DEEP UNFOLDING FOR MULTICHANNEL SOURCE SEPARATION

Scott Wisdom[1]*, John Hershey*[2]*, Jonathan Le Roux*[2]*, and Shinji Watanabe*[2]

[1]Department of Electrical Engineering, University of Washington, Seattle, WA, USA
[2]Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA

ABSTRACT

Deep unfolding has recently been proposed to derive novel deep network architectures from model-based approaches. In this paper, we consider its application to multichannel source separation. We unfold a multichannel Gaussian mixture model (MCGMM), resulting in a deep MCGMM computational network that directly processes complex-valued frequency-domain multichannel audio and has an architecture defined explicitly by a generative model, thus combining the advantages of deep networks and model-based approaches. We further extend the deep MCGMM by modeling the GMM states using an MRF, whose unfolded mean-field inference updates add dynamics across layers. Experiments on source separation for multichannel mixtures of two simultaneous speakers shows that the deep MCGMM leads to improved performance with respect to the original MCGMM model.

Index Terms— Deep unfolding, source separation, multichannel GMM, Markov random field

1. INTRODUCTION AND RELATION TO PRIOR WORK

Exploiting multiple microphones can greatly improve speech enhancement and recognition performance in the presence of noise, other speakers, and reverberation. Multiple microphones enable the use of beamforming [1], multichannel filtering [2] and clustering of spatial features [3, 4]. Multichannel versions of single-channel algorithms have also been proposed, such as multichannel extensions of nonnegative matrix factorization (NMF) [5–7].

Speech acoustic models have previously been used to optimize microphone array beamformers for example, by maximizing likelihood [8]. Recently, however, deep neural network (DNN) speech models have been very successful for single-channel speech enhancement [9–12] and recognition [13, 14]. Their combination with multi-channel methods is not as straight-forward due to the absence of a likelihood function, but there have been a few steps in this direction. Swietojanski et al. [15] proposed a convolutional neural network (CNN) architecture for ASR using multichannel audio, where different microphone channels were pooled together. Hoshen et al. [16] used a CNN-DNN for acoustic modeling on raw time-domain multichannel audio. Nugraha et al. [17] achieved improved source separation for two-channel music recordings using alternating ReLU layers and channel estimation. However, though DNN-based methods can be effective, they require empirical exploration to determine the best network architecture. Furthermore, it is difficult to directly incorporate domain knowledge into generic networks.

Deep unfolding is a method that can incorporate advantages of both neural networks and model-based methods [18]. The basic idea is that any iterative inference algorithm for a generative model which is run for K iterations can be "unfolded" into a K-layer computational network. The architecture and activation functions of the intra- and inter-layer connections are completely defined by the original generative model inference algorithm. Once the network is unfolded, the parameters within layers can be discriminatively trained using labeled data, just as the parameters of a DNN are discriminatively trained. Deep unfolding has been shown to improve single-channel source separation by unfolding iterative NMF multiplicative updates [18, 19].

Other attempts have been made to combine deep networks with generative models. Varani et al. [20] proposed a DNN where the last layer is a GMM. The GMM parameters are discriminatively trained jointly with the DNN parameters for ASR. Hoshen et al.'s approach [16] attempts to mimic the usual feature extraction pipeline in ASR. However, all these methods suffer the same drawback: the optimal network architectures can only be discovered by heuristic experimentation, and it is difficult to directly incorporate insight from domain knowledge.

In this paper, we consider deep unfolding for multichannel source separation. We combine an existing model originally proposed by Attias [21] with a Markov random field (MRF) and show how unfolding inference in this model results in improved source separation performance for multichannel mixtures of two simultaneous speakers. The resulting deep MCGMM computational network directly processes complex-valued frequency-domain multichannel audio and has an architecture defined explicitly by a generative model, thus combining advantages of deep networks and model-based approaches.

