A Study of Learning Based Beamforming Methods for Speech Recognition

Xiong Xiao[1], *Chenglin Xu*[1], *Zhaofeng Zhang*[2], *Shengkui Zhao*[3], *Sining Sun*[4], *Shinji Watanabe*[5]
Longbiao Wang[6], *Lei Xie*[4], *Douglas L. Jones*[3], *Eng Siong Chng*[1], *Haizhou Li*[7,1]

[1]Nanyang Technological University (NTU), Singapore, [2]Nagaoka University of Technology, Japan,
[3]Advanced Digital Sciences Center, Singapore, [4]Northwestern Polytechnical University, China,
[5]Mitsubishi Electric Research Laboratories, USA, [6]Tianjin University, China,
[7]National University of Singapore, Singapore.

{xiaoxiong, xuchenglin}@ntu.edu.sg, s147002@stn.nagaokaut.ac.jp, shengkui.zhao@adsc.com.sg

Abstract

This paper presents a comparative study of three learning based beamforming methods that are specifically designed for robust speech recognition. The three methods are 1) neural network that predicts beamforming weights from generalized cross correlation (GCC) features; 2) neural network that predicts time-frequency (TF) mask which is used to estimate MVDR (minimum variance distortionless response) beamforming weights; 3) maximum likelihood estimation of beamforming weights to fit enhanced features to clean trained Gaussian mixture model. All three methods operate in frequency domain. They are evaluated on the CHiME-4 benchmarking speech recognition task and compared with traditional delay-and-sum and MVDR beamforming methods on the same speech recognition task. Discussions and future research directions are presented.

1. Introduction

Beamforming is an important approach to improve the performance of automatic speech recognition (ASR) in far field scenarios.. Traditional beamforming methods enhance the speech signals to improve signal level criteria, e.g. the signal-to-noise ratio (SNR) of output signal. As these criteria are not directly related to the ASR's performance measure, tradiitonal methods are usually not optimized for the ASR task.

Recently, several learning based beamforming methods are proposed for the ASR task. By learning based methods, we mean these methods learn from a large amount of training data (single or multi-channel), and apply the learned knowledge at run time to estimate parameters for ASR, e.g. beamforming weights. In one approach [1–3], multi-channel raw waveforms are fed into the neural network acoustic model directly. A temporal convolution layer at the bottom of the network is used to approximate the filter-and-sum beamforming operation. After training, the temporal convolution layer learnes a fixed bank of spatial and temporal filters, each with specific looking directions. We call this approach the spatial filter learning approach. In another approach, beamforming filter weights are predicted by neural networks that are jointly optimized with the acoustic model networks. Deep neural network (DNN) is used to predict beamforming weights in frequency domain from generalized cross correlation (GCC) features [4] or spatial covariance matrix (SCM) features [5]. In [6], long short-term memory (LSTM) networks are used to predict the beamforming weights in the time domain directly which has less number of free parameters than the frequency domain. We call this appraoch the spatial filter prediction approach. While the filter learning approach learns a fixed set of spatial filters, the filter prediction approach predicts spatial filters dynamically from the input data. In another approach, neural networks are used to predict time-frequency (TF) mask that specifies whether a TF bin is dominated by speech or noise. The TF mask is used to help estimating the speech and noise SCMs required by beamforming methods, such as the minimum variance distortionless response (MVDR) [7, 8] and generalized eigenvalue (GEV) [9, 10] beamformers. The mask predicting network can be trained by using ideal masks as target [11–13] or by minimizing the ASR cost function [14]. The filter learning, filter predicting, and mask predicting approaches are called discriminative approach in this paper, as the models are trained to minimize the ASR error.

Besides discriminative methods, there is also learning based beamforming methods based on generative modeling of speech features. In [15, 17], a method called LIMABEAM estimates time or frequency domain filter-and-sum weights to maximize the likelihood of the enhanced feature vectors on clean trained HMM/GMM acoustic model. In the unsupervised implementation, multi-pass decoding is required, where the first pass decoding provides the hypothesized text used to obtain HMM state alignment. Beamforming weights can be estimated iteratively to maximize the likelihood of the enhanced features given the state alignment. It is reported that LIMABEAM outperforms delay-and sum beamforming in several ASR tasks.

Although several learning based methods have been proposed in the past, they are usually implemented by different researchers and evaluated on different ASR tasks. As a result, it is difficult to compare their performance. In this paper, we attempt to study three learning based beamforming methods comparatively, with the implementation in the same toolkit, i.e. SignalGraph [25], and evaluation in the same task, i.e. the CHiME-4 speech recognition task [16]. The three methods include a maximum likelihood (ML) beamforming (a variant of LIMABEAM [15]), the spatial filter weight predicting network [4], and the mask predicting network [14].

