Training and Compensation of Class-conditioned NMF Bases for Speech Enhancement ☆

Hanwook Chung[a,*], Roland Badeau[b], Eric Plourde[c], Benoit Champagne[a]

[a]*Department of Electrical and Computer Engineering, McGill University, Montreal, Quebec, Canada*
[b]*LTCI, Télécom ParisTech, Université Paris-Saclay, 75013, Paris, France*
[c]*Department of Electrical and Computer Engineering, Sherbrooke University, Sherbrooke, Quebec, Canada*

Abstract

In this paper, we introduce a training and compensation algorithm of the class-conditioned basis vectors in the non-negative matrix factorization (NMF) model for single-channel speech enhancement. The main goal is to estimate the basis vectors of different signal sources in a way that prevents them from representing other sources, in order to reduce the residual noise components that have features similar to the speech signal. During the proposed training stage, the basis matrices for the clean speech and noises are estimated jointly by constraining them to belong to different classes. To this end, we employ the probabilistic generative model (PGM) of classification, specified by class-conditional densities, as an *a priori* distribution for the basis vectors. The update rules of the NMF and the PGM parameters of classification are jointly obtained by using the variational Bayesian expectation-maximization (VBEM) algorithm, which guarantees convergence to a stationary point. Another goal of the proposed algorithm is to handle a mismatch between the characteristics of the training and test data. This is accomplished during the proposed enhancement stage, where we implement a basis compensation scheme. Specifically, we use extra free basis vectors to capture the features which are not included in the training data. Objective experimental results for different combination of speaker and noise types show that the proposed algorithm can provide better speech enhancement performance than the benchmark algorithms under various conditions.

Keywords: Single-channel speech enhancement, non-negative matrix factorization, probabilistic generative model, classification, variational Bayesian expectation-maximization

1. Introduction

The general objective of speech enhancement algorithms is to remove the background noise from a noisy speech signal to improve its quality or intelligibility. They have been an attractive research area for decades and find various applications including mobile telephony, hearing aid and speech recognition. Numerous single-channel speech enhancement algorithms have been proposed in the past, such as: spectral subtraction [1], minimum mean-square error (MMSE) estimation [2, 3] and subspace decomposition [4]. However, these algorithms tend to provide limited performance in adverse noisy environments, e.g., low input signal-to-noise ratio (SNR) or non-stationary noise conditions, since they use a minimal amount of *a priori* information about the speech and noise. Recently, the non-negative matrix factorization (NMF) approach has been successfully applied to various problems, such as music transcription [5], source separation [6], speech enhancement [7] and image representation [8]. In general, NMF is a dimensionality reduction technique, which decomposes a given matrix into basis and activation matrices with non-negative elements [9, 10]. In audio and speech applications, the magnitude or power spectrum of the (noisy) audio signal is interpreted as a linear combination of the NMF basis vectors, which play a key role in the enhancement process. Deep neural network (DNN) algorithms have gained enormous interest lately. The DNN training aims at estimating the nonlinear mapping function, specified by the weights and biases of the hidden layers, that relates the input features to the output target features. Applications of DNN to speech enhancement and source separation have been introduced in [11, 12, 13]. The NMF and DNN algorithms differ significantly in terms of underlying modeling structure and training requirements; in this paper, we focus on a linear NMF model.

In a supervised NMF-based framework, the basis vectors are typically obtained *a priori* for each source by independently using isolated training data during the training stage. However, there are two main problems in such a framework. The first one is that the basis vectors of the different signal sources, e.g., speech and noise, may share similar characteristics. For example, the basis vectors of the speech spectrum can represent the noise spectrum and hence, the enhanced speech may contain residual noise components which have features similar to the speech signal. One possible remedy is to train the basis vectors of each source in a way that prevents them from representing other sources. In [14], the cross-coherence of the basis vectors is added as a penalty term to the NMF cost function, whereas the cross-reconstruction error terms are considered in

☆Funding for this work was provided by Microsemi Corporation (Ottawa, Canada) and a grant from NSERC (Govt. of Canada)

*Corresponding author, e-mail: hanwook.chung@mail.mcgill.ca (H. Chung)
e-mails: roland.badeau@telecom-paristech.fr (R. Badeau), eric.plourde@usherbrooke.ca (E. Plourde), benoit.champagne@mcgill.ca (B. Champagne)

[15]. The authors in [16, 17, 18] propose to use additional training data which are generated by mixing, e.g., adding or concatenating, the isolated training data of each source. However, the approaches in [16, 17] are based on heuristic multiplicative update (MU) rules which do not guarantee the convergence of the NMF in general [10, 19]. Moreover, the basis vectors in [17, 18] are obtained indirectly by means of the activation matrix estimated from the mixed training data and hence, lack an explicit interpretation in terms of discrimination.

The second problem in a supervised framework is the existence of a mismatch between the characteristics of the training and test data. A common approach to overcome this problem is to add explicit regularization terms to the NMF cost function that incorporate some prior knowledge, such as the temporal continuity [20] or statistical characteristics of the magnitude spectra [21]. In these algorithms, however, the basis vectors are fixed during the enhancement stage, which limits the performance when there is a large mismatch between the training and test data. One alternative approach is to use a basis adaptation scheme during the enhancement stage. In [22], the basis vectors are adapted based on prior distributions modeled by Gamma mixtures. The authors in [23] employ extra validation data for speaker adaptation in a speech-music separation task. In [24], the basis vectors are adapted by using a combination of the original and pre-processed noisy speech samples, the latter being obtained via a classical MMSE-based speech enhancement algorithm. In these algorithms, however, the basis vectors are adapted from the mixtures of multiple sources, e.g., noise and speech, such that the resulting basis vectors may still exhibit features of different sources. Consequently, the enhanced speech may contain some residual noise components and hence, adapting the complete set of basis vectors may limit the enhancement performance.

