HAL open science

Semi-supervised multichannel speech enhancement with variational autoencoders and non-negative matrix factorization

Simon Leglaive, Laurent Girin, Radu Horaud

To cite this version:

Simon Leglaive, Laurent Girin, Radu Horaud. Semi-supervised multichannel speech enhancement with variational autoencoders and non-negative matrix factorization. ICASSP 2019 - IEEE International Conference on Acoustics, Speech and Signal Processing, May 2019, Brighton, United Kingdom. pp.101-105, 10.1109/ICASSP.2019.8683704 . hal-02005102v2

HAL Id: hal-02005102

https://inria.hal.science/hal-02005102v2

Submitted on 30 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

SEMI-SUPERVISED MULTICHANNEL SPEECH ENHANCEMENT WITH VARIATIONAL AUTOENCODERS AND NON-NEGATIVE MATRIX FACTORIZATION

Simon Leglaive[1] *Laurent Girin*[1,2] *Radu Horaud*[1]

[1]Inria Grenoble Rhône-Alpes, France [2]Univ. Grenoble Alpes, Grenoble INP, GIPSA-lab, France

ABSTRACT

In this paper we address speaker-independent multichannel speech enhancement in unknown noisy environments. Our work is based on a well-established multichannel local Gaussian modeling framework. We propose to use a neural network for modeling the speech spectro-temporal content. The parameters of this supervised model are learned using the framework of variational autoencoders. The noisy recording environment is supposed to be unknown, so the noise spectro-temporal modeling remains unsupervised and is based on non-negative matrix factorization (NMF). We develop a Monte Carlo expectation-maximization algorithm and we experimentally show that the proposed approach outperforms its NMF-based counterpart, where speech is modeled using supervised NMF.

***Index Terms*—** Multichannel speech enhancement, local Gaussian modeling, variational autoencoders, non-negative matrix factorization, Monte Carlo expectation-maximization.

1. INTRODUCTION

Speech enhancement is a classical problem of speech processing which aims at recovering a clean speech signal from a noisy recording [1]. In this work we focus on multichannel speech enhancement in additive noise, with static (non moving) speech and noise sources. We follow a statistical approach based on a local Gaussian model of the time-frequency (complex-valued) signal coefficients. This approach has been very popular for addressing the audio source separation problem [2, 3, 4, 5, 6, 7, 8, 9]. In the multichannel case, the covariance matrix of this model can be structured as the product of a time-frequency-dependent *variance*, accounting for the spectro-temporal content of the source signal, and a frequency-dependent *spatial covariance matrix* (SCM), accounting for the spatial properties of the image signal [3]. The variance term was further structured by means of a non-negative matrix factorization (NMF) model in [2, 4, 5, 6]. More recently, within this local Gaussian modeling framework based on SCMs, deep neural networks (DNNs) were investigated as a variance model [8]. However, in [8] the DNN is not really integrated in the generative source modeling, it is rather used as an ad-hoc "deterministic" variance estimator.

For single-channel speech enhancement, DNN-based discriminative approaches have been successfully investigated for estimating either time-frequency masks or clean spectrograms from noisy spectrograms [10]. Very recently, DNN-based generative models, and in particular variational autoencoders (VAEs) [11], have been employed for single-channel speech enhancement [12, 13, 14]. In [13] we highlighted conceptual similarities between this approach and a more conventional generative model based on Itakura-Saito NMF [15]. We also showed that the VAE-based speech model outperformed its NMF-based counterpart.

This work is supported by the ERC Advanced Grant VHIA #340113.

In this paper, we present an extension of [13] to the multichannel case. The speech and noise models are based on the above-mentioned multichannel local Gaussian modeling approach. The speech source spectro-temporal content is further modeled in a supervised way by means of a DNN, whose parameters are learned using the VAE framework. We do not assume any prior knowledge on the recording environment, so the multichannel images of both the speech and the noise signals include a free SCM model. Moreover the noise source spectro-temporal model remains unsupervised and is based on NMF. We propose a Monte Carlo expectation-maximization (MCEM) algorithm [16] for estimating the model parameters. Experiments show that our approach outperforms its NMF counterpart [6], where the speech signal is modeled using supervised NMF. Note that a similar model for multichannel speech enhancement was published independently in [17], after we submitted this paper. It is also combines VAE and NMF within a multichannel local Gaussian modeling framework, but it relies on a fully Bayesian approach.

2. VAE-BASED SPEECH SOURCE MODEL

We work in the short-term Fourier transform (STFT) domain where $\mathbb{B} = \{0, ..., F-1\} \times \{0, ..., N-1\}$ denotes the set of time-frequency bins. For $(f, n) \in \mathbb{B}$, f denotes the frequency index and n the time-frame index. As in [12, 13], independently for every $(f, n) \in \mathbb{B}$, we consider the following generative speech source model involving a latent random vector $\mathbf{z}_n \in \mathbb{R}^L$:

$$\mathbf{z}_n \sim \mathcal{N}(\mathbf{0}, \mathbf{I}); \tag{1}$$

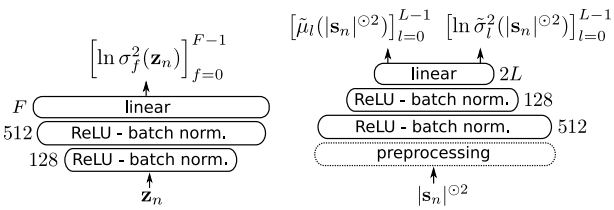
$$s_{fn} \mid \mathbf{z}_n \sim \mathcal{N}_c(0, \sigma_f^2(\mathbf{z}_n)), \tag{2}$$

where $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the multivariate Gaussian distribution for a real-valued random vector, $\mathcal{N}_c(\mu, \sigma^2)$ denotes the univariate complex proper [18] Gaussian distribution and $\mathbf{I}$ is the identity matrix of appropriate size. As represented in Fig. 1a, $\{\sigma_f^2 : \mathbb{R}^L \mapsto \mathbb{R}_+\}_{f=0}^{F-1}$ is a set of non-linear functions provided by a DNN which takes as input $\mathbf{z}_n \in \mathbb{R}^L$. In the following, we will denote by $\boldsymbol{\theta}_s$ the parameters of this *generative network*, which can be seen as a model of the speech short-term power spectral density [19].