2. Learning Based Beamforming Methods

2.1. Spatial Filter Weight Predicting Network

The system diagram of the spatial filter weight predicting network [4] is shown in Fig. 1. On the bottom left of the figure, a network is used to predict the beamforming weights in frequency domain. The weights are then applied on the multichannel inputs to generate enhanced speech, from which features are extracted for acoustic modeling.

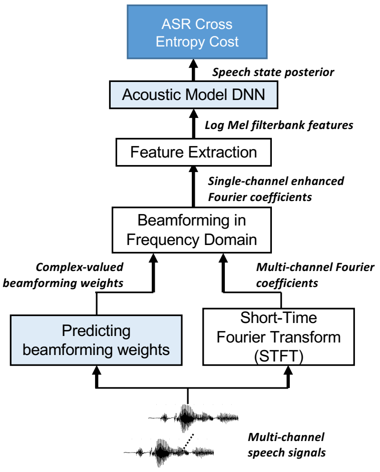
ASR Cross Entropy Cost
Speech state posterior
Acoustic Model DNN
Log Mel filterbank features
Feature Extraction
Single-channel enhanced Fourier coefficients
Beamforming in Frequency Domain
Complex-valued beamforming weights
Multi-channel Fourier coefficients
Predicting beamforming weights
Short-Time Fourier Transform (STFT)
Multi-channel speech signals

Figure 1: Discriminative beamforming weight prediction.

The weight prediction network and the acoustic model network are jointly optimized using the ASR cost function. The weight predicting network is initialized by learning from a delay-and-sum filter on simulated data. Specifically, if the true time difference of arrival (TDOA) of the different channels are known, which is the case for simulated data, we can use the ideal delay-and-sum filter weights as the target for the weight predicting network to learn. The network predicts the real and imaginary of the ideal weights independently. Mean square error (MSE) between the ideal weights and predicted weights is used as the cost function in initialization. After the initialization, the weight predicting network is jointly refined with the acoustic model using back propagation and ASR cost.

The details of the weight predicting network is illustrated in Fig. 2. From the waveforms, we extract feature vectors from GCC function between two channels [18] for every 0.2s long frame with 0.1s shift. The GCC feature vectors encode the phase information of channels and the features extracted from all channel pairs are concatenated to form a single feature vector for each frame. For the CHiME-4 data [16], the dimensionality of the GCC feature vector is 27 for each channel pair. This is because the maximum TDOA is less than 13 samples for the array geometry used in CHiME-4 and 16kHz sampling rate. The bottom right of Fig. 2 shows example GCC features. As different direction of arrival (DOA) angles have different GCC patterns, the GCC features contain information for DNN to determine spatial direction of the source and also TDOA [19]. In this work, a DNN is used to map the GCC features to the beamforming weights in frequency domain. For stable estimation of weights, we take the mean of predicted weight vectors of all frames for each sentence.

While the array geometry is assumed to be fixed in [4], in the two channel track of the CHiME-4 benchmarking task, the geometry of the array depends on the distance bewteen the two microphones randomly selected from a 6-microphone array. We will test whether one single weight predicting network is able to cover several array geometries.

2.2. Time Frequency Mask Predicting Network

The TF mask predicting network is illustrated in Fig. 3. The log power spectra of input signals are mean normalized on an utterance basis and used as features for mask prediction. The mask prediction is carried out for each channel independently, but shares the same LSTM mask predictor. For each channel, two TF masks are predicted by the LSTM network, one speech mask that specifies whether a TF bin is speech dominated and one noise mask. We call this splitted mask. We can also force the speech and noise masks to sum to 1 for each TF bin. This can be implemented by only predicting the speech mask and obtain the noise mask by 1-speech mask.

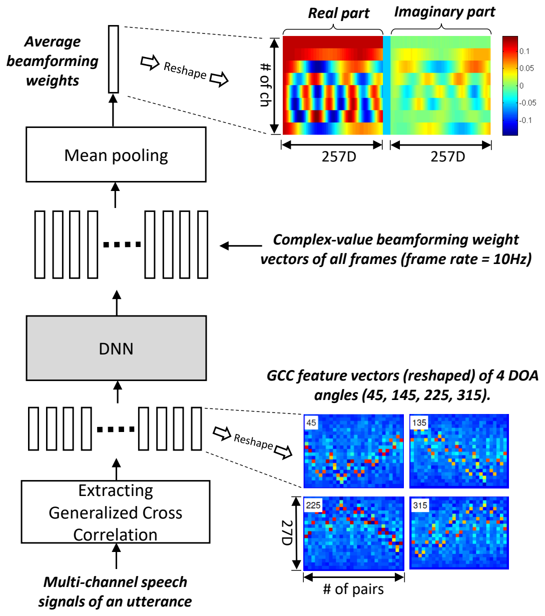
Average beamforming weights
Reshape
Real part
Imaginary part
# of ch
257D
257D
0.1
0.05
0
-0.05
-0.1
Mean pooling
Complex-value beamforming weight vectors of all frames (frame rate = 10Hz)
DNN
GCC feature vectors (reshaped) of 4 DOA angles (45, 145, 225, 315).
Reshape
45
135
225
315
27D
# of pairs
Extracting Generalized Cross Correlation
Multi-channel speech signals of an utterance