In this paper, to overcome these limitations, we introduce a training and compensation algorithm of the class-conditioned basis vectors in the NMF model for single-channel speech enhancement, which is an extension of our previous works on training class-conditioned basis vectors in [25], and basis compensation in [26]. In the proposed framework herein, we consider the probabilistic generative model (PGM) of classification specified by class-conditional densities [27], along with the NMF model [28]. Specifically, the PGM of classification is used as an explicit *a priori* distribution for the basis vectors. During the proposed training stage, the basis matrices for all the clean speech and noise sources are estimated jointly by constraining them to belong to one of several speech and noise classes. Previously in [25], we used a traditional Gaussian-distributed class-conditional density [27], and the model parameters were obtained through a maximum *a posteriori* (MAP) estimator using the expectation-maximization (EM) algorithm. In this paper, we make two key modifications. First, we employ a Gamma-distributed class-conditional density to bring more coherence into the NMF model. Second, the update rules of the NMF model and the PGM parameters for classification are jointly obtained via the variational Bayesian expectation-maximization (VBEM) algorithm, which can be considered as an extension of the EM algorithm [27, 28, 29].

The proposed enhancement stage consists of two steps. First, we perform noise classification based on the posterior class probability (PCP), in order to determine which type of noise is included in the noisy speech. Second, we implement a basis compensation algorithm by adopting the approach in [26]. That is, we use extra free basis vectors for both the clean speech and noise to capture the features which cannot be explained by the limited set of basis vectors due to the hard decision on the noise type as well as features which are not included in the training data. The PGM parameters for classification are employed while inferring the free basis vectors as well as during the noise classification. Previously in [26], the free basis vectors were estimated by using the MU rules, whereas we use the VBEM algorithm in this paper. Experimental results of perceptual evaluation of speech quality (PESQ) [30], source-to-distortion ratio (SDR) [31] and segmental SNR (SSNR) show that the proposed algorithm provides better enhancement performance than the benchmark algorithms under various conditions.

The paper is organized as follows. In Section 2, we review the basic principles of supervised NMF-based single-channel speech enhancement. In Section 3, we introduce the PGMs of the NMF and classification models. The proposed training stage is derived in Section 4, and the proposed enhancement stage is explained in Section 5. Experimental results are presented in Section 6 and Section 7 concludes the paper.

2. NMF-based speech enhancement framework

For a given matrix $\mathbf{V} = [v_{kl}] \in \mathbb{R}_+^{K\times L}$, NMF finds a local optimal decomposition of $\mathbf{V} \approx \mathbf{WH}$, where $\mathbf{W} = [w_{km}] \in \mathbb{R}_+^{K\times M}$ is a basis matrix, $\mathbf{H} = [h_{ml}] \in \mathbb{R}_+^{M\times L}$ is an activation matrix, $\mathbb{R}_+$ denotes the set of non-negative real numbers and M is the number of basis vectors, typically chosen such that $M < \min(K, L)$ [19]. The factorization is obtained by minimizing a suitable cost function, such as the Kullback-Leibler (KL) divergence. In this case, the solutions can be obtained iteratively using the following MU rules [9]

$$\mathbf{W} \leftarrow \mathbf{W} \otimes \frac{(\mathbf{V}/(\mathbf{WH}))\mathbf{H}^T}{\mathbf{1}_{KL}\mathbf{H}^T}, \qquad \mathbf{H} \leftarrow \mathbf{H} \otimes \frac{\mathbf{W}^T(\mathbf{V}/(\mathbf{WH}))}{\mathbf{W}^T\mathbf{1}_{KL}} \quad (1)$$

where the operation $\otimes$ denotes element-wise multiplication, $/$ and the quotient line are element-wise division, $\mathbf{1}_{KL}$ is a $K \times L$ matrix with all entries equal to one, the superscript T is the matrix transpose, and $\leftarrow$ refers to an iterative overwrite.

In NMF-based single-channel speech enhancement, one commonly assumes that the magnitude spectrum of the noisy speech, obtained via short-time Fourier transform (STFT), can be approximated by the sum of the clean speech and noise magnitude spectra [6, 7, 32], i.e., $|Y_{kl}| \approx |S_{kl}| + |N_{kl}|$ where Y_{kl}, S_{kl} and N_{kl} respectively denote the STFT coefficients of the noisy speech, clean speech and noise at the frequency bin $k \in \{1, \ldots, K\}$ and time frame $l \in \{1, \ldots, L\}$. Hence, in this work, $\mathbf{V} = [v_{kl}]$ may contain the magnitude spectral values of the noisy speech, clean speech or noise, as indicated by subscripts or superscripts Y, S and N, respectively.

In a supervised framework, $\mathbf{W}_S$ and $\mathbf{W}_N$ are first obtained during the training stage, by applying (1) to the training data $\mathbf{V}_S$ and $\mathbf{V}_N$. In the enhancement stage, for an online application, the activation vector $\mathbf{h}_l^Y = [(\mathbf{h}_l^S)^T\ (\mathbf{h}_l^N)^T]^T \in \mathbb{R}_+^{(M_S+M_N)\times 1}$ is estimated for the l-th time frame by applying the activation update in (1) to $|\mathbf{y}_l| = [|Y_{kl}|] \in \mathbb{R}_+^{K\times 1}$, while fixing $\mathbf{W}_Y = [\mathbf{W}_S\, \mathbf{W}_N]$. In this work, we instead consider a *mini-batch* online application by concatenating several successive time frames of the noisy speech. That is, we construct a target matrix as $\mathbf{V}_{l_b}^Y = |\mathbf{Y}_{l_b}| \in \mathbb{R}_+^{K\times L_b}$, where $l_b = 1, 2, ...$ is the mini-batch index, $\mathbf{Y}_{l_b}$ is the noisy speech matrix consisting of the time frames $l = (l_b-1)L_b+1, ..., l_bL_b$, L_b is the mini-batch size, and $|\cdot|$ denotes the element-wise magnitude computation. The merit of using a mini-batch approach is that we can not only alleviate the over-complete condition (i.e., $M_S + M_N > L_b$) but also reduce the computation time. For a given l_b-th mini-batch, the activation matrix $\mathbf{H}_{l_b}^Y = [(\mathbf{H}_{l_b}^S)^T\ (\mathbf{H}_{l_b}^N)^T]^T \in \mathbb{R}_+^{(M_S+M_N)\times L_b}$ is obtained by applying the activation update in (1) to $\mathbf{V}_{l_b}^Y$. Subsequently, the clean speech spectrum can be estimated using the Wiener filter as [10]

