	13.32
	256	–	8.54	11.71
	1024	✓	8.70	12.01

5.6. voiceHome

Finally, Table 4 presents results on the voiceHome dataset with the best configuration found on CHiME-4. On the voiceHome database the offline beamformer achieves a WER of 13.30 % and the block-online beamformer 17.70 % at STFT size 1024. The error rates are given separately for the subset of test sentences corrupted by speech-like noises (TV, Dialog, etc.) and other noises. We can observe that the reduction of STFT size is counterproductive for the performance on the speech-like noises. This is probably because the sparseness of speech, or more precisely, the w-disjoint orthogonality [15] is lost: desired speech and interfering speech overlap each other even at the TF bin resolution. The frequency smoothing method, however, leads to improved error rate also for the subset of the database corrupted by speech-like noise, indicating that the sparseness of speech is better preserved and speech and speech-like noises do not overlap so much in individual frequency bins.

Table 4: WER in % on voiceHome eval dataset for different beamformers.

Method	STFT size	Smooth.		noise	
			all	other	sp-like
ch #1	–	–	25.73	18.42	28.30
Offline	1024	–	13.30	12.70	13.52
Online	1024	–	17.70	17.01	17.94
	256	–	17.96	16.85	18.36
	1024	✓	16.85	15.10	17.47

6. CONCLUSIONS

We identified the initial lack of speech observations in some frequency bins as source of degradation when going from offline to block-online mask-based beamforming and proposed two solutions to overcome this issue. The first is a reduction of the STFT size, which, however, compromises the sparseness and w-disjoint orthogonality of speech. The second is the smoothing of the beamforming vectors along the frequency axis. Both methods lead to a reduction of the gap in ASR performance between offline and block-online beamforming, as observed on two challenging ASR tasks.

References

[1] T. Higuchi, N. Ito, T. Yoshioka, and T. Nakatani, "Robust MVDR beamforming using time-frequency masks for online/offline ASR in noise," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar 2016, pp. 5210–5214.

[2] J. Heymann, L. Drude, A. Chinaev, and R. Haeb-Umbach, "BLSTM supported GEV beamformer front-end for the 3rd CHiME challenge," in *Proc. Worksh. Automat. Speech Recognition, Understanding*, 2015, pp. 444–451.

[3] J. Heymann, L. Drude, and R. Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar 2016, pp. 196–200.

[4] Takuya Higuchi, Keisuke Kinoshita, Nobutaka Ito, Shigeki Karita, and Tomohiro Nakatani, "Frame-by-frame closed-form update for mask-based adaptive mvdr beamforming," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. April 2018, IEEE.

[5] Matthias Zöhrer, Lukas Pfeifenberger, Günther Schindler, Holger Fröning, and Franz Pernkopf, "Resource efficient deep eigenvector beamforming," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. April 2018, IEEE.

[6] Christoph Boeddeker, Hakan Erdogan, Takuya Yoshioka, and Reinhold Haeb-Umbach, "Exploring practical aspects of neural mask-based beamforming for far-field speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. April 2018, IEEE.

[7] Yuzhou Liu, Anshuman Ganguly, Krishna Kamath, and Trausti Kristjansson, "Neural network based time-frequency masking and steering vector estimation for two-channel mvdr beamforming," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. April 2018, IEEE.

[8] J. Thiemann and E. Vincent, "An experimental comparison of source separation and beamforming techniques for microphone array signal enhancement," in *2013 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, Sept 2013, pp. 1–5.

[9] Emmanuel Vincent, Shinji Watanabe, Aditya Arie Nugraha, Jon Barker, and Ricard Marxer, "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," *Comput. Speech Lang.*, vol. 46, no. C, pp. 535–557, nov 2017.

[10] John Garofolo, D Graff, D Paul, and D Pallett, "CSR-I (WSJ0) complete LDC93S6A," *Web Download. Philadelphia: Linguistic Data Consortium*, 1993.

[11] Nancy Bertin, Ewen Camberlein, Emmanuel Vincent, Romain Lebarbenchon, Stephane Peillon, Eric Lamand, Sunit Sivasankaran, Frederic Bimbot, Irina Illina, Ariane Tom, Sylvain Fleury, and Eric Jamet, "A French corpus for distant-microphone speech processing in real homes," in *Interspeech*, 2016.

[12] Sunit Sivasankaran, Emmanuel Vincent, and Irina Illina, "A combined evaluation of established and new approaches for speech recognition in varied reverberation conditions," *Computer Speech & Language*, 2017.

[13] S. Gannot, E. Vincent, S. Markovich-Golan, and A. Ozerov, "A consolidated perspective on multi-microphone speech enhancement and source separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. PP, no. 99, pp. 1–1, 2017.

[14] M. Souden, J. Benesty, and S. Affes, "On optimal frequency-domain multichannel linear filtering for noise reduction," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, no. 2, pp. 260–276, 2007.

[15] S. Rickard and O. Yilmaz, "On the approximate w-disjoint orthogonality of speech," in *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 2002, vol. 1, pp. I–529–I–532.

[16] Dang Hai Tran Vu and Reinhold Haeb-Umbach, "Exploiting temporal correlations in joint multichannel speech separation and noise suppression using hidden markov models," in *International Workshop on Acoustic Signal Enhancement (IWAENC2012)*, Sep. 2012.

[17] R. Martin, "Noise power spectral density estimation based on optimal smoothing and minimum statistics," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 5, pp. 504–512, Jul 2001.

[18] J. Heymann, L. Drude, and R. Haeb-Umbach, "A generic neural acoustic beamforming architecture for robust multichannel speech processing," *Computer Speech & Language*, vol. 46, no. Supplement C, pp. 374–385, 2017.

[19] R. Haeb-Umbach J. Heymann, L. Drude, "Wide residual BLSTM network with discriminative speaker adaptation for robust speech recognition," in *CHiME4 Workshop*, 2016.

[20] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, . Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, Rue Marconi 19, Martigny, Dec 2011, number Idiap-RR-04-2012, IEEE Signal Processing Society, IEEE Catalog No.: CFP11SRW-USB.

[21] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *CoRR*, vol. abs/1502.03167, 2015.