Figure 2: Details of weight predicting network. The size of the GCC feature matrix (bottom right) depends on the number of unique channel pairs.

The LSTM network contains one hidden layer, whose activation vector is projected to noise and speech mask vectors by using two independent projection layers. The sigmoid activation function is used for projection layers to ensure that the predicted masks will have value between 0 and 1. For both noise and speech masks, pooling is used to reduce the set of masks of all channels to a single mask. Four types of pooling functions are compared, including mean, median, min, and max. Note that during training, we only uses one channel (the first channel) to predict the mask, and hence pooling is not necessary. Only at testing, we may estimate the masks for all channels and use pooling.

Given the mask, the MVDR beamforming weights can be determined as follows [20],

$$\mathbf{w}(f) = \frac{\Phi_{nn}^{-1}(f)\Phi_{ss}(f)\mathbf{u}}{\mathrm{Tr}[\Phi_{nn}^{-1}(f)\Phi_{yy}(f)]} \tag{1}$$

where $\mathbf{u}$ is a vector with the element for reference channel being 1 and all others being 0. Tr[$\cdot$] denotes trace of a matrix. Φ_{nn} and Φ_{ss} are the noise and speech SCMs estimated as

$$\Phi_{ss}(f) = \frac{\sum_{t=1}^{T} \hat{m}_t^s(f)\mathbf{y}_t(f)\mathbf{y}_t^H(f)}{\sum_{t=1}^{T} \hat{m}_t^s(f)} \tag{2}$$

$$\Phi_{nn}(f) = \frac{\sum_{t=1}^{T} \hat{m}_t^n(f)\mathbf{y}_t(f)\mathbf{y}_t^H(f)}{\sum_{t=1}^{T} \hat{m}_t^n(f)} \tag{3}$$

where $\hat{m}_t^s$ and $\hat{m}_t^n$ are the estimated mask values at frame t and frequency f for speech and noise, respectively. $\mathbf{y}_t(f)$ is the observed signal in frequency domain.

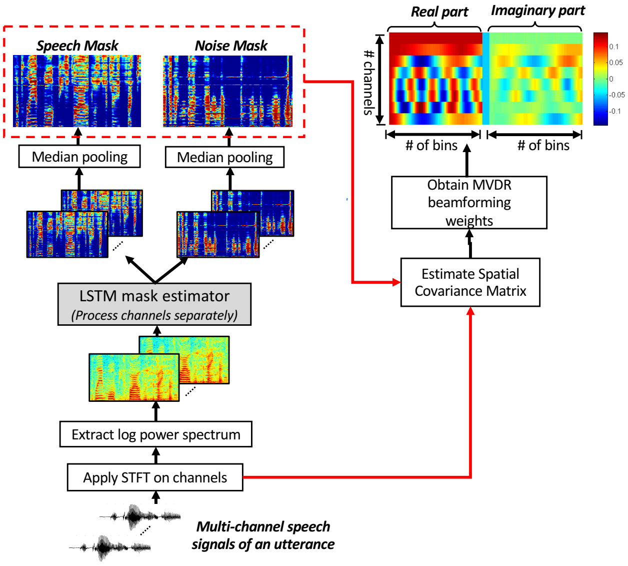
Speech Mask
Noise Mask
Median pooling
Median pooling
LSTM mask estimator
(Process channels separately)
Extract log power spectrum
Apply STFT on channels
Multi-channel speech signals of an utterance
Real part
Imaginary part
# channels
# of bins
# of bins
0.1
0.05
0
-0.05
-0.1
Obtain MVDR beamforming weights
Estimate Spatial Covariance Matrix

Figure 3: Details of mask predicting network and the estimating of MVDR weights.