$$\hat{S}_{kl} = \frac{\hat{p}_{kl}^S}{\hat{p}_{kl}^S + \hat{p}_{kl}^N} Y_{kl} \tag{2}$$

where $\hat{p}_{kl}^S$ and $\hat{p}_{kl}^N$ respectively denote the estimated power spectral densities (PSD) of the clean speech and noise. The latter are obtained via temporal smoothing of the NMF-based periodograms as [24, 25]

$$\hat{p}_{kl}^S = \tau_S \hat{p}_{k,l-1}^S + (1-\tau_S)([\mathbf{W}_S\mathbf{H}_{l_b}^S]_{kl})^2 \tag{3}$$

$$\hat{p}_{kl}^N = \tau_N \hat{p}_{k,l-1}^N + (1-\tau_N)([\mathbf{W}_N\mathbf{H}_{l_b}^N]_{kl})^2 \tag{4}$$

where τ_S and τ_N are the smoothing factors for the speech and noise, and $[\cdot]_{kl}$ denotes the (k,l)-th entry of its matrix argument. Finally, the enhanced speech signal in the time-domain is reconstructed by applying the inverse STFT to (2) followed by the overlap-add method.

3. Probabilistic generative models

In this section, we introduce two underlying PGMs for the proposed framework: the PGM of NMF, where the log-likelihood function (LLF) corresponds to the KL-divergence, is described in Section 3.1, while the PGM of classification, which will be applied to the basis vectors, is presented in Section 3.2.

3.1. NMF model

In [28], the NMF model with KL-divergence is described within a statistical framework as summarized below. Each entry of a non-negative matrix, $\mathbf{V} = [v_{kl}]$, is assumed to be a sum of M latent variables as

$$v_{kl} = \sum_{m=1}^{M} c_{kl}^m. \tag{5}$$

The m-th latent variable, c_{kl}^m, is assumed to be drawn from a Poisson distribution parameterized by w_{km} and h_{ml}

$$p(c_{kl}^m | w_{km}, h_{ml}) = \mathcal{P}(c_{kl}^m | w_{km}h_{ml}) \tag{6}$$

where $\mathcal{P}(x|u) = u^x \exp(-u)/(x!)$ is the Poisson distribution with mean u. Note that the approximation of v_{kl} as a sum of integer variables in (5) can be justified by assuming a large dynamic range for the former quantity, which in practice can be realized by a proper scaling of the magnitude spectra [7, 25, 33].

The maximum likelihood (ML) estimates of the parameters w_{km} and h_{ml}, given the observation v_{kl}, are obtained via the EM algorithm. During the expectation step (E-step), the posterior distribution of the latent variable c_{kl}^m given the observation v_{kl} is calculated. In the maximization step (M-step), the parameters are estimated by maximizing the expected complete-data LLF. The iterative NMF solutions obtained through the EM algorithm have forms similar to the MU rules in (1).

3.2. Classification model

In the classification problem, the input vector $\mathbf{w} = [w_k] \in \mathbb{R}^K$ under test is assigned to one of I classes. The essential part of the classification is to find a partition of the observation space $\mathbb{R}^K$ into decision regions that will minimize the classification error, by using training data and their corresponding class labels. There are two main approaches to solve this problem: *PGM* and *discriminative modeling* [27, 34]. The former approach maximizes the likelihood based on the joint distribution of the input data and class labels, whereas the latter maximizes the PCP. In this work, we consider the PGM since it can provide the necessary *a priori* distributions to be used in the proposed training framework.

The PGM can be described by a class-conditional density based on a Gaussian distribution [25, 27] or a Gaussian mixture model [35]. In this work, we instead employ a Gamma distribution, which is shown to be a conjugate prior to the Poisson model [28], to bring more coherence into the NMF model. By ignoring possible correlations between different entries in $\mathbf{w}$, the class-conditional density based on the Gamma distribution can be expressed as

$$p(\mathbf{w}|d_i = 1) = \prod_{k=1}^{K} \mathcal{G}(w_k; \alpha_k^i, \beta_k) \tag{7}$$

where $\mathcal{G}(x; u, z) = x^{u-1}z^{-u}\exp(-x/z)/\Gamma(u)$ is the Gamma distribution with mean uz, $\Gamma(\cdot)$ is the Gamma function, and u and z are referred to as the shape and scale parameters, respectively. Although we can use class-specific scales β_k^i, we consider a common value of β_k for all classes [27], in order to avoid over-fitting.

For a given training set of $\mathbf{W} = [\mathbf{w}_1, ..., \mathbf{w}_M]$ and $\mathbf{D} = [\mathbf{d}_1, ..., \mathbf{d}_M]$, where $\mathbf{d}_m = [d_{im}]$ with $d_{im} \in \{0, 1\}$ (such that $\sum_i d_{im} = 1$) is an $I \times 1$ target class label vector, and assuming the columns $\mathbf{w}_m$ are independently drawn, the likelihood function is given by

$$p(\mathbf{W}, \mathbf{D}; \boldsymbol{\theta}_C) = \prod_{m=1}^{M} \prod_{i=0}^{I-1} \left[p(\mathbf{w}_m|d_i = 1)p_i\right]^{d_{im}} \tag{8}$$

where $\boldsymbol{\theta}_C = \{\{p_i, \{\alpha_k^i\}_{k=1}^K\}_{i=0}^{I-1}, \{\beta_k\}_{k=1}^K\}$ is a PGM parameter set for classification and $p_i \triangleq p(d_i = 1)$ is the prior class probability. The set $\boldsymbol{\theta}_C$ can be simply estimated via the ML criterion. Using Bayes' theorem, the PCP of class i, given the observation $\mathbf{w}$, can be expressed as

$$p(d_i = 1|\mathbf{w}) = \frac{p(\mathbf{w}|d_i = 1)p_i}{\sum_j p(\mathbf{w}|d_j = 1)p_j}. \tag{9}$$

4. Proposed training stage

In many applications of the EM algorithm, evaluating the posterior distribution or indeed computing expectations with respect to this distribution is analytically intractable. Consequently, it is highly demanding to derive a lower bound for the marginal likelihood of the observed data or to estimate the hyper-parameters. The VBEM algorithm overcomes this difficulty by computing an analytical and efficient approximation to the posterior distribution [27, 29], and also provides an effective estimation of the hyper-parameters. In general, the VBEM algorithm can be considered as an extension of the EM algorithm from the ML or MAP estimation of the single most probable value of each parameter to fully Bayesian estimation in which any unknown parameter is absorbed into the set of latent variables. We employ the VBEM method to develop the proposed training algorithm, as further explained below.