A key contribution of VAEs is to provide an efficient way of learning the parameters $\boldsymbol{\theta}_s$ of such a generative model [11]. Let $\mathbf{s} = \{\mathbf{s}_n \in \mathbb{C}^F\}_{n=0}^{N-1}$ be a training dataset of clean-speech STFT spectra and $\mathbf{z} = \{\mathbf{z}_n \in \mathbb{R}^L\}_{n=0}^{N-1}$ the set of associated latent random vectors. Taking ideas from variational inference, the parameters $\boldsymbol{\theta}_s$ are estimated by maximizing a lower bound $\mathcal{L}(\boldsymbol{\theta}_s, \boldsymbol{\phi})$ of the log-likelihood $\ln p(\mathbf{s}; \boldsymbol{\theta}_s)$ defined by:

$$\mathcal{L}(\boldsymbol{\theta}_s, \boldsymbol{\phi}) = \mathbb{E}_{q(\mathbf{z}|\mathbf{s};\boldsymbol{\phi})}\left[\ln p(\mathbf{s} \mid \mathbf{z}; \boldsymbol{\theta}_s)\right] - D_{\mathrm{KL}}\left(q(\mathbf{z} \mid \mathbf{s}; \boldsymbol{\phi}) \parallel p(\mathbf{z})\right), \tag{3}$$

where $q(\mathbf{z} \mid \mathbf{s}; \boldsymbol{\phi})$ approximates the intractable true posterior distribution $p(\mathbf{z} \mid \mathbf{s}; \boldsymbol{\theta}_s)$, and $D_{\mathrm{KL}}(q \parallel p) = \mathbb{E}_q[\ln(q/p)]$ is the



(a) Generative network. (b) Recognition network.

Fig. 1: Neural network architectures. The number next to each layer indicates its size. Further architecture details are given in Section 5.

Kullback-Leibler divergence. Independently for all the dimensions $l \in \{0, ..., L-1\}$ and all the time frames $n \in \{0, ..., N-1\}$, $q(\mathbf{z} \mid \mathbf{s}; \boldsymbol{\phi})$ is defined by:

$$z_{l,n} \mid \mathbf{s}_n \sim \mathcal{N}\left(\tilde{\mu}_l\left(|\mathbf{s}_n|^{\odot 2}\right), \tilde{\sigma}_l^2\left(|\mathbf{s}_n|^{\odot 2}\right)\right), \tag{4}$$

where $z_{l,n} = (\mathbf{z}_n)_l$ and $(\cdot)^{\odot \cdot}$ denotes element-wise exponentiation. As represented in Fig. 1b, $\{\tilde{\mu}_l : \mathbb{R}_+^F \mapsto \mathbb{R}\}_{l=0}^{L-1}$ and $\{\tilde{\sigma}_l^2 : \mathbb{R}_+^F \mapsto \mathbb{R}_+\}_{l=0}^{L-1}$ are non-linear functions provided by a DNN which takes as input the speech power spectrum at a given time frame. $\boldsymbol{\phi}$ denotes the parameters of this *recognition network*, which also have to be estimated by maximizing the *variational lower bound* defined in (3). Using (1), (2) and (4) we can develop (3) as follows:

$$\mathcal{L}(\boldsymbol{\theta}_s, \boldsymbol{\phi}) \overset{c}{=} -\sum_{f=0}^{F-1}\sum_{n=0}^{N-1} \mathbb{E}_{q(\mathbf{z}_n|\mathbf{s}_n;\boldsymbol{\phi})}\left[d_{\text{IS}}\left(|s_{fn}|^2; \sigma_f^2(\mathbf{z}_n)\right)\right]$$
$$+ \frac{1}{2}\sum_{l=0}^{L-1}\sum_{n=0}^{N-1}\left[\ln \tilde{\sigma}_l^2\left(|\mathbf{s}_n|^{\odot 2}\right) - \tilde{\mu}_l\left(|\mathbf{s}_n|^{\odot 2}\right)^2 - \tilde{\sigma}_l^2\left(|\mathbf{s}_n|^{\odot 2}\right)\right], \tag{5}$$

where $\overset{c}{=}$ denotes equality up to an additive constant and $d_{\text{IS}}(x; y) = x/y - \ln(x/y) - 1$ is the Itakura-Saito divergence. Finally, using the so-called "reparametrization trick" [11] to approximate the intractable expectation in (5), we obtain an objective function which is differentiable with respect to both $\boldsymbol{\theta}_s$ and $\boldsymbol{\phi}$ and can be optimized using gradient-ascent-based algorithms. It is important to note that the only reason why the recognition network is introduced is to learn the parameters of the generative network.

3. COMPLETE MULTICHANNEL MODEL

In the previous section we have seen how to learn the parameters of the generative model (1)-(2). This model can then be used as a speech source signal probabilistic prior for a variety of applications. In this paper we are interested in multichannel speech enhancement. We now present the model developed for this task. In the following, I denotes the number of microphones, and probabilistic models are defined independently for all time-frequency points $(f, n) \in \mathbb{B}$.