The LSTM mask predicting network is initialized by learning from ideal binary mask (IBM) of speech. For simulated data, we can obtain the oracle local SNR for each TF bin. The speech IBM is set to 1 if the local SNR is larger than 0dB and vice versa. Then the LSTM network is trained to predict the speech IBM from single channel log power spectrum by minimizing the mean square error (MSE) between the predicted mask and the IBM. Once initialized, the network in Fig. 3 is used to replace the weight predicting module in Fig. 1, and the LSTM mask predictor is jointly refined with the acoustic model to minimize ASR cost function. The noise projection layer's weights and bias can be initialized as the negative weights and bias of the speech projection layer so the sum of noise and speech masks sum to one for each TF bin. Note that, after joint training, the noise and speech masks usually do not sum to 1.

2.3. Maximum Likelihood Spatial Filter Estimation

We also investigate a modified version of the LIMABEAM [15]. The beamforming parameters are estimated as follows:

$$\begin{aligned}\hat{\mathbf{W}}_{\text{ML}} &= \arg\max_{\mathbf{W}} \frac{1}{T}\log p\left(\mathbf{O}(\mathbf{W});\Theta\right) \\ &\quad + \frac{1}{2}\log\left|\Sigma_{\mathbf{O}}(\mathbf{W})\right| - \frac{\alpha}{2}|\mathbf{W}-\mathbf{W}_0|_F^2 \qquad (4)\end{aligned}$$

where $\mathbf{O}(\mathbf{W})$ is the enhanced feature vectors and is a function of the beamforming weights. Θ is the parameters of the acoustic model and T is the number of frames in the test utterance. The first term in (4) measures the likelihood of the enhanced features evaluated on the acoustic model, which can be an HMM/GMM or GMM. When the acoustic model represents clean features' distribution, it is a reasonable assumption that the higher the likelihood is, the higher the quality of the enhanced features [15]. The second and third terms are added in this work to the orginal LIMABEAM. The second term is the log determinant of the covariance matrix of the enhanced features (also a function of weights) and it acounts for the nonlinear transformation of the feature space [21, 22] due to the beamforming operation. The third term is the Frobenius norm between the weight matrix and its initial values. This term is used to impose L2 norm regularization on the parameters to prevent overfitting. The modified LIMABEAM is called maximum likelihood beamforming (MLBF) in this paper and illustrated in Fig. 4. The three terms in (4) are represented as three cost function nodes in blue color.

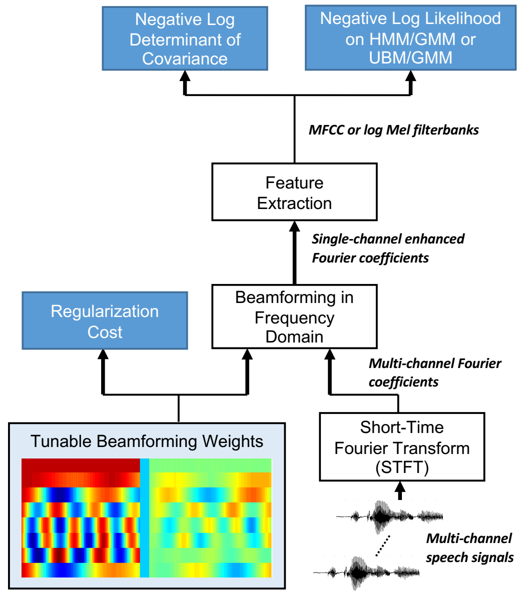
Negative Log Determinant of Covariance
Negative Log Likelihood on HMM/GMM or UBM/GMM
MFCC or log Mel filterbanks
Feature Extraction
Single-channel enhanced Fourier coefficients
Regularization Cost
Beamforming in Frequency Domain
Multi-channel Fourier coefficients
Tunable Beamforming Weights
Short-Time Fourier Transform (STFT)
Multi-channel speech signals

Figure 4: System diagram of maximum likelihood based beamforming weight estimation.

Instead of using HMM/GMM as the acoustic model, we use a single GMM to model the distribution of the clean MFCC features. The advantage of using GMM is that there is no need to perform one extra pass of decoding to obtain the HMM state alignment. However, it is possible that performance will degrade compared with using HMM/GMM.

There are two ways to represent the frequency domain beamforming weights. In the first way, we treat the real and imaginary parts of the weights as free parameters, hence there are $2IK$ free parameters, where I and K are the number of channels and frequency bins, respectively. In the second way, the weights are represented as follows

$$w_i(f) = g_i(f)\exp(j2\pi f\frac{\tau_i}{f_s}) \qquad (5)$$

where $w_i(f)$ and $g_i(f)$ are the weight and gain for channel i at frequency f, respectively. f_s is the sampling frequency, $\frac{\tau_i}{f_s}$ is the TDOA of channel i and assumed to be frequency independent. The first channel is always selected as the reference channel and its TDOA is set to 0. Hence, there are totally $I-1$ free parameters from TDOA, and IK free parameters from gain.