4.1. Prior structures

We first explicitly address the prior structures for the PGM in (6), which will be used in the proposed framework. We denote by M_i the number of basis vectors in class i (such that $M = \sum_i M_i$), and by L_i the number of time frames in class i. For the basis vectors, the likelihood function $p(\mathbf{W}, \mathbf{D}; \boldsymbol{\theta}_C)$ in (8), based on the class-conditional density given by (7), can be simply rearranged as

$$p(\mathbf{W}; \boldsymbol{\theta}_C) = \prod_{i=0}^{I-1} \prod_{m=1}^{M_i} \prod_{k=1}^{K} p_i \mathcal{G}(w_{km}^i; \alpha_k^i, \beta_k) \tag{10}$$

where we omit the dependence on $\mathbf{D}$ hereafter for convenience. For the activations, we follow the prior model based on the Gamma distribution as introduced in [7, 28]:

$$p(h_{ml}^i; a_{ml}^i, b_{ml}^i) = \mathcal{G}\left(h_{ml}^i; a_{ml}^i, \frac{b_{ml}^i}{a_{ml}^i}\right) \tag{11}$$

which provides an intuitive interpretation in terms of the mean value simply given by b_{ml}^i. Moreover, we consider constant values of a_{ml}^i and b_{ml}^i for each class, i.e., $a_{ml}^i = a^i$ and $b_{ml}^i = b^i$, to avoid over-fitting [7, 28]. Assuming that the entries of $\mathbf{H}$ are independently distributed, the prior of $\mathbf{H}$ can be written as

$$p(\mathbf{H}; \mathbf{a}, \mathbf{b}) = \prod_{i=0}^{I-1} \prod_{m=1}^{M_i} \prod_{l=1}^{L_i} p(h_{ml}^i; a^i, b^i) \tag{12}$$

where $\mathbf{a} = \{a^i\}_{i=0}^{I-1}$ and $\mathbf{b} = \{b^i\}_{i=0}^{I-1}$. Note that employing the prior structure in (11) for the basis vectors indicates the class-specific scales in the PGM for classification and hence, limits the enhancement performance due to over-fitting.

4.2. VBEM algorithm

Let us denote by $\boldsymbol{\theta}_L = \{\mathbf{C}, \mathbf{W}, \mathbf{H}\}$ the set of latent variables, where $\mathbf{C} = \{c_{kl}^{m,i}\}$, $\mathbf{W} = \{w_{km}^i\}$, $\mathbf{H} = \{h_{ml}^i\}$, and by $\boldsymbol{\theta}_R = \{\boldsymbol{\theta}_C, \mathbf{a}, \mathbf{b}\}$ the set of hyper-parameters. In the proposed framework, we use the class index $i = 0$ for the speech and $i = 1, ..., I-1$ for the different noise types. For given training data sets of the clean speech and noise, $\mathbf{V} = \{\mathbf{V}^i\}$, the marginal LLF can be written as

$$\begin{aligned}
\ln p(\mathbf{V}; \boldsymbol{\theta}_R) &\geq \sum_{\mathbf{C}} \iint q(\mathbf{C}, \mathbf{W}, \mathbf{H}) \ln \frac{p(\mathbf{V}, \mathbf{C}, \mathbf{W}, \mathbf{H}; \boldsymbol{\theta}_R)}{q(\mathbf{C}, \mathbf{W}, \mathbf{H})} d\mathbf{W} d\mathbf{H} \\
&= \underbrace{\mathbb{E}_{q(\boldsymbol{\theta}_L)}[\ln p(\mathbf{V}, \boldsymbol{\theta}_L; \boldsymbol{\theta}_R)]}_{\triangleq \mathcal{L}_V(q(\boldsymbol{\theta}_L); \boldsymbol{\theta}_R)} - \underbrace{\mathbb{E}_{q(\boldsymbol{\theta}_L)}[\ln q(\boldsymbol{\theta}_L)]}_{\triangleq -\mathcal{L}_E(q(\boldsymbol{\theta}_L))} \\
&\triangleq \mathcal{L}_B(q(\boldsymbol{\theta}_L); \boldsymbol{\theta}_R)
\end{aligned} \tag{13}$$

where $q(\cdot)$ is an arbitrary distribution (often referred to as a *variational distribution*), $\mathbb{E}_{g(x)}[f(x)]$ indicates an expectation of $f(x)$ with respect to $g(x)$. The term $\mathcal{L}_B(q(\boldsymbol{\theta}_L); \boldsymbol{\theta}_R)$ defines the lower bound on $\ln p(\mathbf{V}; \boldsymbol{\theta}_R)$, where the equality holds for $q(\boldsymbol{\theta}_L) = p(\boldsymbol{\theta}_L|\mathbf{V}; \boldsymbol{\theta}_R)$ [27, 28]. A detailed expression of the lower bound is given in Appendix A. Analogous to the EM algorithm, the VBEM algorithm consists of two stages. During the E-step, the goal is to estimate $q(\boldsymbol{\theta}_L)$ which approximates the exact posterior distribution $p(\boldsymbol{\theta}_L|\mathbf{V}; \boldsymbol{\theta}_R)$. In the M-step, the hyper-parameters are obtained by maximizing the lower bound in (13) computed with a *fixed* $q(\boldsymbol{\theta}_L)$. That is, the term $\mathcal{L}_E(q(\boldsymbol{\theta}_L))$, which denotes the *entropy* of $q(\boldsymbol{\theta}_L)$, can be considered as a constant value and hence, maximizing the lower bound becomes equivalent to maximizing the *energy* $\mathcal{L}_V(q(\boldsymbol{\theta}_L); \boldsymbol{\theta}_R)$.