2. SOURCE SEPARATION USING MULTICHANNEL GMM

We assume that J acoustic sources x^j are recorded by I microphones. Let $Y_{f,t} \in \mathbb{C}^I$ be the complex-valued STFT coefficients of the I microphones at frame $t \in \{1..T\}$ and frequency $f \in \{1..F\}$. The STFT window and FFT lengths are both taken to be $N_w = 2(F-1)$. The ith microphone signal is given by

$$Y^i_{f,t} = \sum_j B^{i,j}_f X^j_{f,t} + V^i_{f,t}, \tag{1}$$

where $X^j_{f,t}$ is the STFT coefficient of the jth source, $V^i_{f,t}$ is additive, zero-mean, circular, complex-valued Gaussian noise, and $B^{i,j}_f$ is the value at frequency f of the FFT of the channel $b^{i,j}_{1:N_c}$ from source j to microphone i, where we assume a narrowband channel model: that is, the channel impulse response $b^{i,j}_{1:N_c}$ is shorter than the analysis window length: $N_c \leq N_w$. By using a narrowband assumption, the effect of the channel is a complex-valued gain $B^{i,j}_f$ in each frequency bin f for each microphone-source pair (i, j).

This work was done while S. Wisdom was an intern at MERL and at JSALT 2015 in UW, Seattle, which was supported by JHU via grants from NSF (IIS), Google, Microsoft, Amazon, Mitsubishi Electric, and MERL.

In this paper, we model each source as a zero-mean, circular, complex-valued Gaussian, the variance of which is dependent on the source state. Attias [21] originally proposed such a model, using a multinomial distribution for the source states, leading to a GMM for each source. The sources are mapped onto an array of microphones via linear time-invariant channel models B_f. The model is formulated as follows: for each time t, a source's state is given by $z^j_t \in \{1..Z\}$, which controls a pattern of variances across frequency, given by $1/\gamma^{j,z}_f$, where $\gamma^{j,z}_f$ are state-dependent precisions. That is,

$$\left(X^j_{f,t}|z^j_t = z\right) \sim \mathcal{N}_{\mathbb{C}}(0, 1/\gamma^{j,z}_f). \tag{2}$$

Each channel is assumed to have a small amount of additive, independent, zero-mean, circular, complex-valued Gaussian noise. The observations are thus distributed as

$$\left(Y^i_{f,t}|X^{1:J}_{f,t}\right) \sim \mathcal{N}_{\mathbb{C}}\Big(\sum_j B^{i,j}_f X^j_{f,t}, 1/\psi^i_f\Big), \tag{3}$$

where ψ^i_f is a precision for the additive sensor noise $V^i_{f,t}$. The states z^j for source j have priors $\pi^{j,z} := p(z^j = z)$, where z is a value in $\{1..Z\}$. The channel model B_f is here considered a parameter.

Exact inference in this model is intractable because the E-step requires summing over an exponential number of terms ($\mathcal{O}\left(Z^J\right)$) in the marginalization over states. However, an approximate variational algorithm [22] can be derived, which was done by Attias [21]. The approximate inference algorithm uses the variational approximation

$$q(X^{1:J}_{f,t}, z^{1:J}_t) = \Bigg[\prod_f \prod_j q(X^j_{t,f}|z^j_t)\Bigg]\Bigg[\prod_j q(z^j_t)\Bigg], \tag{4}$$

where $q(X^j_{f,t}|z^j_{f,t} = z) = \mathcal{N}_{\mathbb{C}}\big(X^j_{f,t}; \bar{\mu}^{j,z}_{f,t}, 1/\bar{\gamma}^{j,z}_f\big)$, and $q(z^j_{f,t} = z) = \bar{\pi}^{j,z}_t$. In this variational approximation, $\bar{\mu}^{j,z}_{f,t}$ is the state-dependent variational posterior mean and $\bar{\gamma}^{j,z}_f$ is the state-dependent variational posterior precision of source j at time-frequency (t, f). The variational updates are given in Attias [21, eq. (10)-(15)].

3. DEEP UNFOLDING OF GENERATIVE MODELS

Here we apply the deep unfolding framework in the context of the MCGMM. A key difference with the application considered in previous deep unfolding work [18, 19] is that several updates in the complex-valued unfolded MCGMM involve non-holomorphic functions of complex-valued variables. Because of these non-holomorphic functions, the usual complex gradient is not sufficient to perform gradient descent. Also, gradients in a real-imaginary representation can be algebraically cumbersome. Fortunately, we can overcome these issues by using a generalization of the complex gradient defined using Wirtinger calculus [23].