3. Experiments

3.1. Settings

We evaluate the three learning based beamforming methods on the 2-channel and 6-channel tracks of the CHiME-4 task [16]. The baseline DNN acoustic model is used, except that the fMLLR [23] features are replaced by 40D log Mel filterbank features, due to the fact that fMLLR needs to be dynamically estimated and makes it difficult to conduct joint training of beamforming networks and acoustic model. No pre-emphasis or

DC removal is applied. Delta and acceleration features are appended and then 11 frames of feature vectors are concatenated to form the input for the DNN acoustic model. Two types of DNN acoustic model is used, one is trained from the fifth channel (called ch5 model, channel 5 is the single best channel in the array), while the other is trained from all the 6 channels (called chall model). The baseline trigram language model is used if not otherwise specified. Speech recognition is performed using the sequentially trained DNN acoustic model, i.e. the state-level minimum Bayes risk (SMBR) model [24].

All the three learning based beamforming methods are implemented in SignalGraph, a Matlab based toolkit for applying deep learning to signal processing [25]. The beamforming weight predicting network uses either a 3 hidden layer DNN or an 1 hidden layer LSTM, both using 1024 hidden nodes. The input to the network is 27D (1 microphone pair) for 2-channel case and 405D (15 microphone pairs) for 6-channel case. The output dimension is 257x2x2=1028 for 2-channel case and 257x2x6=3084 for 6-channel case, as 257 complex numbers (512 FFT length) need to be predicted for each channel. The network is initilized on 71680 simulated sentences (10 times of the official simulated training data) generated by ourselves using the provided simulation tool. After initilization, the network is refined together with the ch5 acoustic model (trained with cross entropy, or CE, criterion) by using the frame level CE cost function. As we will use the SMBR model for decoding, we fixed the acoustic model during joint training to prevent the acoustic feature space from drifting too much from the one used to train the SMBR model.

The mask predicting network is implemented by using a one hidden layer LSTM containing 1024 memory cells. The memory cells' outputs are projected to noise and speech masks by using two 1024 to 257 affine transorms in projection layers. The network is initialized on the 71680 simulated sentences (same as the data used to initialize the weight predicting network). After initialization, the network is jointly refined with the ch5 acoustic model in the same way as the joint training of weight predicting network.

The GMM used in the ML beamforming is trained from the close talk version of the 1680 real training sentences. The GMM uses 39D MFCC features and diagonal covariance matrix, and contains either 512 or 1024 Gaussians. The beamforming parameters are estimated iteratively using the expectation-maximization (EM) algorithm [26]. At most 3 EM iterations are used. At the E step of each EM iteration, the posteriors of the Gaussians are re-estimated using the enhanced features. At the M step, the beamforming parameters are re-estimated given the updated Gaussian posteriors by using the L-BFGS algorithm [27]. Due to the iterative nature of the EM algorithm, the real time factor is usually 1-5 for the whole estimation process for each sentence, which is much slower than the other two methods. When L2 regualization is used, it is used on all parameters except for the TDOAs.

3.2. Results of Beamforming Weight Predicting Network

The performance of weight predicting network is shown in Table 1. Row 2 and 3 show the results of MSE training in which the neural networks learn from the ideal unweighted delay-and-sum beamforming and simulated data. Comparing with the weighted delay-and-sum implemented in BeamformIt [28] (row 1), the neural networks perform slightly worse in overall, and LSTM performs slightly better than DNN. Row 4 and 5 show the results of CE training in which the neural networks are refined using the ASR cross entropy cost function on the official training data. For the two channel case, moderate imporvement is obtained by CE training over MSE training for DNN model (17.2% versas 16.5% on real data), while the results of LSTM model is mixed which could be due to overfitting.

Table 1: Recognition word error rate (WER %) obtained by weight predicting network on the CHiME-4 task. "DNN*" and "LSTM*" refer to CE refined model only using 1680 real recorded training sentences. The 5 channel case does not include the second channel. "DS" refers to BeamformIt.