1) Variational E-step: Based on the *mean-field* approximation [27, 29], we assume that $q(\mathbf{C}, \mathbf{W}, \mathbf{H})$ can be factorized as (e.g., [28, 32, 36])

$$\begin{aligned}
q(\mathbf{C}, \mathbf{W}, \mathbf{H}) &= q(\mathbf{C})q(\mathbf{W})q(\mathbf{H}) \\
&= \Big(\prod_{i,k,l} q(\mathbf{c}_{kl}^i)\Big)\Big(\prod_{i,k,m} q(w_{km}^i)\Big)\Big(\prod_{i,m,l} q(h_{ml}^i)\Big)
\end{aligned} \tag{14}$$

where $\mathbf{c}_{kl}^i = [c_{kl}^{1,i}, ..., c_{kl}^{M_i,i}]$. The resulting local optimal solutions can be found as [27, 28]:

$$q(\mathbf{C})^{(r+1)} \propto \exp\Big(\mathbb{E}_{q(\mathbf{W})^{(r)} q(\mathbf{H})^{(r)}}[\ln p(\mathbf{V}, \boldsymbol{\theta}_L; \boldsymbol{\theta}_R)]\Big) \tag{15}$$

$$q(\mathbf{W})^{(r+1)} \propto \exp\Big(\mathbb{E}_{q(\mathbf{C})^{(r+1)} q(\mathbf{H})^{(r)}}[\ln p(\mathbf{V}, \boldsymbol{\theta}_L; \boldsymbol{\theta}_R)]\Big) \tag{16}$$

$$q(\mathbf{H})^{(r+1)} \propto \exp\Big(\mathbb{E}_{q(\mathbf{C})^{(r+1)} q(\mathbf{W})^{(r+1)}}[\ln p(\mathbf{V}, \boldsymbol{\theta}_L; \boldsymbol{\theta}_R)]\Big) \tag{17}$$

where the superscript (r) denotes the r-th iteration. For convenience, we hereafter omit the superscript (r) and also drop the latent variables inside the subscript $q(\cdot)$ of the expectation operator, e.g., $\mathbb{E}_{q(w_{km}^i)}[w_{km}^i] = \mathbb{E}_q[w_{km}^i]$.

First, the distribution $q(\mathbf{c}_{kl}^i)$ in (15) is shown to follow a multinomial distribution [28]:

$$\mathcal{M}(\mathbf{c}_{kl}^i; v_{kl}^i, \bar{\mathbf{p}}_{kl}^i) = \delta\left(v_{kl}^i - \sum_{m=1}^{M_i} c_{kl}^{m,i}\right) v_{kl}^i! \prod_{m=1}^{M_i} \frac{(\bar{p}_{kl}^{m,i})^{c_{kl}^{m,i}}}{c_{kl}^{m,i}!} \tag{18}$$

where $\delta(x)$ is the Kronecker delta function defined by $\delta(x)=1$ when $x=0$ and $\delta(x)=0$ otherwise. The entries of $\bar{\mathbf{p}}_{kl}^{i}=[\bar{p}_{kl}^{m,i}]$ are given by

$$\bar{p}_{kl}^{m,i}=\frac{\exp\left(\mathbb{E}_q[\ln w_{km}^{i}]+\mathbb{E}_q[\ln h_{ml}^{i}]\right)}{\sum_{m=1}^{M_i}\exp\left(\mathbb{E}_q[\ln w_{km}^{i}]+\mathbb{E}_q[\ln h_{ml}^{i}]\right)}. \tag{19}$$

Next, the distribution $q(w_{km}^{i})$ in (16) is obtained as

$$\begin{aligned} q(w_{km}^{i}) &\propto \exp\Bigg[\Big(\alpha_k^{i}+\sum_{l=1}^{L_i}\mathbb{E}_q[c_{kl}^{m,i}]-1\Big)\ln w_{km}^{i} \\ &\quad -\Big(\frac{1}{\beta_k}+\sum_{l=1}^{L_i}\mathbb{E}_q[h_{ml}^{i}]\Big)w_{km}^{i}\Bigg] \\ &\propto \mathcal{G}(w_{km}^{i};\bar{\alpha}_{km}^{i},\bar{\beta}_{km}^{i}) \end{aligned} \tag{20}$$

where the parameters are given by

$$\bar{\alpha}_{km}^{i}=\alpha_k^{i}+\sum_{l=1}^{L_i}\mathbb{E}_q[c_{kl}^{m,i}],\ \ \bar{\beta}_{km}^{i}=\Big(\frac{1}{\beta_k}+\sum_{l=1}^{L_i}\mathbb{E}_q[h_{ml}^{i}]\Big)^{-1}. \tag{21}$$

Finally, the distribution $q(h_{ml}^{i})$ in (17) is also found to follow a Gamma distribution $\mathcal{G}(h_{ml}^{i};\bar{a}_{ml}^{i},\bar{b}_{ml}^{i})$ [28], where the parameters are given by

$$\bar{a}_{ml}^{i}=a^{i}+\sum_{k=1}^{K}\mathbb{E}_q[c_{kl}^{m,i}],\ \ \bar{b}_{ml}^{i}=\Big(\frac{a^{i}}{b^{i}}+\sum_{k=1}^{K}\mathbb{E}_q[w_{km}^{i}]\Big)^{-1}. \tag{22}$$

The sufficient statistics (expectations) are given below:

$$\mathbb{E}_q[c_{kl}^{m,i}]=v_{kl}^{i}\bar{p}_{kl}^{m,i} \tag{23}$$

$$\mathbb{E}_q[\ln w_{km}^{i}]=\Psi(\bar{\alpha}_{km}^{i})+\ln\bar{\beta}_{km}^{i},\ \ \mathbb{E}_q[w_{km}^{i}]=\bar{\alpha}_{km}^{i}\bar{\beta}_{km}^{i} \tag{24}$$

$$\mathbb{E}_q[\ln h_{ml}^{i}]=\Psi(\bar{a}_{ml}^{i})+\ln\bar{b}_{ml}^{i},\ \ \mathbb{E}_q[h_{ml}^{i}]=\bar{a}_{ml}^{i}\bar{b}_{ml}^{i} \tag{25}$$

where $\Psi(x)=d\ln\Gamma(x)/dx$ is the digamma function [28].