The multichannel speech signal $\mathbf{s}_{fn} \in \mathbb{C}^I$ is modeled as follows:

$$\mathbf{s}_{fn} \mid \mathbf{z}_n \sim \mathcal{N}_c\big(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{s},f}(\mathbf{z}_n)\big), \tag{6}$$

where $\boldsymbol{\Sigma}_{\mathbf{s},f}(\mathbf{z}_n) := \sigma_f^2(\mathbf{z}_n)\mathbf{R}_{\mathbf{s},f}$. $\sigma_f^2(\cdot)$ was introduced in (2) and models the speech signal power spectral density, $\mathbf{z}_n$ follows the prior in (1), $\mathbf{R}_{\mathbf{s},f} \in \mathbb{C}^{I\times I}$ is the speech signal spatial covariance matrix (which is Hermitian positive definite) [3] and $\mathcal{N}_c(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the multivariate complex proper Gaussian distribution.

The multichannel noise signal $\mathbf{b}_{fn} \in \mathbb{C}^I$ is modeled as follows:

$$\mathbf{b}_{fn} \sim \mathcal{N}_c\big(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{b},fn}\big), \tag{7}$$

where $\boldsymbol{\Sigma}_{\mathbf{b},fn} := (\mathbf{W}_b\mathbf{H}_b)_{f,n}\mathbf{R}_{\mathbf{b},f}$, with $\mathbf{W}_b = [w_{b,fk}]_{f,k} \in \mathbb{R}_+^{F\times K_b}$, $\mathbf{H}_b = [h_{b,kn}]_{k,n} \in \mathbb{R}_+^{K_b\times N}$ and $\mathbf{R}_{\mathbf{b},f} \in \mathbb{C}^{I\times I}$ is the noise SCM (which is also Hermitian positive definite). The noise spectro-temporal content is thus modeled by NMF [15], and it remains unsupervised, i.e. $\mathbf{W}_b$ and $\mathbf{H}_b$ will be estimated from test data (and so will be the speech and noise SCMs).

The noisy mixture signal $\mathbf{x}_{fn} \in \mathbb{C}^I$ is finally modeled by:

$$\mathbf{x}_{fn} = \sqrt{g_n}\mathbf{s}_{fn} + \mathbf{b}_{fn}, \tag{8}$$

where $g_n \in \mathbb{R}_+$ is a gain parameter whose importance has been experimentally shown in [13]. We further assume the independence of the speech and noise signals so that:

$$\mathbf{x}_{fn} \mid \mathbf{z}_n \sim \mathcal{N}_c(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{x},fn}(\mathbf{z}_n)), \tag{9}$$

where $\boldsymbol{\Sigma}_{\mathbf{x},fn}(\mathbf{z}_n) := g_n\boldsymbol{\Sigma}_{\mathbf{s},f}(\mathbf{z}_n) + \boldsymbol{\Sigma}_{\mathbf{b},fn}$.

4. MCEM ALGORITHM AND INFERENCE

Let $\boldsymbol{\theta}_u = \left\{\mathbf{W}_b, \mathbf{H}_b, \{\mathbf{R}_{\mathbf{s},f}, \mathbf{R}_{\mathbf{b},f}\}_{f=0}^{F-1}, \mathbf{g} = [g_0, ..., g_{N-1}]^\top\right\}$ be the set of model parameters to be estimated. In this section we develop an MCEM algorithm [16] for this aim. Remember that the parameters $\boldsymbol{\theta}_s$ of the speech source model have been learned during a training phase (see Section 2). The set of observed data is denoted by $\mathbf{x} = \{\mathbf{x}_{fn}\}_{(f,n)\in\mathbb{B}}$ while the set of latent variables is $\mathbf{z} = \{\mathbf{z}_n\}_{n=0}^{N-1}$. We will also use $\mathbf{x}_n = \{\mathbf{x}_{fn}\}_{f=0}^{F-1}$ to denote all the observations for a given time frame $n \in \{0, ..., N-1\}$.

Monte Carlo E-Step. Let $\boldsymbol{\theta}_u^\star$ denote the current model parameters. At the E-step of a standard EM algorithm, we would compute the following conditional expectation of the complete-data log-likelihood $Q(\boldsymbol{\theta}_u; \boldsymbol{\theta}_u^\star) = \mathbb{E}_{p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u^\star)}\left[\ln p(\mathbf{x}, \mathbf{z}; \boldsymbol{\theta}_s, \boldsymbol{\theta}_u)\right]$. However, due to the non-linear relationship between the latent variables and the observations, we cannot compute this expectation in an analytical form. We thus approximate $Q(\boldsymbol{\theta}_u; \boldsymbol{\theta}_u^\star)$ using an empirical average:

$$\tilde{Q}(\boldsymbol{\theta}_u; \boldsymbol{\theta}_u^\star) \overset{c}{=} -\frac{1}{R}\sum_{r=1}^{R}\sum_{(f,n)\in\mathbb{B}}\Big[\operatorname{tr}\Big(\mathbf{x}_{fn}\mathbf{x}_{fn}^H\Big[\boldsymbol{\Sigma}_{\mathbf{x},fn}\big(\mathbf{z}_n^{(r)}\big)\Big]^{-1}\Big)$$
$$+ \ln\det\Big(\boldsymbol{\Sigma}_{\mathbf{x},fn}\big(\mathbf{z}_n^{(r)}\big)\Big)\Big], \tag{10}$$

where $\cdot^H$ denotes the Hermitian transpose operator, $\operatorname{tr}(\cdot)$ and $\det(\cdot)$ denote respectively the trace and determinant of a matrix, and $\{\mathbf{z}_n^{(r)}\}_{r=1}^R$ is a sequence of samples asymptotically drawn from the posterior $p(\mathbf{z}_n \mid \mathbf{x}_n; \boldsymbol{\theta}_s, \boldsymbol{\theta}_u^\star)$ using a Markov chain Monte Carlo (MCMC) method. This approach forms the basis of the MCEM algorithm. Here we use the Metropolis-Hastings algorithm [20]. Note that unlike the standard EM algorithm, the likelihood is not guaranteed to increase at each iteration. Nevertheless, some convergence results in terms of stationary point of the likelihood can be obtained under suitable conditions [21].

