DEEP LEARNING BASED SPEECH BEAMFORMING

Kaizhi Qian[1], Yang Zhang[2*], Shiyu Chang[2], Xuesong Yang[1], Dinei Florencio[3], Mark Hasegawa-Johnson[1]*

[1]University of Illinois at Urbana-Champaign, USA
[2] IBM T. J. Watson Research Center, USA [3]Microsoft Research, USA

{kqian3,xyang45,jhasegaw}@illinois.edu, {yang.zhang2,shiyu.chang}@ibm.com, dinei@microsoft.com

ABSTRACT

Multi-channel speech enhancement with ad-hoc sensors has been a challenging task. Speech model guided beamforming algorithms are able to recover natural sounding speech, but the speech models tend to be oversimplified or the inference would otherwise be too complicated. On the other hand, deep learning based enhancement approaches are able to learn complicated speech distributions and perform efficient inference, but they are unable to deal with variable number of input channels. Also, deep learning approaches introduce a lot of errors, particularly in the presence of unseen noise types and settings. We have therefore proposed an enhancement framework called DEEPBEAM, which combines the two complementary classes of algorithms. DEEPBEAM introduces a beamforming filter to produce natural sounding speech, but the filter coefficients are determined with the help of a monaural speech enhancement neural network. Experiments on synthetic and real-world data show that DEEPBEAM is able to produce clean, dry and natural sounding speech, and is robust against unseen noise.

Index Terms— multi-channel speech enhancement, ad-hoc sensors, beamforming, deep learning, WaveNet

1. INTRODUCTION

Multi-channel speech enhancement with ad-hoc sensors has long been a challenging task [1]. As the traditional benchmark in multi-channel enhancement tasks, beamforming algorithms do not work well with with ad-hoc microphones. This is because most beamformers need to calibrate the speaker location as well as the interference characteristics, so that it can turn its beam toward the speaker, while suppressing the interference. However, neither of the two vital information can be accurately measured, due to the missing sensor position information and microphone heterogeneity [2].

Another class of beamforming algorithms avoid measuring the speaker position and interference. Instead, they introduce prior knowledge on speech, and find the optimal beamformer by maximizing the "speechness" criteria, such as sample kurtosis [3], negentropy [4], speech prior distributions [5, 6], fitting glottal residual [7] etc. In particular, the GRAB algorithm [7] is able to outperform the closest microphone strategy even in very adverse real-world scenarios. Despite their success, these algorithms are bottlenecked by their oversimplified prior knowledge. For example, GRAB only models glottal energy, resulting in vocal tract ambiguity.

On the other hand, deep learning techniques are well known for their ability to capture complex probability dependencies and efficient inference, and thus have been widely used in single-channel speech enhancement tasks [8–13]. Unfortunately, directly applying deep enhancement networks to multi-channel enhancement suffers from two difficulties. First, deep enhancement techniques often produce a lot of artifacts and nonlinear distortions [11, 12] which are perceptually undesirable. Second, neural networks often generalize poorly to unseen noise and configurations, whereas in speech enhancement with ad-hoc sensors, such variability is large.

As it turns out, these problems can in turn be resolved by traditional beamforming. Therefore, several algorithms [14–18] have been proposed that applies deep learning to predict time-frequency masks, and then beamforming to produce the enhanced speech. However, these methods are confined to frequency domain, which suffers from two problems for our application. First, they to not work well for ad-hoc microphones, because of the spatial correlation estimation errors. Second, our application is for human consumption, but the frequency-domain methods suffer from phase distortions and discontinuities, which impede perceptual quality.

Motivated by this observation, we have proposed an enhancement framework for ad-hoc microphones called DEEPBEAM, which combines deep learning and beamforming, and which directly works on waveform. DEEPBEAM introduces a time-domain beamforming filter to produce natural sounding speech, but the filter coefficients are iteratively determined with the help of WaveNet [19]. It can be shown that despite the error-prone enhancement network, DEEPBEAM is able to converge approximately to the optimal beamformer under some assumptions. Experiments on both the simulated and real-world data show that DEEPBEAM is able to produce clean, dry and natural sounding speech, and generalize well to various settings.