2) Variational M-step: The hyper-parameter set $\boldsymbol{\theta}_R$ is estimated by maximizing $\mathcal{L}_V(q(\boldsymbol{\theta}_L)^{(r+1)};\boldsymbol{\theta}_R)$. Setting the partial derivative of $\mathcal{L}_V(q(\boldsymbol{\theta}_L)^{(r+1)};\boldsymbol{\theta}_R)$ with respect to $\boldsymbol{\theta}_R$ to zero, the PGM parameters for classification, $\boldsymbol{\theta}_C$, are obtained as

$$\alpha_k^{i}\leftarrow\alpha_k^{i}-\frac{\Psi(\alpha_k^{i})-\alpha_q^{i}}{\Psi'(\alpha_k^{i})} \tag{26}$$

$$\beta_k=\frac{\sum_{i=0}^{I-1}\sum_{m=1}^{M_i}\mathbb{E}_q[w_{km}^{i}]}{\sum_{i=0}^{I-1}M_i\alpha_k^{i}} \tag{27}$$

where $\alpha_q^{i}=\sum_{m=1}^{M_i}(\mathbb{E}_q[\ln w_{km}^{i}]-\ln\beta_k)/M_i$, and $\Psi'(x)$ is the derivative of the digamma function $\Psi(x)$ with respect to x, i.e., $\Psi'(x)=d\Psi(x)/dx$. The prior class probability is simply estimated by $p_i=M_i/M$. The shape and scale parameters, **a** and **b**, are obtained as in [28]:

$$a^{i}\leftarrow a^{i}-\frac{\ln a^{i}-\Psi(a^{i})+1-a_q^{i}}{1/a^{i}-\Psi'(a^{i})} \tag{28}$$

$$b^{i}=\frac{1}{M_iL_i}\sum_{m=1}^{M_i}\sum_{l=1}^{L_i}\mathbb{E}_q[h_{ml}^{i}] \tag{29}$$

where $a_q^{i}=\sum_m\sum_l(\mathbb{E}_q[h_{ml}^{i}]/b^{i}-\mathbb{E}_q[\ln h_{ml}^{i}]+\ln b^{i})/(KM_i)$.

To avoid scale indeterminacies in w_{km} and h_{ml} which appear as a product in the distribution (6), we include a normalization step. Motivated by [37], we normalize $\mathbb{E}_q[w_{km}^{i}]$ and $\exp(\mathbb{E}_q[\ln w_{km}^{i}])$ such that they sum up to 1 with respect to k after computing (20). For initialization, we generate positive random numbers and subsequently apply the MU rules in (1) to $\mathbf{V}$ for several iterations [7, 32], where we found that 10 iterations are sufficient. The resulting $\mathbf{W}^{i}$ and $\mathbf{H}^{i}$ are used as the initial values for the sufficient statistics, i.e., $\mathbb{E}_q[w_{km}^{i}]$, $\mathbb{E}_q[\ln w_{km}^{i}]$, $\mathbb{E}_q[h_{ml}^{i}]$ and $\mathbb{E}_q[\ln h_{ml}^{i}]$. To initialize $\boldsymbol{\theta}_C$, we apply (26) and (27) to the initial values of $\mathbb{E}_q[w_{km}^{i}]$ and $\mathbb{E}_q[\ln w_{km}^{i}]$. The shape and scale parameters for the activations are initialized by $a^{i}=0.001$ and $b^{i}=10$. We use 200 iterations for the VBEM algorithm, whereas 5 iterations are used for estimating the hyper-parameters in (26) and (28).

The proposed training stage can be interpreted as follows. During the E-step, the basis vectors are adjusted based on their priors which define the classification boundaries. Hence, the basis vectors are estimated by constraining them to belong to different classes. During the M-step, the hyper-parameters (i.e., the PGM parameters for classification $\boldsymbol{\theta}_C$) are re-estimated, which define new classification boundaries.

5. Proposed enhancement stage

A number of attempts of combining the classical speech enhancement algorithms and the NMF-based framework have been made in the literature. In [24, 26, 38], a classical method is used as a pre-processor to first remove some stationary background noise, and the NMF-based algorithm is subsequently applied to further improve the enhancement performance. The authors in [39] implement the classical and NMF-based algorithms independently, and evaluate the geometric mean over them to estimate the clean speech spectrum. We combine both approaches and propose to use the weighted geometric mean (WGM) of the pre-processed noisy speech and its improvement via Wiener filtering. Regarding the pre-processor, we use the well-knwon MMSE short-time spectral amplitude (STSA) estimator [2], where the noise PSD is estimated based on [40]. The proposed enhancement stage consists of two steps, i.e., noise classification followed by basis compensation, which are explained in the following subsections. We denote by $\bar{\mathbf{S}}_{l_b}\in\mathbb{C}^{K\times L_b}$ the pre-processed noisy speech and by $\bar{\mathbf{N}}_{l_b}=\mathbf{Y}_{l_b}-\bar{\mathbf{S}}_{l_b}$ the pre-estimated noise.

5.1. Noise classification

In many NMF-based speech enhancement algorithms, the background noise type is assumed to be known *a priori*. In the proposed framework, we perform noise classification for the l_b-th mini-batch, to select a single noise type among different classes which has features similar to the noise included in the noisy speech. To this end, one possible approach is to apply the activation update given by (1) to $|\mathbf{Y}_{l_b}|$ for each noise type by fixing its corresponding basis matrix and observe the reconstruction error (i.e., KL-divergence), such as in [41]. However,

this method requires additional iterations in which the computational cost increases with respect to the number of noise types.