At the m-th iteration of the Metropolis-Hastings algorithm and independently for all $n \in \{0, ..., N-1\}$, we first draw a sample $\tilde{\mathbf{z}}_n$ from a random walk proposal distribution: $\mathbf{z}_n \mid \mathbf{z}_n^{(m-1)} \sim \mathcal{N}(\mathbf{z}_n^{(m-1)}, \epsilon^2\mathbf{I})$. Using the symmetry of this proposal distribution, we then compute the following acceptance probability:

$$\alpha_n = \min\left(1, \frac{p(\tilde{\mathbf{z}}_n)\prod_{f=0}^{F-1} p(\mathbf{x}_{fn} \mid \tilde{\mathbf{z}}_n; \boldsymbol{\theta}_s, \boldsymbol{\theta}_u^\star)}{p\left(\mathbf{z}_n^{(m-1)}\right)\prod_{f=0}^{F-1} p\left(\mathbf{x}_{fn} \mid \mathbf{z}_n^{(m-1)}; \boldsymbol{\theta}_s, \boldsymbol{\theta}_u^\star\right)}\right). \tag{11}$$

Then we draw u_n from a uniform distribution $\mathcal{U}([0,1])$. If $u_n < \alpha_n$, we accept the sample and set $\mathbf{z}_n^{(m)} = \tilde{\mathbf{z}}_n$, otherwise we reject the sample and set $\mathbf{z}_n^{(m)} = \mathbf{z}_n^{(m-1)}$. We only keep the last R samples for computing $\tilde{Q}(\boldsymbol{\theta}_u;\boldsymbol{\theta}_u^\star)$, i.e. we discard the samples drawn during a so-called burn-in period.

M-Step. At the M-step, we want to minimize the cost function $\mathcal{C}(\boldsymbol{\theta}_u) := -RQ(\boldsymbol{\theta}_u;\boldsymbol{\theta}_u^\star)$ with respect to $\boldsymbol{\theta}_u$. Similarly to [6], we adopt a majorization-minimization approach for solving this optimization problem. From (10), (9), (7), (6) and using inequalities defined in Appendix A, we can show that $\mathcal{C}(\boldsymbol{\theta}_u) \leq \mathcal{G}(\boldsymbol{\theta}_u,\boldsymbol{\Phi},\boldsymbol{\Omega})$ where the majorizing function is defined by:

$$\begin{aligned}\mathcal{G}(\boldsymbol{\theta}_u,\boldsymbol{\Phi},\boldsymbol{\Omega}) &= -IFNR \\ &+ \sum_{r=1}^{R}\sum_{(f,n)\in\mathbb{B}}\Bigg[\frac{1}{g_n\sigma_f^2\left(\mathbf{z}_n^{(r)}\right)}\operatorname{tr}\left(\mathbf{x}_{fn}\mathbf{x}_{fn}^H\left(\boldsymbol{\Phi}_{0,fn}^{(r)}\right)^H\mathbf{R}_{\mathbf{s},f}^{-1}\boldsymbol{\Phi}_{0,fn}^{(r)}\right) \\ &\quad + \sum_{k=1}^{K_b}\Bigg(\frac{1}{w_{b,fk}h_{b,kn}}\operatorname{tr}\left(\mathbf{x}_{fn}\mathbf{x}_{fn}^H\left(\boldsymbol{\Phi}_{k,fn}^{(r)}\right)^H\mathbf{R}_{\mathbf{b},f}^{-1}\boldsymbol{\Phi}_{k,fn}^{(r)}\right) \\ &\qquad + w_{b,fk}h_{b,kn}\operatorname{tr}\left(\left(\boldsymbol{\Omega}_{fn}^{(r)}\right)^{-1}\mathbf{R}_{\mathbf{b},f}\right)\Bigg) \\ &\quad + g_n\sigma_f^2\left(\mathbf{z}_n^{(r)}\right)\operatorname{tr}\left(\left(\boldsymbol{\Omega}_{fn}^{(r)}\right)^{-1}\mathbf{R}_{\mathbf{s},f}\right) + \ln\det\left(\boldsymbol{\Omega}_{fn}^{(r)}\right)\Bigg], \quad (12)\end{aligned}$$

where $\boldsymbol{\Phi} = \left\{\boldsymbol{\Phi}_{k,fn}^{(r)} \in \mathbb{C}^{I\times I}\right\}_{r,k,f,n}$, $\boldsymbol{\Omega} = \left\{\boldsymbol{\Omega}_{fn}^{(r)} \in \mathbb{C}^{I\times I}\right\}_{r,f,n}$ are *auxiliary variables*. This upper bound is tight, i.e. $\mathcal{C}(\boldsymbol{\theta}_u) = \mathcal{G}(\boldsymbol{\theta}_u,\boldsymbol{\Phi},\boldsymbol{\Omega})$ if

$$\boldsymbol{\Omega}_{fn}^{(r)} = \boldsymbol{\Sigma}_{\mathbf{x},fn}\left(\mathbf{z}_n^{(r)}\right); \quad (13)$$

$$\boldsymbol{\Phi}_{0,fn}^{(r)} = g_n\sigma_f^2\left(\mathbf{z}_n^{(r)}\right)\mathbf{R}_{\mathbf{s},f}\left[\boldsymbol{\Sigma}_{\mathbf{x},fn}\left(\mathbf{z}_n^{(r)}\right)\right]^{-1}; \quad (14)$$

$$\boldsymbol{\Phi}_{k,fn}^{(r)} = w_{b,fk}h_{b,kn}\mathbf{R}_{\mathbf{b},f}\left[\boldsymbol{\Sigma}_{\mathbf{x},fn}\left(\mathbf{z}_n^{(r)}\right)\right]^{-1} \forall k \in \{1,...,K_b\}. \quad (15)$$