2. PROBLEM FORMULATION

To formally define the problem, denote $s[t]$ as the clean speech signal. Suppose there are K channels of observed signals, $y_k[t], k = 1, \cdots, K$, which are represented as

$$y_k[t] = s[t] * i_k[t] + n[t] * j_k[t] \tag{1}$$

where $*$ denotes discrete convolution, $n(t)$ denotes additive noise. $i_k[t]$ and $j_k[t]$ are the impulse responses of the signal reverberation and noise reverberation in the k-th channel respectively. Our goal is to design a τ-tap beamformer $h_k[t], k = 1, \cdots, K$, whose output is defined as

$$x[t] = \sum_{k=1}^{K} y_k[t] * h_k[t] \tag{2}$$

For notational brevity, define

$$\begin{aligned} &\boldsymbol{s} = [s[1], \cdots, s[T]]^T \quad \boldsymbol{x} = [x[1], \cdots, x[T]]^T \\ &\boldsymbol{y}_k = [y_k[1], \cdots, y_k[T]]^T \quad \boldsymbol{y} = [\boldsymbol{y}_1^T, \cdots, \boldsymbol{y}_K^T]^T \\ &\boldsymbol{h} = [h_1[1], \cdots, h_1[\tau], h_2[1], \cdots, h_K[\tau]]^T \end{aligned} \tag{3}$$

* Denotes equal contribution.

This paper was funded by QNRF grant NPRP 7-766-1-140.

which are all random vectors. Also define convolutional matrices

$$\boldsymbol{Y}_k = \begin{bmatrix} y_k[1] & & & \\ y_k[2] & y_k[1] & & \\ \vdots & \vdots & \ddots & \\ y_k[\tau] & y_k[\tau-1] & \cdots & y_k[1] \\ \vdots & \vdots & & \vdots \\ y_k[T] & y_k[T-1] & \cdots & y_k[T-\tau+1] \end{bmatrix} \tag{4}$$

and

$$\boldsymbol{Y} = [\boldsymbol{Y}_1, \cdots, \boldsymbol{Y}_K] \tag{5}$$

With these notations, Eq. (2) can be simplified as

$$\boldsymbol{x} = \boldsymbol{Y}\boldsymbol{h} \tag{6}$$

The target of designing the beamformer is to minimize the weighted mean squared error (MSE):

$$\min_{\boldsymbol{x}=\boldsymbol{Y}\boldsymbol{h}} \mathbb{E}\left[\|\boldsymbol{x}-\boldsymbol{s}\|_{\boldsymbol{W}}^2 | \boldsymbol{y}\right] \tag{7}$$

where $\|\boldsymbol{x}\|_{\boldsymbol{W}}^2 = \boldsymbol{x}^T\boldsymbol{W}\boldsymbol{x}$; $\boldsymbol{W}$ is a positive definite weight matrix, which, in our case, is a diagonal matrix of $\mathrm{Var}^{-1}(s[t]|\boldsymbol{y})$.

Eq. (7) is a Wiener filtering problem [20], whose solution is

$$\boldsymbol{x}^* = \boldsymbol{P}\mathbb{E}[\boldsymbol{s}|\boldsymbol{y}] \tag{8}$$

where

$$\boldsymbol{P} = \boldsymbol{Y}(\boldsymbol{Y}^T\boldsymbol{W}\boldsymbol{Y})^{-1}\boldsymbol{Y}^T\boldsymbol{W} \tag{9}$$

is in fact the *projection matrix* onto the beamforming output space. So by Eq. (8), $\boldsymbol{x}^*$ is essentially projecting $\mathbb{E}[\boldsymbol{s}|\boldsymbol{y}]$ onto the space that is representable by the beamforming filter.

As shown by Eq. (8), solving the Wiener filtering problem requires computing $\mathbb{E}[\boldsymbol{s}|\boldsymbol{y}]$, which, due to the complex probabilistic dependencies, we would like to introduce a deep neural network to learn. However, as discussed, training a neural network to directly predict $\mathbb{E}[\boldsymbol{s}|\boldsymbol{y}]$ from the multi-channel input $\boldsymbol{y}$ suffers from inflexible input dimensions, artifacts and poor generalization. DEEPBEAM tries to resolve these problems and find an approximate solution.