Row	Model	Cost	6 channels		5 channels		2 channels	
			Eval		Eval		Eval	
			Real	Simu	Real	Simu	Real	Simu
1	DS	-	14.8	12.6	13.6	14.2	17.2	18.2
2	DNN	MSE	15.8	13.8	13.5	16.5	17.2	18.5
3	LSTM		14.7	13.4	12.9	14.9	16.5	18.3
4	DNN	CE	15.0	11.4	15.9	11.6	16.5	16.8
5	LSTM		14.6	11.5	14.7	11.6	16.8	17.3
6	DNN*		13.6	16.0	-			
7	LSTM*		14.6	14.5				

For 6 channel case, CE training obtains significant improvement overal MSE training on simulated data, but not on real data. One possible reason is that the target signal's gain is not equal at different channels for real data. Sometimes, channels may even totally fail to receive signals. The neural networks takes GCC features as input where the gain information is largely removed. Hence, the neural networks are unable to predict the gains of channels properly. To investigate the issue, we conducted two more experiments. First, we train the neural networks without using the second channel (5 channel case) which is known to have poor signal quality for real data. This leads to better performance of MSE trained models (row 2 and 3) on real data (as the bad channel is removed), but worse results on simulated data (as a good channel is removed). This pattern is also observed for the BeamformIt results (row 1). However, the CE trained models still perform poorly on real data. Second, we refine the neural networks only on 1680 real sentences of the official training set (row 6 and 7 of the 6 channel case). WER on real data is improved for DNN model, however, WER on simulated data gets much worse for both models. This shows that the CE training should use data similar to the eval data.

In summary, we found that the weight predicting framework [4] do not consistently outperform BeamformIt on CHiME-4, while it does outperform BeamformIt significantly on the AMI corpus. We hypothesize that this is because the AMI is a far field scenario and the gains of the channels are similar, while CHiME-4 is a near field scenario where the gains of channels could be very different. As the network only uses GCC as input, it is not able to estimate the channel gains proporly.

3.3. Results of Mask Predicting Network

The performance obtained with mask predicting network and MVDR beamforming is shown in Table 3 for 2 channel case and Table 2 for 6 channel case. Let's go through the 6 channel case as the results of 2 channel case will be similar. A conventional MVDR beamforming [29] with TDOA tracking, frequency dependent channel gain estimation, and noise estimation using 0.5s noises prior to the test utterance obtains a WER of 12.0% on the real eval data (row 3). By comparison, the MVDR using masks predicted by IBM-initialized LSTM produces a WER of 12.8% (row 4). By using ASR cost function to

Table 2: Recognition word error rate (WER %) obtained by mask predicting network on the CHiME-4 6-channel track. "Split Mask" specifies whether we estimate speech and noise masks separately. LM: "3" means trigram, "5" means 5-gram, while "R" is RNN LM rescoring.

Row	Settings						Dev		Eval	
	#ch for mask	ASR cost	Split Mask	Pooling	#Pass	LM	Real	Simu	Real	Simu
1	1-channel track					3	12.4	14.8	21.6	22.0
2	Delay-and-sum (BeamformIt)						8.2	9.4	13.6	14.2
3	Traditional MVDR						7.6	6.6	12.0	8.2
4	First channel	No	No	No	1		8.3	7.1	12.8	19.5
5		Yes			1		7.3	6.4	10.9	15.2
6					3		6.4	6.1	9.4	11.1
7			Yes		1		6.5	6.1	10.1	11.9
8					3		6.1	6.0	9.0	9.9
9	Estimate masks for all 6 channels, then pool the masks			max	1		6.6	6.0	10.2	10.0
10				min	1		6.6	6.0	10.3	8.9
11				mean	1		6.4	5.9	9.8	9.2
12				median	1		6.2	6.0	9.5	8.9
13					3		6.1	5.9	8.9	9.6
14					3	5	4.8	4.9	7.4	7.9
15					3	R	4.1	4.3	6.3	6.9

fine tune the LSTM mask predictor, the WER reduces to 10.9% (row 5). The reason for poor performance on simulated eval data is that the first channel of this data set has much lower SNR than other channels and the network predicts the mask from the first channel only. In overall, the results show the effectiveness of using ASR cost function to fine tune mask predictor.

We investigated several approaches to further improve the performance on mask based MVDR. The first is to use multiple passes of mask estimation and beamforming. Specifically, the mask estimation (using enhanced speech) and beamforming can be performed alternately until converge. In row 6, applying the mask estimation and beamforming 3 times is found to reduce WER further to 9.4% (3 passes) from 10.9% (1 pass). The second approach we studied is the splitted mask, i.e. predicting the speech and noise masks independently. Comparing row 7 to row 5, using splitted masks consistently outperforms using unsplitted mask. Lastly, we investigated the use of mask pooling. From row 9 onwards, the masks of all the 6 channels are estimated and pooled. It is observed that median pooling produces the best performance, which agrees with findings in [11]. For the 2 channel case, no pooling is used. We investigated the mask predicting using concatenation of two channels' log power spectra. Comparing row 7 and 8 of Table 3, concatenated input outperforms the single channel input significantly. By combining all the methods together, we obtain the best WER on the real eval data in row 13 in Table 2, with a WER of 8.9%. This represents a 3.1% absolute WER reduction compared to conventional MVDR.