Although not jointly convex, $\mathcal{G}(\boldsymbol{\theta}_u,\boldsymbol{\Phi},\boldsymbol{\Omega})$ is separately convex in each of the individual parameters g_n, $w_{b,fk}$, $h_{b,kn}$, $\mathbf{R}_{\mathbf{s},f}$ and $\mathbf{R}_{\mathbf{b},f}$. We will therefore update individually and alternatively those parameters, which corresponds to a (block) coordinate approach. For one of those parameters, the general procedure consists in canceling the partial derivative of $\mathcal{G}(\boldsymbol{\theta}_u,\boldsymbol{\Phi},\boldsymbol{\Omega})$. In this way we obtain an update that depends on the auxiliary variables. We then inject in this update the expressions of the auxiliary variables that make the upper bound tight, which are given by (13)-(15). For notational convenience, let us first introduce:

$$\mathbf{M}_{fn}^{(r)} := \left(\boldsymbol{\Sigma}_{\mathbf{x},fn}\left(\mathbf{z}_n^{(r)}\right)\right)^{-1}\mathbf{x}_{fn}\mathbf{x}_{fn}^H\left(\boldsymbol{\Sigma}_{\mathbf{x},fn}\left(\mathbf{z}_n^{(r)}\right)\right)^{-1}. \quad (16)$$

The resulting updates are given as follows:

$$w_{fk} \leftarrow w_{fk}\left[\frac{\sum_{r=1}^{R}\sum_{n=0}^{N-1}h_{kn}\operatorname{tr}\left[\mathbf{M}_{fn}^{(r)}\mathbf{R}_{\mathbf{b},f}\right]}{\sum_{r=1}^{R}\sum_{n=0}^{N-1}h_{kn}\operatorname{tr}\left[\left(\boldsymbol{\Sigma}_{\mathbf{x},fn}\left(\mathbf{z}_n^{(r)}\right)\right)^{-1}\mathbf{R}_{\mathbf{b},f}\right]}\right]^{1/2}; \quad (17)$$

$$h_{kn} \leftarrow h_{kn}\left[\frac{\sum_{r=1}^{R}\sum_{f=0}^{F-1}w_{fk}\operatorname{tr}\left[\mathbf{M}_{fn}^{(r)}\mathbf{R}_{\mathbf{b},f}\right]}{\sum_{r=1}^{R}\sum_{f=0}^{F-1}w_{fk}\operatorname{tr}\left[\left(\boldsymbol{\Sigma}_{\mathbf{x},fn}\left(\mathbf{z}_n^{(r)}\right)\right)^{-1}\mathbf{R}_{\mathbf{b},f}\right]}\right]^{1/2}; \quad (18)$$

$$g_n \leftarrow g_n\left[\frac{\sum_{r=1}^{R}\sum_{f=0}^{F-1}\sigma_f^2\left(\mathbf{z}_n^{(r)}\right)\operatorname{tr}\left[\mathbf{M}_{fn}^{(r)}\mathbf{R}_{\mathbf{s},f}\right]}{\sum_{r=1}^{R}\sum_{f=0}^{F-1}\sigma_f^2\left(\mathbf{z}_n^{(r)}\right)\operatorname{tr}\left[\left(\boldsymbol{\Sigma}_{\mathbf{x},fn}\left(\mathbf{z}_n^{(r)}\right)\right)^{-1}\mathbf{R}_{\mathbf{s},f}\right]}\right]^{1/2}. \quad (19)$$

Updating the SCMs is done by solving the two following algebraic Riccati equations (see the procedure in [6, Appendix I]):

$$\begin{aligned}&\mathbf{R}_{\mathbf{s},f}\left[\sum_{r=1}^{R}\sum_{n=0}^{N-1}g_n\sigma_f^2\left(\mathbf{z}_n^{(r)}\right)\left(\boldsymbol{\Sigma}_{\mathbf{x},fn}\left(\mathbf{z}_n^{(r)}\right)\right)^{-1}\right]\mathbf{R}_{\mathbf{s},f} \\ &\quad = \tilde{\mathbf{R}}_{\mathbf{s},f}\left[\sum_{r=1}^{R}\sum_{n=0}^{N-1}g_n\sigma_f^2\left(\mathbf{z}_n^{(r)}\right)\mathbf{M}_{fn}^{(r)}\right]\tilde{\mathbf{R}}_{\mathbf{s},f}; \quad (20)\end{aligned}$$

$$\begin{aligned}&\mathbf{R}_{\mathbf{b},f}\left[\sum_{r=1}^{R}\sum_{n=0}^{N-1}\left(\mathbf{W}_b\mathbf{H}_b\right)_{f,n}\left(\boldsymbol{\Sigma}_{\mathbf{x},fn}\left(\mathbf{z}_n^{(r)}\right)\right)^{-1}\right]\mathbf{R}_{\mathbf{b},f} \\ &\quad = \tilde{\mathbf{R}}_{\mathbf{b},f}\left[\sum_{r=1}^{R}\sum_{n=0}^{N-1}\left(\mathbf{W}_b\mathbf{H}_b\right)_{f,n}\mathbf{M}_{fn}^{(r)}\right]\tilde{\mathbf{R}}_{\mathbf{b},f}, \quad (21)\end{aligned}$$

where $\tilde{\mathbf{R}}_{\mathbf{s},f}$ and $\tilde{\mathbf{R}}_{\mathbf{b},f}$ denote the SCMs before update. Note that $\boldsymbol{\Sigma}_{\mathbf{x},fn}\left(\mathbf{z}_n^{(r)}\right)$ and therefore $\mathbf{M}_{fn}^{(r)}$ should be updated after each one of the updates (17)-(21).