3. THE DEEPBEAM FRAMEWORK

In this section, we will describe the DEEPBEAM algorithm. We will first outline the algorithm, and then describe the neural network structure it applies. Finally, a convergence analysis is introduced.

3.1. The Algorithm Overview

As mentioned, DEEPBEAM introduces a deep enhancement network to learn the posterior expectation, while addressing its limitations. First, DEEPBEAM are regularized by the beamformer to generalize well to unseen noise and microphone configurations. Second, it tolerates the distortions and artifacts generated by the neural network. Formally, the neural network outputs an inaccurate prediction of the posterior expectation $\mathbb{E}[\boldsymbol{s}|\boldsymbol{\xi}]$,

$$f(\boldsymbol{\xi}) = \mathbb{E}[\boldsymbol{s}|\boldsymbol{\xi}] + \boldsymbol{\varepsilon}(\boldsymbol{\xi}) \tag{10}$$

where $\boldsymbol{\xi}$ is a *single-channel* noisy observation, and $\boldsymbol{\varepsilon}(\boldsymbol{\xi})$ is the prediction error. The goal of DEEPBEAM is to approximate the optimal beamformer given the inaccurate enhancement network. Alg. 1 shows the description of the DEEPBEAM algorithm. A graph of the DEEPBEAM framework is shown in Fig. 1.

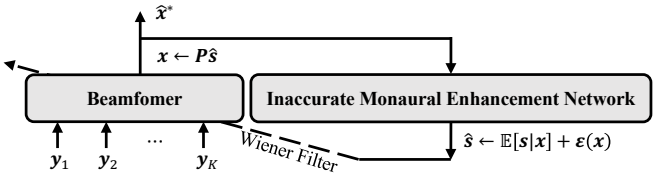

Fig. 1: DEEPBEAM framework.

Algorithm 1 The DEEPBEAM algorithm.

Input: Multi-channel noisy speech observations $\boldsymbol{y}$;
A neural network that predicts $f(\boldsymbol{\xi})$ (Eq. (10)) from any single-channel noisy observation $\boldsymbol{\xi}$.
Output: Beamformer output $\hat{\boldsymbol{x}}^*$.

Initialization:
Find the 'cleanest' channel k^* by finding the channel that has the smallest 0.4 quantile of its squared sample points.
Set $\boldsymbol{x}^{(0)} = \boldsymbol{y}_{k^*}$.
Iteration:
for $n = 1$ to maximum number of iterations **do**
  Feed $\boldsymbol{x}^{(n-1)}$ to the monaural enhancement network, and obtain its output

$$\hat{\boldsymbol{s}}^{(n)} = f(\boldsymbol{x}^{(n-1)}) = \mathbb{E}[\boldsymbol{s}|\boldsymbol{x}^{(n-1)}] + \boldsymbol{\varepsilon}(\boldsymbol{x}^{(n-1)}) \tag{11}$$

  Update the beamformer coefficients and output

$$\boldsymbol{x}^{(n)} = \boldsymbol{P}\hat{\boldsymbol{s}}^{(n)} \tag{12}$$

end for
return $\hat{\boldsymbol{x}}^* = \boldsymbol{x}^{(N)}$

Alg. 1 essentially alternates between the posterior expectation and projection iteratively. It will be shown in section 3.3 that as long as the error term $\boldsymbol{\varepsilon}$ is not too large, this iteration will approximately converge to the optimal beamformer output.

One elegance of DEEPBEAM is that $\boldsymbol{x}^{(n)}$ can be regarded as a noisy observation, and shares some statistical structures with the true noisy observations, $\boldsymbol{y}_k$. To see this, notice that by Eq. (12), $\boldsymbol{x}^{(n)}$ is the output of a beamformer on $\boldsymbol{y}$. Therefore, it can be shown that $\boldsymbol{x}^{(n)}$ also takes the form of Eq. (1), with the same speech and noise source, but with a different impulse response. This justifies the use of one monaural enhancement network to take care of all the $\boldsymbol{x}^{(n)}$.

3.2. Enhancement Network Structure

DEEPBEAM is a general framework, in which the choice of the neural network structure is not fixed. The following network structure is just one of the structures that produce competitive results.