3.4. Results of Maximum Likelihood Weight Estimation

The performance of MLBF on the 6 channel track is shown in Table 4. Row 1 shows that by only estimating 5 TDOAs of channel 2-6 using the MLBF, a WER of 17.1% is obtained, which is significantly lower than 1 channel case (21.6%) shown in row 1 of Table 2. By only using TDOAs, the signals are aligned and added together, similar to unweighted delay-and-sum beamforming. If frequency dependent gains are also estimated and L2 norm is tuned, the WER can be further reduced to 16.1% (row 3). We also tried to use frequency independent gains (row 4), i.e. only uses one global gain for each channel, the same WER of 16.1% WER is obtained. We improve the frequency dependent gain estimation by using frequency independent gains as the initial gains $\mathbf{W}_0$ in equation (4). The L2 regularization ensures that the frequency dependent gains are not too far from the frequent independent gains. Results in row 5 show that this way of initialization and L2 regularization reduce the WER significantly to 14.5%.

Table 3: Recognition word error rate (WER %) obtained by mask predicting network on the CHiME-4 2-channel track.

Row	Settings						Dev		Eval	
	#ch for mask	ASR cost	Split Mask	#Pass	AM	LM	Real	Simu	Real	Simu
1	Not applicable for BeamformIt				ch5	3	10.9	12.4	20.4	19.0
2							11.9	13.1	20.8	20.2
3					chall		10.1	11.7	17.2	18.2
7	First channel	No	No	1			9.8	10.4	16.6	17.0
8	ch 1&2			1			9.2	10.2	15.5	14.9
9	First channel	Yes		1			9.4	10.1	15.7	16.2
10				3			9.1	10.0	15.0	15.0
11			Yes	1			8.9	10.0	15.2	15.3
12				3			8.8	9.9	14.5	14.3
13				3			8.4	9.5	14.4	14.2
14				3		5	7.0	8.1	12.3	12.1
15				3		R	6.1	7.1	10.8	10.7

Table 4: Recognition WER (%) obtained by MLBF on the CHiME-4 6-channel track.

Row	Settings				Eval	
	Parameters	Init.	Gain	#Gau.	Real	Simu
1	TDOA + Gain	No	None	512	17.1	17.6
2		No	Freq. Dependent	512	16.2	14.6
3		No	Freq. Dependent	1024	16.1	14.5
4		No	Freq. Independent	1024	16.1	14.5
5		Row 4	Freq. Dependent	1024	14.5	12.2
6	Real + Imag.	MVDR using mask prediction			9.5	8.9
7		Row 6	-	1024	9.2	8.3

We initialize the real and imaginary parts of the weights with the weights generated by the mask based MVDR (shown in row 6, also row 12 of Table 2). L2 regularization is applied to prevent big deviations of the weights from the initial weights. Results show that the WERs on both simulated and real data are improved moderately. It is worth noting that there is a big gap in performance between row 5 and 7. This could be due to different parameterization of weights and/or the MLBF may easily stuck in a local minimum of cost function.

3.5. Discussions and Future Works

In this paper, we conducted a comparative study of three learning based beamforming methods for far field speech recognition. We found that the MVDR beamformer using LSTM predicted time frequency masks perform the best, while the beamforming filter weight predicting network and MLBF also improve the ASR performance significantly compared to the single channel baseline. In terms of computational cost, the weight predicting network is the most efficient, followed by mask predicting network. Both of these networks are faster than real time. The MLBF is the slowest due to iterative weight optimization at run time.

The better performance of MVDR formulation could be due to that the noise information is important in this task. While the mask based MVDR explicitly makes use of noise estimation, the weight predicting network does not use noise information since only the phase-carrying GCC features are used as input. Although the MLBF has access to the raw noisy data in frequency domain, it does not find good weight solution similar to the MVDR's, possibly due to the local minimum problem of EM algorithm. Hence, the future works could be done to add noise information explicitly to these two types of methods. Another observation is that for near field scenario, it is important to estimate the channel gains as shown in the results of MLBF. The weight predicting network may be improved by explicitly predicting the gains and also use MVDR weights as the supervision during initialization. The MLBF could be integrated with traditional methods. For example, besides maximizing likelihood, one can also maximize the output SNR so more supervision information is used and better solution could be obtained.

4. Acknowledgments

Xiong Xiao and Chenglin Xu are supported by DSO funded project MAISON DSOCL14045.

5. References

[1] Y. Hoshen, R. J. Weiss, and K. W. Wilson, "Speech acoustic modeling from row multichannel waveforms," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 4624–4628, 2015.