3.1. Unfolding the multichannel GMM

In this section, we formulate a complex-valued computational network inspired by the unfolded MCGMM variational algorithm. Algorithm 1 describes the sequence of updates performed in each layer of the network. A computational graph of the last two layers of the network is shown in figure 1.

We make two simplifications to the variational updates of the MCGMM in order to make them easier to take gradients through. We desire to avoid matrix inversions and to make the updates synchronous across all sources in each layer.

First, instead of solving a $J \times J$ linear system of equations to estimate the variational source estimates, as in Attias [21, eq. (12)], we elect to perform "synchronous" updates of the state-dependent source means given by (8), which are an alternative variational update. If these updates are performed on one source at a time, the variational bound is maintained. But performing these updates synchronously breaks the variational lower bound on the log-likelihood. In practice, we have not observed degradation of the separation performance, as long as the synchronous updates are preceded by at least a few iterations of the original variational updates. Using the synchronous updates, the output estimate of source j in layer k is the variational posterior mean $\hat{X}^{j,(k)}_{f,t}$, given by (11).

Our second simplification concerns the cross-source covariance matrix $\hat{\Sigma}^{\hat{X}\hat{X}}_{f,t}$. Attias [21] assumed a full covariance matrix, which necessitates a matrix inversion in the M-step for the update of the channel B (14). To avoid this matrix inversion, we make the reasonable assumption that sources are uncorrelated, and constrain the cross-source covariance matrix $\hat{\Sigma}^{\hat{X}\hat{X}}_{f,t}$ in (13) to be diagonal.

In preliminary experiments, discriminatively training the state priors $\pi^{(k)}$ and GMM variances $\gamma^{(k)}$ in each layer through these updates using the cost function in (19) did not yield substantial improvements. As such, in the next section we consider extending the generative model such that it has greater representational power. In particular, we will focus on improving estimation of source states, since these are essential for effective source separation.

3.2. MRF extension of the MCGMM

We would like to improve the unfolded MCGMM network's ability to estimate the correct state for each source at the output. One way to accomplish this is to add feedback to the network such that the estimated posterior log-likelihoods $L^{j,z,(k)}_t$ of the states in layer k (9) use information about the estimated posterior state likelihoods $\pi^{j,z,(k-1)}_t$ (10) in the previous layer, $k-1$.

Such a mechanism exists in a deep unfolded pairwise binary Markov random field (MRF): unfolding mean-field inference in a binary MRF leads to a deep feed-forward sigmoid network [18]. Given a MRF with M hidden binary random variables s_m, log potentials Ψ_{ss}, and the log-likelihood of the observed data L_{obs}, the posterior distribution can be written as

$$p(s|v) \propto \exp\left(\frac{1}{2}s^T A s + s^T c + s^T L_{\text{obs}}\right) \tag{5}$$

where $s := s_{1:M}$, $A \in \mathbb{R}^{M\times M}$, $A_{m,m} = 0$ for all m, $A_{m_1,m_2} = A_{m_2,m_1}$ for $m_1 \neq m_2$, $c \in \mathbb{R}^M$ are derived from the log potentials Ψ_{ss}, and $L_{\text{obs}} \in \mathbb{R}^M$ [18, Appendix A].

The variational posterior probability $\bar{\pi}^{(k)} := \{q^{(k)}(s_m)\}_{m=1:M}$ in iteration k of the mean-field inference algorithm is then

$$\bar{\pi}^{(k)} = \sigma\left(A\bar{\pi}^{(k-1)} + c + L_{\text{obs}}\right), \tag{6}$$

where σ is the element-wise sigmoid function. Notice that A and $c + L_{\text{obs}}$ define an affine transformation, and if these parameters are untied across layers, $A^{(k)}$ and $c^{(k)}$, then equation (6) is equivalent to one layer of a deep feed-forward sigmoid network. Discriminatively training the $A^{(k)}$ and $c^{(k)}$ in each layer is equivalent to finding a different set of log potential functions for the MRF for each iteration, such that the result of K iterations of inference minimizes the discriminative cost function. The expression $A^{(k)}\bar{\pi}^{(k-1)} + c^{(k)}$ is essentially a prior on the state log-likelihoods that varies from iteration to iteration, with feedback from the previously estimated state likelihoods $\bar{\pi}^{(k-1)}$.