The enhancement network applied here is similar to [12], which is inspired by WaveNet [19]. Formally, denote the *quantized* speech samples as $\tilde{s}[t]$, and the samples of $\boldsymbol{x}^{(n)}$ as $x^{(n)}[t]$. Then the enhancement network predicts the posterior probability mass function (PMF) of $\tilde{s}[t]$:

$$p(\tilde{s}[t]|\boldsymbol{x}^{(n)}) \approx p(\tilde{s}[t]|x^{(n)}[t-\tau_r], \cdots, x^{(n)}[t+\tau_r]) \tag{13}$$

Here we have restricted the probabilistic dependency to span τ_r time steps. Cross-entropy is applied as the loss function.

Similar to WaveNet, the enhancement network consists of two modules. The first module, called the dilated convolution module,

contains a stack of dilated convolutional layers with residual connections and skip outputs. The second module, called the post processing module, sums all the skip outputs and feeds them into a stack of fully connected layers before producing the final output.

There are two major differences from the standard WaveNet structure. First, the input to the enhancement network is the noisy observation waveform $\boldsymbol{x}^{(n)}$ instead of the clean speech. Second, to account for the future dependencies, the convolutional layers are noncausal 1×3 instead of the causal 1×2.

After the posterior distribution is predicted, the posterior moments, $\boldsymbol{E}[\boldsymbol{s}|\boldsymbol{x}^{(n)}]$ and $\mathrm{Var}[s[t]|\boldsymbol{y}]$ (for computing $\boldsymbol{W}$), are computed as the moments of the predicted PMF.

3.3. Convergence Analysis

In order to analyze the convergence property of DEEPBEAM, we assume the following bound on the error term

$$\mathbb{E}[\|\boldsymbol{P}\boldsymbol{\varepsilon}(\boldsymbol{x}^{(n)})\|_{\boldsymbol{W}}^2|\boldsymbol{y}] \leq \rho\mathbb{E}[\|\boldsymbol{x}^{(n)} - \boldsymbol{s}\|_{\boldsymbol{W}}^2|\boldsymbol{y}] \tag{14}$$

where $\rho < 0.5$ is some constant. This assumption is actually not quite stringent, because it does not bound the weighted norm of $\boldsymbol{\varepsilon}(\boldsymbol{x}^{(n)})$ itself, but its projected value $\boldsymbol{P}\boldsymbol{\varepsilon}(\boldsymbol{x}^{(n)})$. In fact, the projection can drastically reduce the weighted norm of the error term. For example, most of the artifacts and nonlinear distortions that the enhancement network introduces cannot possibly be generated by beamforming on $\boldsymbol{y}$, and therefore will be removed by the projection. The only errors that are likely to remain are residual noise and reverberations. This is one advantage of combining beamforming filter and neural network. This assumption is also very intuitive. It means that the projected output error is always smaller than input error.

Then, we have the following theorem.

Theorem 1. *Suppose Eq.* (14) *holds. Then*

$$\limsup_{n\to\infty}\mathbb{E}[\|\boldsymbol{x}^{(n)} - \boldsymbol{x}^*\|_{\boldsymbol{W}}^2|\boldsymbol{y}] \leq u \tag{15}$$

where

$$\begin{aligned} u = &\frac{2\rho}{1-2\rho}\mathbb{E}[\|\boldsymbol{s} - \boldsymbol{x}^*\|_{\boldsymbol{W}}^2|\boldsymbol{y}] \\ &+ \frac{2}{1-2\rho}\sup_n \mathbb{E}[\|\boldsymbol{P}\mathbb{E}[\boldsymbol{s}|\boldsymbol{x}^{(n)}] - \boldsymbol{x}^*\|_{\boldsymbol{W}}^2|\boldsymbol{y}] \end{aligned} \tag{16}$$

Proof. On one hand, from Eqs. (11) and (12)

$$\begin{aligned} &\mathbb{E}[\|\boldsymbol{P}\boldsymbol{\varepsilon}(\boldsymbol{x}^{(n)})\|_{\boldsymbol{W}}^2|\boldsymbol{y}] = \mathbb{E}[\|\boldsymbol{x}^{(n+1)} - \boldsymbol{P}\mathbb{E}[\boldsymbol{s}|\boldsymbol{x}^{(n)}]\|_{\boldsymbol{W}}^2|\boldsymbol{y}] \\ \geq &\frac{1}{2}\mathbb{E}[\|\boldsymbol{x}^{(n+1)} - \boldsymbol{x}^*\|_{\boldsymbol{W}}^2|\boldsymbol{y}] - \mathbb{E}[\|\boldsymbol{P}\mathbb{E}[\boldsymbol{s}|\boldsymbol{x}^{(n)}] - \boldsymbol{x}^*\|_{\boldsymbol{W}}^2|\boldsymbol{y}] \end{aligned} \tag{17}$$