In the proposed method, we use the PGM-based classifier given by (9). That is, we evaluate the PCP based on (9) and $\boldsymbol{\theta}_C$ for $i = \{1, ...I-1\}$, and select the noise type with the highest PCP value. As a simple approach, we can first estimate a noise classification basis vector $\mathbf{w}_C = [w_k^C] \in \mathbb{R}_+^K$ by applying the MU rules in (1) to $|\bar{\mathbf{N}}_{l_b}|$, and use it as the input to the classifier. However, we can further reduce the computational cost by simply using the $|\bar{\mathbf{N}}_{l_b}|$ due to the property of NMF (i.e., the target matrix is represented as a linear combination of the basis vectors), since we can avoid additional iterations for computing $\mathbf{w}_C$. To further improve the classification performance, we consider both $\mathbf{Y}_{l_b}$ and $\bar{\mathbf{N}}_{l_b}$. That is, we compute the geometric mean of the magnitude spectra of the noisy speech and pre-estimated noise (i.e., $|\mathbf{Y}_{l_b} \otimes \bar{\mathbf{N}}_{l_b}|^{1/2} \in \mathbb{R}^{K \times L_b}$), to amplify the noise components. Subsequently, we average over the rows and normalize the resulting column vector using the l_1-norm, where the corresponding vector will be denoted by $\tilde{\mathbf{w}}_C \in \mathbb{R}_+^K$.

Regarding the classifier, we found that employing the Gamma distribution in (7) directly for computing the PCP resulted in poor classification performance. One main reason is that the Gamma distribution can lead to numerical instability, since $\Gamma(\alpha)$ rapidly approaches infinity as α increases. Hence, we instead use the approximated Gaussian distribution[1] as the class-conditional density, which is indeed simpler to compute than the Gamma distribution:

$$p(\tilde{\mathbf{w}}_C | d_i = 1) \approx \mathcal{N}(\tilde{\mathbf{w}}_C; \tilde{\boldsymbol{\mu}}_i, \tilde{\boldsymbol{\Sigma}}_i) \tag{30}$$

where $\tilde{\boldsymbol{\mu}}_i = [\tilde{\mu}_{ik}]$ and $\tilde{\boldsymbol{\Sigma}}_i = \text{diag}\{\tilde{\sigma}_{ik}^2\}$ are the mean vector and diagonal covariance matrix of the Gaussian distribution with entries $\tilde{\mu}_{ik} = \alpha_k^i \beta_k$ and $\tilde{\sigma}_{ik}^2 = \alpha_k^i \beta_k^2$. The underlying motivation for using the form in (30) is similar to the application of the Laplace approximation [27], which aims at finding a Gaussian approximation to the original distribution. According to this approach, the mean and variance of the approximated Gaussian distribution are obtained based on the mode and second order derivative at the mode of the original distribution, respectively. However, since the mode of the Gamma distribution is defined only for $\alpha > 1$, we instead use its mean and variance. Furthermore, we use the average value of $\tilde{\boldsymbol{\Sigma}}_i$ over all i for the covariance in (30), which leads to computing the (exponential of the squared) Mahalanobis distance. The latter is known to further reduce the computational cost compared to using the Gaussian model with class-specific variances [42].

5.2. Basis compensation

Once the noise type is determined, we implement a basis compensation scheme by adopting the approach proposed in [26]. That is, we use extra free basis vectors for both the clean speech and noise to capture the features which cannot be explained by the limited set of basis vectors due to the hard decision on a single noise type, as well as features which are not included in the training data. We denote by $\mathbf{W}_{l_b}^{SF} = [w_{km}^{SF}] \in \mathbb{R}_+^{K \times M_{SF}}$ and $\mathbf{W}_{l_b}^{NF} = [w_{km}^{NF}] \in \mathbb{R}_+^{K \times M_{NF}}$ (such that $M_{SF} < M_S$ and $M_{NF} < M_N$) the free basis matrices of the clean speech and noise, respectively.

For the l_b-th mini-batch, motivated by [24] and [26], we aim at factorizing $\mathbf{V}_{l_b} = [|\mathbf{Y}_{l_b}| \; |\bar{\mathbf{S}}_{l_b}|] \in \mathbb{R}_+^{K \times 2L_b}$ into the product of $\mathbf{W}_{l_b} = [\mathbf{W}_S \; \mathbf{W}_{l_b}^{SF} \; \mathbf{W}_N \; \mathbf{W}_{l_b}^{NF}] = [w_{km}] \in \mathbb{R}_+^{K \times M_Y}$ and $\mathbf{H}_{l_b} = [\mathbf{H}_{l_b}^Y \; \mathbf{H}_{l_b}^{\bar{S}}] = [h_{ml}] \in \mathbb{R}_+^{M_Y \times 2L_b}$, where $M_Y = M_S + M_{SF} + M_N + M_{NF}$. We use the VBEM algorithm introduced in Section IV, to estimate the variational distributions $q(\mathbf{W}_{l_b}^{SF})$, $q(\mathbf{W}_{l_b}^{NF})$ and $q(\mathbf{H}_{l_b})$. At each iteration, the distribution $q(\mathbf{C})$ is first inferred as (18), where the parameters are given by (19). Second, we estimate the parameters of $q(\mathbf{W}_{l_b}^{SF})$ and $q(\mathbf{W}_{l_b}^{NF})$, while fixing the parameters of $q(\mathbf{W}_S)$ and $q(\mathbf{W}_N)$. Specifically, the parameters of $q(w_{km}^{SF})$ and $q(w_{km}^{NF})$, which correspond to the ones in $q(w_{km})$ for the intervals $M_S < m \leq M_S + M_{SF}$ and $M_S + M_{SF} + M_N < m \leq M_Y$, respectively, are computed based on (21). The parameters of $q(\mathbf{H}_{l_b})$ are then simply obtained by using (22). Subsequently, the mean value of the noisy speech activation prior b_{l_b} is obtained by applying (29) to $\mathbf{H}_{l_b}$ as

$$b_{l_b} = \frac{1}{2M_Y L_b} \sum_{m=1}^{M_Y} \sum_{l=1}^{2L_b} \mathbb{E}_q[h_{ml}]. \tag{31}$$

In contrast to the scale parameter, we fix the shape parameters (i.e., $a_{l_b}^S$ and $a_{l_b}^N$) as in [7], which controls the degree of sparsity [28], mainly to reduce the computational cost since their updates require additional iterations as given by (28).