To apply this in our model we can replace the multinomial state $z^j_t \in \{1..Z\}$ of a source with a MRF as in the above. To do this, let

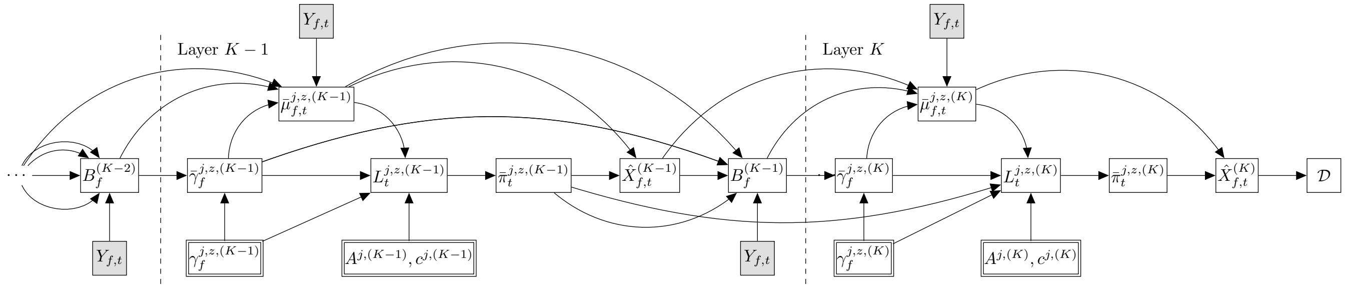

Fig. 1. Last two layers of the unfolded deep MCGMM. Boxes with double lines are the discriminatively-trained source parameters, and shaded boxes represent the observed data.

```
Algorithm 1: Simplified variational EM algorithm for the MCGMM, where $\langle(\cdot)_t\rangle_t := \frac{1}{T}\sum_{t=1}^{T}(\cdot)_t$.
Data: Multichannel mixture STFT $Y_{1:F,1:T}$, sensor precision $\psi_f$, source parameters $\gamma_{1:F}^{1:J,1:Z}$, $\pi^{1:J,1:Z}$, initial channel estimate $B_{1:F}^{(0)}$
Result: Estimated source STFTs $\hat{X}_{1:F,1:T}^{1:J,(K)}$ and layer-wise
        intermediate variables
for $k = 1 : K$ do
    Run E-step:
    $\bar{\gamma}_f^{j,z,(k)} = \big[B_f^{(k-1)}\big]_{:,j}^H \psi_f \big[B_f^{(k-1)}\big]_{:,j} + \gamma_f^{j,z,(k)}$    (7)
    $\bar{\mu}_{f,t}^{j,z,(k)} = \frac{\big[B_f^{(k-1)}\big]_{:,j}^H \psi_f}{\bar{\gamma}_f^{j,z,(k)}} \Big(Y_{f,t} - \big[B_f^{(k-1)}\big]_{:,\backslash j} \hat{X}_{f,t}^{\backslash j,(k-1)}\Big)$    (8)
    $L_t^{j,z,(k)} = \log \pi^{j,z} + \sum_f \log \frac{\gamma_f^{j,z,(k)}}{\bar{\gamma}_f^{j,z,(k)}} \dots$
        $\dots + \sum_f \bar{\gamma}_f^{j,z,(k)} \left|\bar{\mu}_f^{j,z,(k)}\right|^2$    (9)
    $\bar{\pi}_t^{j,z,(k)} = \mathrm{softmax}\Big(L_t^{j,1:Z,(k)}\Big)$    (10)
    $\hat{X}_{f,t}^{j,(k)} = \sum_z \bar{\pi}_t^{j,z,(k)} \bar{\mu}_{f,t}^{j,z,(k)}$    (11)
    Run M-step:
    $\hat{\Sigma}_f^{YX} = \Big\langle Y_{f,t} \big(\hat{X}_{f,t}^{(k)}\big)^H \Big\rangle_t$    (12)
    $\big[\hat{\Sigma}_f^{\hat{X}\hat{X}}\big]_{j,j} = \Big\langle \sum_z \bar{\pi}_t^{j,z,(k)} \Big(\frac{1}{\bar{\gamma}_f^{j,z,(k)}} + \left|\bar{\mu}_{f,t}^{j,z,(k)}\right|^2\Big)\Big\rangle_t$    (13)
    $B_f^{(k)} = \hat{\Sigma}_f^{Y\hat{X}} \left(\hat{\Sigma}_f^{\hat{X}\hat{X}}\right)^{-1}$    (14)
end
```