On the other hand, by orthogonality principle

$$\mathbb{E}[\|\boldsymbol{x}^{(n)} - \boldsymbol{s}\|_{\boldsymbol{W}}^2|\boldsymbol{y}] = \mathbb{E}[\|\boldsymbol{x}^{(n)} - \boldsymbol{x}^*\|_{\boldsymbol{W}}^2|\boldsymbol{y}] + \mathbb{E}[\|\boldsymbol{s} - \boldsymbol{x}^*\|_{\boldsymbol{W}}^2|\boldsymbol{y}] \tag{18}$$

Combining Eqs. (14), (17) and (18), we have

$$\mathbb{E}[\|\boldsymbol{x}^{(n+1)} - \boldsymbol{x}^*\|_{\boldsymbol{W}}^2|\boldsymbol{y}] \leq 2\rho\mathbb{E}[\|\boldsymbol{x}^{(n)} - \boldsymbol{x}^*\|_{\boldsymbol{W}}^2|\boldsymbol{y}] + (1-2\rho)u \tag{19}$$

Create an auxiliary sequence

$$a^{(n)} = \mathbb{E}[\|\boldsymbol{x}^{(n)} - \boldsymbol{x}^*\|_{\boldsymbol{W}}^2|\boldsymbol{y}] - u \tag{20}$$

Then by Eq. (19),

$$a^{(n+1)} \leq (2\rho)^n a^{(1)} \tag{21}$$

Taking $\limsup_{n\to\infty}$ on both sides of Eq. (21) concludes the proof. □

If $u = 0$, then Eq. (15) implies mean square convergence to the optimal beamformer output. In actuality, u is nonzero, but it tends to be very small. The first term of u measures the distance between the optimal beamformer output and the true speech. According to our empirical study, when the number of channel is sufficient, the optimal beamformer is able to recover the true speech very well, so the first term is small. The second term of u measures the distance between two posterior expectations $\boldsymbol{P}\mathbb{E}[\boldsymbol{s}|\boldsymbol{x}^{(n)}]$ and $\boldsymbol{P}\mathbb{E}[\boldsymbol{s}|\boldsymbol{y}]$. The former is conditional on single-channel noisy speech, and the latter on multiple-channel noisy speech. Considering that the speech sample space is highly structured, and that the noisy speech $\boldsymbol{x}^{(n)}$ is relatively clean already, both posterior expectations should be close to the true speech, and thereby close to each other. In a nutshell, with a small u, the DEEPBEAM prediction is highly accurate. Section 4.4 will verify the convergence behavior of DEEPBEAM empirically.

4. EXPERIMENTS

This section first introduces how the enhancement network is configured and trained, and then presents the results of experiments on both simulated and real-world data. Audio samples can be found in `http://tiny.cc/a1qjoy`.

4.1. Enhancement Network Configurations

The enhancement network hyperparameter configurations follow [19]. It has 4 blocks of 10 dilated convolution layers. There are two post processing layers. The hidden node dimension is 32, and the skip node dimension is 256. The clean speech is quantized into 256 level via μ-law companding, and thus the output dimension is 256. The activation function in the dilated convolutional layers is the gated activation unit; that in the post processing layers is the ReLU function. The output activation is softmax.

The enhancement network is trained on simulated data *only*, which is generated in the same way as in [7]. The speech source, noise source and eight microphones are randomly placed into a randomly sized cubic room. The impulse response from each source to each microphone is generated using the image-source method [21, 22]. The noisy observations are generated according to Eq. (1). The reverberation time is uniformly randomly drawn from [100, 300] ms. The energy ratio between the speech source and noise source, E_r, is uniformly randomly drawn from $[-5, 20]$ dB. The speech content is drawn from VCTK [23], which contains 109 speakers. The noise content contains 90 minutes of audio drawn from [24–26]. The total duration of the training audio is 8 hours. The enhancement network is trained using ADAM optimizer for 400,000 iterations.