The criterion $\mathcal{C}(\boldsymbol{\theta}_u)$ suffers from scale indeterminacies between $\mathbf{R}_{\mathbf{b},f}$ and $w_{b,fk}$, and also between $w_{b,fk}$ and $h_{b,kn}$. At each iteration of the algorithm, we compensate these indeterminacies by normalizing the noise SCM so that $\operatorname{tr}(\mathbf{R}_{\mathbf{b},f}) = 1$ (and scaling the rows of $\mathbf{W}_b$ accordingly so that the criterion remains unchanged) and by normalizing the noise dictionary matrix so that $\sum_{f=0}^{F-1} w_{b,fk} = 1$ (and scaling the columns of $\mathbf{H}_b$ accordingly).

Speech reconstruction. Once the unsupervised model parameters are estimated with the MCEM algorithm, we need to estimate the multichannel speech signal. For all $(f,n) \in \mathbb{B}$, let $\tilde{\mathbf{s}}_{fn} = \sqrt{g_n}\mathbf{s}_{fn}$ be the scaled version of the multichannel speech signal. We estimate this vector according to its posterior mean:

$$\begin{aligned}\hat{\tilde{\mathbf{s}}}_{fn} &= \mathbb{E}_{p(\tilde{\mathbf{s}}_{fn}|\mathbf{x}_{fn};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u)}\left[\tilde{\mathbf{s}}_{fn}\right] \\ &= \mathbb{E}_{p(\mathbf{z}_n|\mathbf{x}_n;\boldsymbol{\theta}_s,\boldsymbol{\theta}_u)}\left[\mathbb{E}_{p(\tilde{\mathbf{s}}_{fn}|\mathbf{x}_n,\mathbf{z}_n;\boldsymbol{\theta}_s,\boldsymbol{\theta}_u)}\left[\tilde{\mathbf{s}}_{fn}\right]\right] \\ &= \mathbb{E}_{p(\mathbf{z}_n|\mathbf{x}_n;\boldsymbol{\theta}_s,\boldsymbol{\theta}_u)}\left[g_n\boldsymbol{\Sigma}_{\mathbf{s},f}(\mathbf{z}_n)\left(\boldsymbol{\Sigma}_{\mathbf{x},fn}(\mathbf{z}_n)\right)^{-1}\right]\mathbf{x}_{fn}. \quad (22)\end{aligned}$$

As before, (22) cannot be computed in an analytical form, and it is approximated with the Metropolis-Hastings algorithm as done in the E-Step. This speech signal estimate corresponds to a probabilistic multichannel Wiener filtering averaged over all possible realizations of the latent variables according to their posterior distribution. Note that the noise signal can be estimated in a similar way.

5. EXPERIMENTS

Baseline method: We compare our method with its NMF counterpart proposed in [6]. In this paper, the multichannel speech signal is modeled by $\mathbf{s}_{fn} \sim \mathcal{N}_c(0,(\mathbf{W}_s\mathbf{H}_s)_{f,n}\mathbf{R}_{\mathbf{s},f})$, where $\mathbf{W}_s \in \mathbb{R}_+^{F\times K_s}$ and $\mathbf{H}_b \in \mathbb{R}_+^{K_s\times N}$. Compared with the proposed model defined in (6), the deep-learning based variance model is replaced by a more conventional NMF model. For this baseline method, we also consider a supervised speech model by learning the NMF dictionary matrix $\mathbf{W}_s$ on the training dataset. The noise model remains unsupervised and is identical to the one defined in (7).

Database: The supervised speech model parameters are learned from the training set of the TIMIT database [22]. It contains almost 4 hours of single-channel 16-kHz speech signals, distributed

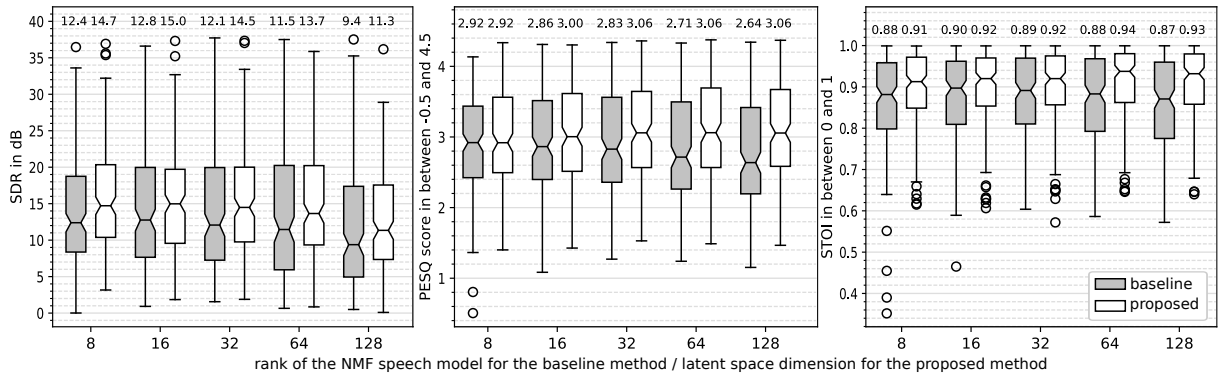


Fig. 2: Speech enhancement results in terms of SDR, PESQ and STOI measures. The value of the median is indicated above each boxplot.

over 462 speakers. For the evaluation of the speech enhancement algorithms, we created 168 stereophonic noisy mixtures of about 3 seconds each. We used the monophonic clean speech signals from the TIMIT test set (speakers and sentences are different than in the training set) and converted them to stereophonic signals as follows: We uniformly drew a direction of arrival within [-90, 90]° and we delayed one channel with respect to the other accordingly, assuming a free field propagation. We then mixed the resulting signals with stereophonic non-stationary noise signals from the DEMAND database [23] (using channels 1 and 3), at a 0 dB signal-to-noise ratio. The different noisy recording environments are domestic, nature, office, indoor public spaces, street and transportation.