each multinomial state z_t^j be mapped to Z binary random variables $s_t^{j,z}$ in a fully-connected MRF, where $s_t^{j,1:Z}$ is constrained to be one-hot. We use the variational approximation $q(s_t^{j,1:Z}) = \prod_z \bar{\pi}_t^{j,z}$ for the binary random variables $s_t^{j,z}$, with variational probabilities $\bar{\pi}_t^{j,z} := q(s_t^{j,z} = 1, s_t^{j,z'} = 0, \ \forall z' \neq z)$. Rather than performing unconstrained mean-field updates, here for continuity with our GMM model, we constrain the variational posterior to behave like multinomial mixture states. As such $\bar{\pi}_f^{j,z}$ is the variational probability that the zth element of $s_t^{j,1:Z}$ is set to 1, and the other elements are set to 0. Then, if we unfold mean field inference for the hidden binary states $s_t^{j,z}$, we replace the multinomial prior $\log \pi^{j,z}$ in the update (9) with

$$L_{\text{prior},t}^{j,z,(k)} = A^{(k)} \bar{\pi}_t^{j,z,(k-1)} + c^{(k)}, \tag{15}$$

where the parameters $A^{(k)} \in \mathbb{R}^{Z \times Z}$ and $c^{(k)} \in \mathbb{R}^Z$ can be layer-dependent. When $A^{(k)} = 0$ and $c^{(k)} = \log \pi^{j,z}$ for all k, the new update (16) simplifies to the original variational update (9). Although the synchronous mean-field updates break the variational bound, we expect discriminative training to compensate such approximations.

The new update for $L_t^{j,z,(k)}$ that replaces (9) is thus

$$L_t^{j,z,(k)} = L_{\text{prior},t}^{j,z,(k)} + \alpha L_{\text{acoustic},t}^{j,z,(k)}, \tag{16}$$

with

$$L_{\text{acoustic},t}^{j,z,(k)} = \sum_f \log \frac{\gamma_f^{j,z,(k)}}{\bar{\gamma}_f^{j,z,(k)}} + \sum_f \bar{\gamma}_f^{j,z,(k)} \left|\bar{\mu}_f^{j,z,(k)}\right|^2. \tag{17}$$

Equation (17) is the part of the log-likelihood corresponding to acoustic information and α is an "acoustic weight" that expresses the importance of the acoustic evidence over the prior. We refer to the resulting network as a deep MCGMM (DMCGMM).

4. EXPERIMENTS AND DISCUSSION

We use a modified version[1] of the SimData and multicondition training (mcTrain) data components of the REVERB challenge dataset [24]. Each file consists of a single-channel speech utterance from the WSJCAM0 dataset [25] reverberated using measured 8-channel reverberation impulse responses (RIRs) in different rooms. SimData uses RIRs from three different rooms, and mcTrain uses RIRs from 6 different rooms. Stationary noise that was recorded in each particular room is added at 20 dB SNR. To create a dataset of overlapping speech, a second speech signal is reverberated using a RIR from a different position in the same room and added to the original file. No normalization of the power of the reverberated speech sources is performed, in order to simulate realistic conditions. The power ratio between the spatial images of speaker 1 and speaker 2 ranges from about -15 dB to $+15$ dB. All mixes are between 6 and 10 seconds long. The training set contains 15763 mixes, the development set contains 965 mixes, and the evaluation set contains 1435 mixes.