4.2. Simulated Data Evaluation

The simulated data for evaluation is generated the same way as the training data, except for two differences. First, the source energy ratio, E_r, is set to four levels, -10 dB, 0 dB, 10 dB, and 20 dB. Second, both the speaker and noise can be either seen or unseen in the training set, leading to four different scenarios to test generalizability. It is worth highlighting that the unseen speaker utterances and unseen noise are both drawn from different corpora from training, TIMIT [27] and FreeSFX [28] respectively. Each utterance is 3 seconds in length. The total length of the dataset is 12 minutes.

DEEPBEAM is compared with GRAB [7], MVDR[1] [29], IVA [5] and the closest channel (CLOSEST), in term of two criteria:

[1] Clean speech is given for voice activity detection.

Table 1: Simulated Data Evaluation Results.

	E_r =	-10	0	10	20
SNR (dB)	DEEPBEAM S1	18.5	22.0	26.5	28.4
	DEEPBEAM S2	17.1	20.3	25.9	27.4
	DEEPBEAM S3	15.3	19.5	24.1	27.6
	DEEPBEAM S4	14.1	19.0	23.1	28.5
	GRAB	2.48	12.5	21.6	25.4
	CLOSEST	-5.13	3.38	14.9	24.8
	MVDR	8.41	12.9	22.6	26.7
	IVA	10.3	13.3	16.8	19.2
DRR (dB)	DEEPBEAM S1	3.45	8.97	11.2	11.5
	DEEPBEAM S2	7.38	11.9	12.6	11.5
	DEEPBEAM S3	5.60	4.85	8.43	9.78
	DEEPBEAM S4	2.11	6.68	7.10	9.31
	GRAB	-0.83	1.70	3.63	3.68
	CLOSEST	8.56	7.32	7.67	8.44
	MVDR	-2.17	-3.47	-3.42	-4.13
	IVA	-8.92	-8.77	-8.81	-8.99

S1: seen speaker, seen noise; S2: seen speaker, unseen noise;
S3: unseen speaker, seen noise; S4: unseen speaker, unseen noise.

- **Signal-to-Noise Ratio (SNR)**: The energy ratio of processed clean speech over processed noise in dB.
- **Direct-to-Reverberant Ratio (DRR)**: the ratio of the energy of direct path speech in the processed output over that of its reverberation in dB. Direct path and reverberation are defined as clean dry speech convolved with the peak portion and tail portion of processed room impulse response. The peak portion is defined as ± 6 ms within the highest peak; the tail portion is defined as ± 6 ms beyond.

Table 1 shows the results. As expected, DEEPBEAM's performance drops from S1, where both noise and speaker are seen during training, to S4, where neither is seen. However, in terms of SNR, even DEEPBEAM S4 significantly outperforms MVDR, which is the benchmark in noise suppression. In terms of DRR, DEEPBEAM matches or surpasses CLOSEST except for -10 dB. GRAB performs poorer than in [7], because each utterance is reduced from 10 seconds to 3 seconds, which is more realistic but challenging. In short, of "cleanness" and "dryness", most algorithms can only achieve one, but DEEPBEAM can achieve *both* with superior performance.

4.3. Real-world Data Evaluation

DEEPBEAM and the baselines are also evaluated on the real-world dataset introduced in [7], which consists of two utterances by two speakers mixed with five types of noises, all recorded in a real conference room using eight randomly positioned microphones. The source energy ratio is set such that the SNR for the closest microphone is 10 dB. The utterance in each scenario is around 1 minute long, so the total length of the dataset is 10 minutes.

Besides SNR, a subjective test similar to [7] is performed on Amazon Mechanical Turk. Each utterance is broken into six sentences. In each test unit, called HIT, a subject is presented with one sentence processed by the five algorithms, and asked to assign an MOS [30] to each of them. Each HIT is assigned to 10 subjects.