[2] T. N. Sainath, R. J. Weiss, K. W. Wilson, A. Narayanan, M. Bacchiani, and Andrew, "Speaker location and microphone spacing invariant acoustic modeling from raw multichannel waveforms," *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 30–36, 2015.

[3] T. N. Sainath, R. J. Weiss, K. W. Wilson, A. Narayanan, and M. Bacchiani, "Factored spatial and spectral multichannel raw waveform CLDNNs," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2016-May, pp. 5075–5079, 2016.

[4] X. Xiao, S. Watanabe, H. Erdogan, L. Lu, J. Hershey, M. L. Seltzer, G. Chen, Y. Zhang, M. Mandel, and D. Yu, "Deep beamforming networks for multi-channel speech recognition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2016-May, 2016, pp. 5745–5749.

[5] X. Xiao, S. Watanabe, E. S. Chng, and H. Li, "Beamforming Networks Using Spatial Covariance Features for Far-field Speech Recognition," *accepted by APSIPA ASC*, 2016.

[6] B. Li, T. N. Sainath, R. J. Weiss, K. W. Wilson, and M. Bacchiani, "Neural Network Adaptive Beamforming for Robust Multichannel Speech Recognition," *INTERSPEECH*, 2016.

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

[8] J. Capon, "High-resolution frequency-wavenumber spectrum analysis," *Proceedings of the IEEE*, vol. 57, no. 8, pp. 1408–1418, 1969.

[9] E. Warsitz and R. Haeb-Umbach, "Blind acoustic beamforming based on generalized eigenvalue decomposition," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 5, pp. 1529–1539, 2007.

[10] A. Krueger, E. Warsitz, and R. Haeb-Umbach, "Speech enhancement with a GSC-like structure employing eigenvector-based transfer function ratios estimation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 1, pp. 206–219, 2011.

[11] J. Heymann, L. Drude, A. Chinaev, and R. Haeb-Umbach, "BLSTM supported GEV beamformer front-end for the 3RD CHiME challenge," *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 444–451, 2015.

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

[13] H. Erdogan, J. Hershey, S. Watanabe, and M. Mandel, "Improved MVDR Beamforming using Single-channel Mask Prediction Networks," *INTERSPEECH*, 2016. [Online]. Available: http://www.merl.com/publications/docs/TR2016-072.pdf

[14] X. Xiao, S. Zhao, D. L. Jones, E. S. Chng, and H. Li, "On Time-Frequency Mask Estimation for MVDR Beamforming With Application in Robust Speech Recognition," *submitted to ICASSP 2017*.

[15] M. L. Seltzer, B. Raj, and R. 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.

[16] E. Vincent, S. Watanabe, A. A. Nugraha, J. Barker, and R. Marxer, "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," *to appear in Computer Speech and Language*.

[17] M. L. Seltzer, "Microphone Array Processing for Robust Speech Recognition," *PhD thesis, CMU*, no. July, p. 163, 2003.

[18] C. Knapp and G. C. Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 24, no. 4, pp. 320–327, 1976.

[19] X. Xiao, S. Zhao, X. Zhong, D. L. Jones, E. S. Chng, and H. Li, "A learning-based approach to direction of arrival estimation in noisy and reverberant environments," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2015-Augus, pp. 2814–2818, 2015.

[20] M. Souden, J. Benesty, and S. Affes, "On optimal frequency-domain multichannel linear filtering for noise reduction," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, no. 2, pp. 260–276, 2010.

[21] D. H. H. Nguyen, X. Xiao, E. S. Chng, and H. Li, "Feature Adaptation Using Linear Spectro-Temporal Transform for Robust Speech Recognition," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 24, no. 6, pp. 1006–1019, 2016.

[22] Z. Zhang, X. Xiao, L. Wang, J. Dang, M. Iwahashi, E. S. Chng, and H. Li, "Multi-channel feature adaptation for robust speech recognition," *ISCSLP*, 2016.

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

[24] K. Veselý, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks," *INTERSPEECH*, pp. 2345–2349, 2013.

[25] X. Xiao, "SignalGraph: a deep learning toolkit for signal processing," 2016. [Online]. Available: https://github.com/singaxiong/SignalGraph

[26] D. Dempster, A.P. and Laird, N.M. and Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[27] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Mathematical Programming*, vol. 45, no. 1-3, pp. 503–528, 1989.

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

[29] S. Zhao, X. Xiao, Z. Zhang, T. N. T. Nguyen, X. Zhong, B. Ren, L. Wang, D. L. Jones, E. S. Chng, and H. Li, "Robust speech recognition using beamforming with adaptive microphone gains and multichannel noise reduction," *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 460–467, 2016.