The initial source precisions $\gamma_f^{j,z,(0)}$ were trained on a gender-specific split of the WSJCAM0 training set. That is, two separate 256-component GMMs were trained for male and female speakers. Each GMM was first trained on the log-magnitude STFTs. Then, using the frame labels ℓ from the result, the GMM precisions γ_f^z were set to be $1/\sum_{t:\ell(t)=z} |X_{f,t}|^2$. Then these gender-specific GMMs

[1] Thanks to Michael Mandel for building this dataset during JSALT 2015.

Table 1. Source separation results on the evaluation set for the MCGMM and the deep MCGMM (DMCGMM). Units are in dB, given as SDRs of the source image (SDRim) and of the source (SDR). Results are given for various desired input SDRim.

MCGMM var.	DMCGMM	Desired input SDRim in dB							
EM layers	layers	-9	-6	-3	0	3	6	9	All
		Mean output SDRim in dB							
No proc.	—	-9.61	-5.87	-3.00	-0.01	2.97	5.84	9.51	-0.06
10	—	-0.55	3.17	5.42	6.75	7.51	7.68	7.12	5.88
15	—	-0.60	3.18	5.48	6.83	7.60	7.75	7.14	5.94
10	1	-0.70	3.16	5.59	6.97	7.70	7.86	7.21	6.02
10	2	-0.20	3.54	5.86	7.14	7.81	7.94	7.17	6.23
10	3	**0.78**	**4.28**	**6.17**	7.18	7.61	7.53	6.57	6.32
10	4	-0.17	3.73	5.96	**7.29**	**7.96**	**8.16**	**7.40**	**6.37**
		Mean output SDR in dB							
No proc.[2]	—	-27.32	-23.07	-22.03	-19.74	-18.43	-17.80	-18.39	-20.50
10	—	-0.43	1.77	3.52	4.48	5.44	6.17	6.86	4.19
15	—	-0.40	1.80	3.48	4.36	5.25	5.92	6.51	4.07
10	1	0.34	2.46	4.10	4.93	5.76	6.34	6.88	4.63
10	2	0.82	2.89	**4.49**	**5.27**	**6.07**	**6.59**	**7.07**	**4.97**
10	3	**1.11**	**3.01**	4.43	5.08	5.74	6.18	6.47	4.80
10	4	0.88	2.88	4.36	5.13	5.89	6.45	6.96	4.85

were concatenated into a 512-component GMM. The MRF parameters are initialized as $A^{(0)} = 0$ and $c^{(0)} = \log \pi^z$. Both sources use the same source model.

Since our main interest here is to observe the performance improvement of the DMCGMM over the conventional MCGMM, we used an oracle least-squares initialization for the channel model for each file:

$$B_f^{(0)} = \hat{\Sigma}_f^{YX} \left(\hat{\Sigma}_f^{XX} \right)^{-1}, \qquad (18)$$

where $\hat{\Sigma}_f^{YX}$ is the frequency-domain cross-covariance between the microphone observations $Y_{f,t}$ and reference sources $X_{f,t}$, and $\hat{\Sigma}_f^{XX}$ is the covariance between the reference sources $X_{f,t}$. The sensor precision ψ_f is set to 5 times the sample precision of the data $Y_{f,1:T}$.

For each file, 10 iterations of variational EM updates, as described in Section 2, are run. The output of these iterations is fed to a network of K DMCGMM layers, as described in Section 3.1. The parameters $\Theta^{(k)} = \{A^{(k)}, c^{(k)}, \gamma_f^{j,z,(k)}\}$ are untied between layers and discriminatively trained. We use an "error-to-source" (ESR) cost function given by

$$\mathcal{D}_{ESR}(\hat{X}_{f,t}^{(K)}, X_{f,t}) = \sum_j \frac{\sum_{f,t} \left| \hat{X}_{f,t}^{j,(K)} - X_{f,t}^j \right|^2}{\sum_{f,t} \left| X_{f,t}^j \right|^2}, \qquad (19)$$

where $\hat{X}_{f,t}^{(K)}$ are the estimated source STFT coefficients from the last (Kth) layer and $X_{f,t}$ are the clean single-channel references. By minimizing (19), the signal-to-noise ratio of both sources is maximized. Refer to the supplementary materials [26] for a detailed description and derivation of the gradients.