Table 2 shows the results. As can be seen, DEEPBEAM outperforms the other algorithms by a large margin. In particular, DEEPBEAM achieves > 4 MOS in some noise types. These results are very impressive, because DEEPBEAM is only trained on simulated data. The real-world data differ significantly from the simulated data in terms of speakers, noise types and recording environment. What's more, some microphones are contaminated by strong electric noise, which is not accounted for in Eq. (1). Still, DEEPBEAM manages to conquer all the unexpected. Neural network used to be vulnerable to unseen scenarios, but DEEPBEAM has now made it robust.

Table 2: Realworld Data Evaluation Results.

Noise Type		N1	N2	N3	N4	N5
SNR (dB)	DEEPBEAM	**20.1**	**20.0**	**16.9**	**19.6**	**18.7**
	GRAB	18.9	17.4	12.4	18.5	17.4
	CLOSEST	10.0	10.0	10.0	10.0	10.0
	MVDR	10.8	16.5	7.72	14.0	13.4
	IVA	11.7	9.74	6.83	12.4	15.9
MOS	DEEPBEAM	**3.83**	**3.72**	**3.63**	**4.09**	**4.20**
	GRAB	3.10	3.06	2.93	3.71	3.45
	CLOSEST	2.74	2.68	3.02	3.55	3.50
	MVDR	2.05	2.40	2.28	2.71	2.62
	IVA	1.73	2.03	1.75	1.78	2.08

N1: cell phone; N2: CombBind machine; N3:paper shuffle;
N4: door slide; N5: footsteps.

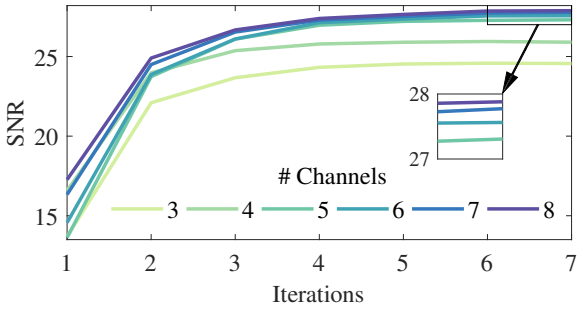

Fig. 2: SNR convergence curves with different numbers of channels.

4.4. Empirical Convergence Analysis

In order to empirically test whether DEEPBEAM has a good convergence property, 10 sets of eight-channel simulated data are generated with the S1 setting and $E_r = 10$. To study different number of channels, in each sub-test, K channels are randomly drawn from each set of data for DEEPBEAM prediction, and the resulting SNR convergence curves of the 10 sets are averaged. K runs from 3 to 8.

Fig. 2 shows all the averaged convergence curves. As can be seen, DEEPBEAM converges well in all the sub-tests, which supports our convergence discussions in section 3.3. Also, the more channels DEEPBEAM has, the higher convergence level it can reach, which shows that DEEPBEAM is able to accommodate different numbers of channels using only one monaural network. We also see that the marginal benefit of having one more channel diminishes.

5. CONCLUSION

We have proposed DEEPBEAM as a solution to multi-channel speech enhancement with ad-hoc sensors. DEEPBEAM combines the complementary beamforming and deep learning techniques, and has exhibited superior performance and generalizability in terms of noise suppression, reverberation cancellation and perceptual quality. DEEPBEAM has made one step closer to resolving the long lasting crux of low perceptual quality and poor generalizability in deep enhancement networks, which demonstrates the power of bridging the signal processing and deep learning areas.

6. REFERENCES

[1] Michael Brandstein and Darren Ward, *Microphone Arrays: Signal Processing Techniques and Applications*, Springer Science & Business Media, 2013.

[2] Shmulik Markovich-Golan, Alexander Bertrand, Marc Moonen, and Sharon Gannot, "Optimal distributed minimum-variance beamforming approaches for speech enhancement in wireless acoustic sensor networks," *Signal Processing*, vol. 107, pp. 4–20, 2015.

[3] Bradford W Gillespie, Henrique S Malvar, and Dinei AF Florêncio, "Speech dereverberation via maximum-kurtosis subband adaptive filtering," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2001, vol. 6, pp. 3701–3704.

[4] Kenichi Kumatani, John McDonough, Barbara Rauch, Dietrich Klakow, Philip N Garner, and Weifeng Li, "Beamforming with a maximum negentropy criterion," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 5, pp. 994–1008, 2009.