Parameters setting: The STFT is computed using a 64-ms sine window (i.e. $F = 513$) with 75%-overlap. The proposed and baseline methods are compared for different settings of the dimension of the latent space L and the rank K_s of the NMF speech model. For fair comparison, we set $L = K_s \in \{8, 16, 32, 64, 128\}$. The rank of the noise model is arbitrarily fixed to $K_b = 10$ for both methods. Unsupervised NMF parameters are randomly initialized and SCMs are initialized with identity matrices. For the proposed method, $\mathbf{g}$ is initialized with an all-ones vector. Similarly as in [13], the Markov chain at the first MCEM iteration is initialized using the recognition network taking the noisy mixture as input. Then, at each new E-Step, we use the last sample drawn at the previous E-Step to initialize the Markov chain. We then run 40 iterations of the Metropolis-Hastings algorithm with $\epsilon^2 = 0.01$, and we discard the first 30 samples as the burn-in period. The algorithms for the proposed and the baseline methods are run for 50 iterations.

Neural network: The generative and recognition networks are represented in Fig. 1. Each network is feed-forward and fully-connected, with hidden layers using rectified linear units (ReLUs) [24] and batch normalization [25]. Output layers use identity activation functions, so they output real-valued coefficients which is the reason why we consider logarithm of variances. The preprocessing layer of the recognition network consists in computing the logarithm of the input speech power spectra and applying a frequency-wise standardization (zero-mean and unit-variance) based on the training set. For jointly learning the parameters of the generative and recognition networks (see Section 2), we use the Adam optimizer [26] with a step size of 10^{-3}, exponential decay rates of 0.9 and 0.999 for the first and second moment estimates, respectively, and an epsilon of 10^{-7} for preventing division by zero. 20% of the TIMIT training set was kept as a validation set, and early stopping with a patience of 10 epochs was used. Weights were initialized using the uniform initializer described in [27]. The network architecture was chosen based on the cost function on the validation set.

Results: The speech enhancement results are presented in Fig. 2 in terms of signal-to-distortion ratio (SDR) [28], perceptual evaluation of speech quality (PESQ) measure [29] and short-time objective intelligibility (STOI) measure [30]. We see that according to all measures, the proposed method outperforms its NMF-based counterpart. In terms of SDR, for the best configuration of the two methods ($L = K_s = 16$), we have a 2.2 dB improvement in median. It is also interesting to observe that all the performance measures for the baseline method tend to decrease as K_s increases (above 16). Increasing the number of parameters makes the optimization problem more difficult to solve. For the proposed method, this behaviour also appears (although to a lesser extent) in terms of SDR only, not in terms PESQ or STOI. Our better performance is however obtained at the expense of a higher computational time. Due to the Monte Carlo E-Step, and with the above-mentioned parameters setting, one iteration of the proposed method is about seven times more expensive than one iteration of the baseline method. For reproducibility, a Python implementation of our algorithm and audio examples are available online.[1]

6. CONCLUSION

In this work, we proposed a multichannel speech enhancement method using a probabilistic speech source model based on variational autoencoders. The noise model was based on unsupervised NMF, which makes the method nicely flexible and able to adapt to any (non-stationary) noisy environment. Experiments demonstrated that the proposed approach outperforms its NMF-based counterpart. The main limitation of this work is the increased computational time. In future work, we will replace the MCEM algorithm by a variational EM algorithm, in order to speed up the inference.

A. INEQUALITIES

Let $\mathbf{\Sigma} \in \mathbb{C}^{I \times I}$ be a positive definite matrix. As $\mathbf{\Sigma} \mapsto \ln\det(\mathbf{\Sigma})$ is a concave function, it can be majorized at an arbitrary point using a first-order Taylor expansion: $\ln\det(\mathbf{\Sigma}) \leq \ln\det(\mathbf{\Sigma}_0) + \mathrm{tr}\left(\mathbf{\Sigma}_0^{-1}\mathbf{\Sigma}\right) - I$, where $\mathbf{\Sigma}_0 \in \mathbb{C}^{I \times I}$ is a positive definite matrix [31, p. 641]. Moreover, equality holds if $\mathbf{\Sigma}_0 = \mathbf{\Sigma}$.

For any positive definite matrix $\mathbf{\Sigma} \in \mathbb{C}^{I \times I}$, and any matrix $\mathbf{A} \in \mathbb{C}^{I \times I}$, $\mathbf{\Sigma} \mapsto \mathrm{tr}\left(\mathbf{\Sigma}^{-1}\mathbf{A}\right)$ is a convex function. Therefore, for any set of Hermitian positive definite matrices $\left\{\mathbf{\Sigma}_k \in \mathbb{C}^{I \times I}\right\}_k$, Jensen's trace inequality gives $\mathrm{tr}\left(\left(\sum_k \mathbf{\Sigma}_k\right)^{-1}\mathbf{A}\right) \leq \sum_k \mathrm{tr}\left(\mathbf{\Sigma}_k^{-1}\mathbf{\Phi}_k\mathbf{A}\mathbf{\Phi}_k^H\right)$, where $\mathbf{\Phi}_k \in \mathbb{C}^{I \times I}$ and $\sum_k \mathbf{\Phi}_k = \mathbf{I}$ [32]. Moreover, equality holds if $\mathbf{\Phi}_k = \mathbf{\Sigma}_k\left(\sum_{k'} \mathbf{\Sigma}_{k'}\right)^{-1}$.

[1] https://team.inria.fr/perception/icassp-2019-mvae/

7. REFERENCES

[1] Philipos C. Loizou, *Speech enhancement: theory and practice*, CRC press, 2007.