Performance is measured using signal-to-distortion ratios of the first channel of the source image estimates (SDRim) and the source estimates (SDR). SDRim is computed using `bss_eval_images` from the BSS Eval toolbox [27], and SDR is computed as the signal-to-noise ratio when the reference signal is allowed an arbitrary gain estimated using least-squares. The SDRs for "no processing" in table 1 are very low because the noisy mixtures contain a large amount of reverberation[2]. We incrementally train DMCGMMs layerwise, using the parameters with the best validation mean SDRim for each successive layer. To ensure the GMM source precisions $\gamma_f^{j,z,(k)}$ remain nonnegative, we optimize $\lambda_f^{j,z,(k)} := \log \gamma_f^{j,z,(k)}$, and replace all instances of $\gamma_f^{j,z,(k)}$ in the updates with $\exp \lambda_f^{j,z,(k)}$. Stochastic gradient descent is used for backpropagation with a batch size of 2 files (about 500 STFT frames on average). An initial learning rate of 20 gave the best results. Momentum of 0.9 is used. A validation set, with 65 randomly selected files from the development set, is scored every 300 gradient steps and used for early stopping. Stochastic gradient descent is performed for 30 epochs, with files randomly shuffled in each epoch.

The Bespoke Network Toolbox (BeNToBox)[3] was used for implementation in Matlab.. All computations are performed on a Nvidia GPU using the Matlab Parallel Processing Toolbox. For a 10 second mixture, this implementation takes about 2 seconds to perform the MCGMM variational algorithm and about 500 milliseconds to perform a DMCGMM forward pass and gradient computation for backpropagation. Code to replicate our experiments is available in the supplementary materials [26].

Table 1 shows the results on the evaluation set, which are given in SDRim and SDR. Scores are averaged over in categories based on input SDR. Notice that both SDRim and SDR generally increase as the number of discriminatively-trained DMCGMM layers increases.

5. CONCLUSION AND FUTURE WORK

In this paper, we have unfolded an existing model-based variational inference algorithm for separating multimicrophone mixtures into a novel computational network. The resulting network was augmented with deep sigmoid network-like components that estimate source states to create the DMCGMM. The DMCGMM directly processes complex-valued frequency-domain inputs, and by discriminatively training DMCGMM source model parameters achieves superior performance over the model-based variational inference algorithm.

In the future, we will explore other enhancements and generalizations of this network, including incorporation of recurrence, more sophisticated extensions of the model, other types of cost functions such as cross-entropy on the source states, and combination with automatic speech recognition systems.

[2] Note that only the gain is adapted because a more flexible adaptation between the reference and signal would constitute an oracle microphone array method, and our aim here is to show the SNR without processing.

[3] Available from `github.com/stwisdom/bentobox`.

6. REFERENCES

[1] E. Habets, J. Benesty, I. Cohen, S. Gannot, and J. Dmochowski, "New Insights Into the MVDR Beamformer in Room Acoustics," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 1, pp. 158–170, 2010.

[2] M. Souden, S. Araki, K. Kinoshita, T. Nakatani, and H. Sawada, "A Multichannel MMSE-Based Framework for Speech Source Separation and Noise Reduction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 9, pp. 1913–1928, Sept. 2013.

[3] N. Duong, E. Vincent, and R. Gribonval, "Under-Determined Reverberant Audio Source Separation Using a Full-Rank Spatial Covariance Model," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1830–1840, Sept. 2010.

[4] M. I. Mandel, R. J. Weiss, and D. P. Ellis, "Model-based expectation-maximization source separation and localization," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 18, no. 2, pp. 382–394, 2010.

[5] A. Ozerov and C. Févotte, "Multichannel Nonnegative Matrix Factorization in Convolutive Mixtures for Audio Source Separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 550–563, Mar. 2010.

[6] H. Sawada, H. Kameoka, S. Araki, and N. Ueda, "Multichannel Extensions of Non-Negative Matrix Factorization With Complex-Valued Data," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 971–982, May 2013.