[5] Taesu Kim, Hagai T Attias, Soo-Young Lee, and Te-Won Lee, "Blind source separation exploiting higher-order frequency dependencies," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 70–79, 2007.

[6] Daichi Kitamura, Nobutaka Ono, Hiroshi Sawada, Hirokazu Kameoka, and Hiroshi Saruwatari, "Determined blind source separation unifying independent vector analysis and nonnegative matrix factorization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 9, pp. 1626–1641, 2016.

[7] Yang Zhang, Dinei Florêncio, and Mark Hasegawa-Johnson, "Glottal model based speech beamforming for ad-hoc microphone arrays," *INTERSPEECH*, pp. 2675–2679, 2017.

[8] Jitong Chen and Deliang Wang, "Long short-term memory for speaker generalization in supervised speech separation," in *INTERSPEECH*, 2016, pp. 3314–3318.

[9] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis, "Deep learning for monaural speech separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 1562–1566.

[10] Felix Weninger, John R Hershey, Jonathan Le Roux, and Björn Schuller, "Discriminatively trained recurrent neural networks for single-channel speech separation," in *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2014, pp. 577–581.

[11] Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, Dinei Florêncio, and Mark Hasegawa-Johnson, "Speech enhancement using Bayesian Wavenet," *INTERSPEECH*, pp. 2013–2017, 2017.

[12] Dario Rethage, Jordi Pons, and Xavier Serra, "A Wavenet for speech denoising," *arXiv preprint arXiv:1706.07162*, 2017.

[13] Santiago Pascual, Antonio Bonafonte, and Joan Serrà, "SEGAN: Speech enhancement generative adversarial network," *arXiv preprint arXiv:1703.09452*, 2017.

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

[15] Hakan Erdogan, John R Hershey, Shinji Watanabe, Michael I Mandel, and Jonathan Le Roux, "Improved MVDR beamforming using single-channel mask prediction networks.," in *INTERSPEECH*, 2016, pp. 1981–1985.

[16] Xiong Xiao, Shengkui Zhao, Douglas L Jones, Eng Siong Chng, and Haizhou Li, "On time-frequency mask estimation for MVDR beamforming with application in robust speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 3246–3250.

[17] Xueliang Zhang, Zhong-Qiu Wang, and DeLiang Wang, "A speech enhancement algorithm by iterating single-and multi-microphone processing and its application to robust ASR," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 276–280.

[18] Lukas Pfeifenberger, Matthias Zöhrer, and Franz Pernkopf, "DNN-based speech mask estimation for eigenvector beamforming," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 66–70.

[19] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu, "WaveNet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[20] Norbert Wiener, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*, vol. 7, MIT press Cambridge, MA, 1949.

[21] Jont B Allen and David 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, 1979.

[22] Eric A Lehmann and Anders M Johansson, "Diffuse reverberation model for efficient image-source simulation of room impulse responses," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 6, pp. 1429–1439, 2010.

[23] Junichi Yamagishi, "English multi-speaker corpus for CSTR voice cloning toolkit," `http://homepages.inf.ed.ac.uk/jyamagis/page3/page58/page58.html`.

[24] Anurag Kumar and Dinei Florêncio, "Speech enhancement in multiple-noise conditions using deep neural networks," *INTERSPEECH*, 2016.

[25] "Freesound," `https://freesound.org/`, 2015.

[26] Guoning Hu, "100 nonspeech sounds," `http://web.cse.ohio-state.edu/pnl/corpus/HuNonspeech/HuCorpus.html`, 2015.

[27] John S Garofolo, Lori F Lamel, William M Fisher, Jonathon G Fiscus, and David S Pallett, "DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1," *NASA STI/Recon Technical Report n*, vol. 93, 1993.

[28] "FreeSFX," `http://www.freesfx.co.uk/`, 2017.

[29] Lloyd Griffiths and CW Jim, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Transactions on Antennas and Propagation*, vol. 30, no. 1, pp. 27–34, 1982.

[30] Flávio Ribeiro, Dinei Florêncio, Cha Zhang, and Michael Seltzer, "CrowdMOS: An approach for crowdsourcing mean opinion score studies," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2011, pp. 2416–2419.