[2] Alexey Ozerov and Cédric Févotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, no. 3, pp. 550–563, 2010.

[3] Ngoc Q. K. Duong, Emmanuel Vincent, and Rémi Gribonval, "Underdetermined reverberant audio source separation using a full-rank spatial covariance model," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, no. 7, pp. 1830–1840, 2010.

[4] Simon Arberet, Alexey Ozerov, Ngoc QK Duong, Emmanuel Vincent, Rémi Gribonval, Frédéric Bimbot, and Pierre Vandergheynst, "Nonnegative matrix factorization and spatial covariance model for underdetermined reverberant audio source separation," in *Proc. IEEE Int. Conf. Information Sciences, Signal Process. and Applications (ISSPA)*, 2010, pp. 1–4.

[5] Alexey Ozerov, Emmanuel Vincent, and Frédéric Bimbot, "A general flexible framework for the handling of prior information in audio source separation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 4, pp. 1118–1133, 2012.

[6] Hiroshi Sawada, Hirokazu Kameoka, Shoko Araki, and Naonori Ueda, "Multichannel extensions of non-negative matrix factorization with complex-valued data," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 5, pp. 971–982, 2013.

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

[8] Aditya Arie Nugraha, Antoine Liutkus, and Emmanuel Vincent, "Multichannel audio source separation with deep neural networks," *IEEE Trans. Audio, Speech, Language Process.*, vol. 24, no. 9, pp. 1652–1664, 2016.

[9] Simon Leglaive, Roland Badeau, and Gaël Richard, "Multichannel audio source separation with probabilistic reverberation priors," *IEEE Trans. Audio, Speech, Language Process.*, vol. 24, no. 12, pp. 2453–2465, 2016.

[10] DeLiang Wang and Jitong Chen, "Supervised speech separation based on deep learning: An overview," *IEEE Trans. Audio, Speech, Language Process.*, vol. 26, no. 10, pp. 1702–1726, 2018.

[11] Diederik P. Kingma and Max Welling, "Auto-encoding variational Bayes," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2014.

[12] Yoshiaki Bando, Masato Mimura, Katsutoshi Itoyama, Kazuyoshi Yoshii, and Tatsuya Kawahara, "Statistical speech enhancement based on probabilistic integration of variational autoencoder and nonnegative matrix factorization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2018, pp. 716–720.

[13] Simon Leglaive, Laurent Girin, and Radu Horaud, "A variance modeling framework based on variational autoencoders for speech enhancement," 2018, Proc. IEEE Int. Workshop Machine Learning Signal Process. (MLSP).

[14] Simon Leglaive, Umut Şimşekli, Antoine Liutkus, Laurent Girin, and Radu Horaud, "Speech enhancement with variational autoencoders and alpha-stable distributions," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2019.

[15] Cédric Févotte, Nancy Bertin, and Jean-Louis Durrieu, "Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis," *Neural computation*, vol. 21, no. 3, pp. 793–830, 2009.

[16] Greg C.G. Wei and Martin A. Tanner, "A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms," *Journal of the American statistical Association*, vol. 85, no. 411, pp. 699–704, 1990.

[17] Kouhei Sekiguchi, Yoshiaki Bando, Kazuyoshi Yoshii, and Tatsuya Kawahara, "Bayesian multichannel speech enhancement with a deep speech prior," in *Proc. Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2018, pp. 1233–1239.

[18] F. D. Neeser and J. L. Massey, "Proper complex random processes with applications to information theory," *IEEE Trans. Information Theory*, vol. 39, no. 4, pp. 1293–1302, 1993.

[19] Antoine Liutkus, Roland Badeau, and Gäel Richard, "Gaussian processes for underdetermined source separation," *IEEE Trans. Signal Process.*, vol. 59, no. 7, pp. 3155–3167, 2011.

[20] Christian P. Robert and George Casella, *Monte Carlo Statistical Methods*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

[21] K.S. Chan and Johannes Ledolter, "Monte Carlo EM estimation for time series models involving counts," *Journal of the American Statistical Association*, vol. 90, no. 429, pp. 242–252, 1995.

[22] John S. Garofolo, Lori F. Lamel, William M. Fisher, Jonathan G. Fiscus, David S. Pallett, Nancy L. Dahlgren, and Victor Zue, "TIMIT acoustic phonetic continuous speech corpus," in *Linguistic data consortium*, 1993.

[23] Joachim Thiemann, Nobutaka Ito, and Emmanuel Vincent, "The Diverse Environments Multi-channel Acoustic Noise Database (DEMAND): A database of multichannel environmental noise recordings," in *Proc. Int. Cong. on Acoust.*, 2013.

[24] Vinod Nair and Geoffrey E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proc. Int. Conf. Machine Learning (ICML)*, 2010, pp. 807–814.

[25] Sergey Ioffe and Christian Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *Proc. Int. Conf. Machine Learning (ICML)*, 2015, pp. 448–456.

[26] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2015.

[27] Xavier Glorot and Yoshua Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. Int. Conf. Artif. Intelligence and Stat.*, 2010, pp. 249–256.

[28] Emmanuel Vincent, Rémi Gribonval, and Cédric Févotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, no. 4, pp. 1462–1469, 2006.

[29] Antony W. Rix, John G. Beerends, Michael P. Hollier, and Andries P. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2001, pp. 749–752.

[30] Cees H. Taal, Richard C. Hendriks, Richard Heusdens, and Jesper Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 7, pp. 2125–2136, 2011.

[31] Stephen Boyd and Lieven Vandenberghe, *Convex optimization*, Cambridge university press, 2004.

[32] Frank Hansen and Gert K. Pedersen, "Jensen's operator inequality," *Bulletin of the London Mathematical Society*, vol. 35, no. 4, pp. 553–564, 2003.