[7] T. Higuchi and H. Kameoka, "Unified approach for underdetermined BSS, VAD, dereverberation and DOA estimation with multichannel factorial HMM," in *Proc. GlobalSIP*, Dec. 2014, pp. 562–566.

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

[9] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *Proc. ICASSP*, Vancouver, Canada, May 2013, pp. 7092–7096.

[10] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Deep learning for monaural speech separation," in *Proc. ICASSP*, Florence, Italy, 2014, pp. 1562–1566, IEEE.

[11] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Phase-sensitive and Recognition-boosted Speech Separation Using Deep Recurrent Neural Networks," in *Proc. ICASSP*, Brisbane, Australia, 2015.

[12] F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. Le Roux, J. R. Hershey, and B. Schuller, "Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR," in *Latent Variable Analysis and Signal Separation*, pp. 91–99. Springer, 2015.

[13] M. L. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Proc. ICASSP*. 2013, pp. 7398–7402, IEEE.

[14] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, vol. 22, no. 4, pp. 745–777, 2014.

[15] P. Swietojanski, A. Ghoshal, and S. Renals, "Convolutional Neural Networks for Distant Speech Recognition," *IEEE Signal Processing Letters*, vol. 21, no. 9, pp. 1120–1124, Sept. 2014.

[16] Y. Hoshen, R. J. Weiss, and K. W. Wilson, "Speech acoustic modeling from raw multichannel waveforms," in *Proc. ICASSP*, Brisbane, Australia, Apr. 2015.

[17] A. A. Nugraha, A. Liutkus, and E. Vincent, "Multichannel audio source separation with deep neural networks," report, INRIA, June 2015.

[18] J. R. Hershey, J. Le Roux, and F. Weninger, "Deep Unfolding: Model-Based Inspiration of Novel Deep Architectures," *arXiv:1409.2574 [cs, stat]*, Sept. 2014, arXiv: 1409.2574.

[19] J. Le Roux, J. R. Hershey, and F. J. Weninger, "Deep NMF for Speech Enhancement," in *Proc. ICASSP*, Brisbane, Australia, 2015.

[20] E. Variani, E. McDermott, and G. Heigold, "A Gaussian Mixture Model Layer Jointly Optimized with Discriminative Features within a Deep Neural Network Archtecture," in *Proc. ICASSP*, Brisbane, Australia, 2015.

[21] H. Attias, "New EM algorithms for source separation and deconvolution with a microphone array," in *Proc. ICASSP*, Apr. 2003, vol. 5, pp. V–297–300 vol.5.

[22] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, "An Introduction to Variational Methods for Graphical Models," *Machine Learning*, vol. 37, no. 2, pp. 183–233, Nov. 1999.

[23] K. Kreutz-Delgado, "The Complex Gradient Operator and the CR-Calculus," *arXiv:0906.4835 [math]*, June 2009, arXiv: 0906.4835.

[24] K. Kinoshita, M. Delcroix, T. Yoshioka, T. Nakatani, E. Habets, R. Haeb-Umbach, V. Leutnant, A. Sehr, W. Kellermann, and R. Maas, "The REVERB challenge: A common evaluation framework for dereverberation and recognition of reverberant speech," in *Proc. WASPAA*, New Paltz, NY, 2013.

[25] T. Robinson, J. Fransen, D. Pye, J. Foote, and S. Renals, "Wsjcam0: A British English Speech Corpus For Large Vocabulary Continuous Speech Recognition," in *Proc. ICASSP*, Detroit, MI, 1995, pp. 81–84.

[26] S. Wisdom, J. R. Hershey, J. Le Roux, and S. Watanabe, "Deep MCGMM project webpage: Supplementary materials," `http://www.merl.com/demos/deep-MCGMM`, 2015, [Online].

[27] E. Vincent, S. Araki, F. Theis, G. Nolte, P. Bofill, H. Sawada, A. Ozerov, V. Gowreesunker, D. Lutter, and N. Q. Duong, "The signal separation evaluation campaign (2007–2010): Achievements and remaining challenges," *Signal Processing*, vol. 92, no. 8, pp. 1928–1936, 2012.