After estimating $q(\mathbf{W}_{l_b}^{SF})$, $q(\mathbf{W}_{l_b}^{NF})$ and $q(\mathbf{H}_{l_b})$, we compute the smoothed PSDs of the clean speech and noise based on (3) and (4), where the periodograms are obtained from the mean values[2] of $q(\mathbf{W}_{l_b})$ and $q(\mathbf{H}_{l_b})$. Specifically, the mini-batch clean speech PSD, $\hat{\mathbf{P}}_{l_b}^S = [\hat{p}_{kl}^S] \in \mathbb{R}_+^{K \times L_b}$, is computed by replacing $\mathbf{W}_S$ with $[\mathbb{E}_q[\mathbf{W}_S] \; \mathbb{E}_q[\mathbf{W}_{l_b}^{SF}]] \in \mathbb{R}_+^{K \times (M_S + M_{SF})}$ and $\mathbf{H}_{l_b}^S$ with the first $M_S + M_{SF}$ rows of $\mathbb{E}_q[\tilde{\mathbf{H}}_{l_b}] = (\mathbb{E}_q[\mathbf{H}_{l_b}^Y] + \mathbb{E}_q[\mathbf{H}_{l_b}^{\bar{S}}])/2 \in \mathbb{R}_+^{M_Y \times L_b}$. A similar procedure is carried out for the mini-batch noise PSD $\hat{\mathbf{P}}_{l_b}^N = [\hat{p}_{kl}^N] \in \mathbb{R}_+^{K \times L_b}$. Then, we estimate the clean speech spectrum where the magnitude is obtained via the WGM of $|\bar{\mathbf{S}}_{l_b}|$ and Wiener-filtered $|\bar{\mathbf{S}}_{l_b}|$, and the phase is taken from the noisy speech. Since $\angle \mathbf{Y}_{l_b} = \angle \bar{\mathbf{S}}_{l_b}$ [2], the enhanced speech spectrum can be written as

$$\begin{aligned} \hat{\mathbf{S}}_{l_b} &= \left(\left| \bar{\mathbf{S}}_{l_b} \right|^{\nu_{l_b}} \otimes \left| \frac{\hat{\mathbf{P}}_{l_b}^S}{\hat{\mathbf{P}}_{l_b}^S + \hat{\mathbf{P}}_{l_b}^N} \otimes \bar{\mathbf{S}}_{l_b} \right|^{1-\nu_{l_b}} \right) \otimes e^{j \angle \mathbf{Y}_{l_b}} \\ &= \left(\frac{\hat{\mathbf{P}}_{l_b}^S}{\hat{\mathbf{P}}_{l_b}^S + \hat{\mathbf{P}}_{l_b}^N} \right)^{1-\nu_{l_b}} \otimes \bar{\mathbf{S}}_{l_b} \end{aligned} \tag{32}$$

[1] Note that this approximation is employed only for the noise classification. The inference on $q(w_{km}^i)$ does not suffer from the extreme value of the Gamma function, i.e., the extreme value of the digamma function ($-\infty$) appearing in $\mathbb{E}_q[\ln(\cdot)]$ in (24) and (25) is handled by the exponential in (19).

[2] Alternatively, based on [7], we can compute the smoothed PSD based on the sufficient statistics of $c_{kl}^{m,i}$ in (23) where $\bar{p}_{kl}^{m,i}$ is given by (19). However, we verified through experiments that using $\mathbb{E}_q[w_{km}^i]$ provided better enhancement performance as well as reduced computational cost.

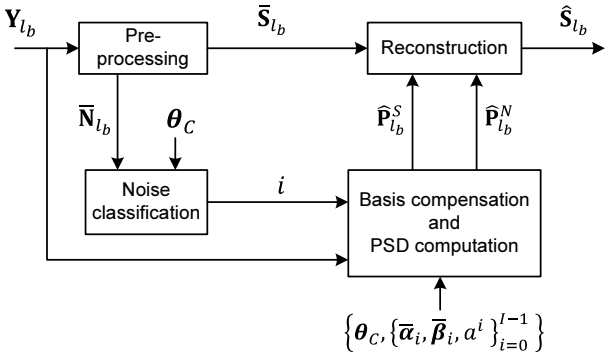

Figure 1: A simplified block diagram of the proposed VNCP-BC method.

where $0 \leq \nu_{l_b} \leq 1$ is the weighting factor. The motivation of using the WGM is to control the effect of pre-processing. For a high input SNR, for instance, the classical method tends to show a reasonable enhancement performance, which implies that Wiener filtering the pre-processed signal may further distort the enhanced speech quality. Hence, it is necessary to put more weight on $\bar{\mathbf{S}}_{l_b}$ by selecting a large ν_{l_b}. In contrast, the classical method results in a poor enhanced speech quality for a low input SNR and hence, further improvement is necessary. This can be specified by applying more weight on the Wiener-filtered $\bar{\mathbf{S}}_{l_b}$ by selecting a small ν_{l_b}. Based on these aspects, we use the logistic function for selecting ν_{l_b}:

$$\nu_{l_b} = \frac{\rho_1}{1 + \exp(-\rho_2 R_{l_b})} \tag{33}$$

where $R_{l_b} = 10\log_{10}(\sum_k \sum_{l \in l_b} \hat{P}^S_{kl} / \sum_k \sum_{l \in l_b} \hat{P}^N_{kl})$ is the estimated input SNR in dB for the l_b-th mini-batch. The parameters ρ_1 and ρ_2 respectively define the range of $\nu_{l_b} \in (0, \rho_1)$ and the slope of the sigmoid function, where we use $\rho_1 = 1$ and $\rho_2 = 0.5$ through the experiments.

For the l_b-th mini-batch, the parameters of $q(\mathbf{W}^{NF}_{l_b})$ are initialized by applying the NMF algorithm to $|\bar{\mathbf{N}}_{l_b}|$ for 2 iterations. Specifically, since $M_{NF} > L_b$ (i.e., over-complete), we use the sparse NMF algorithm which is simply implemented by adding the sparsity parameter (we use 0.5) to the denominator of the activation update in (1). In contrast, the parameters of $q(\mathbf{W}^{SF}_{l_b})$ are initialized from the ones estimated in the previous mini-batch frame index. The parameters of $q(\mathbf{H}_{l_b})$ are initialized by generating positive random numbers. We use 5 iterations for the VBEM algorithm.

The proposed algorithm, i.e., variational inference on the NMF model based on class probabilities and basis compensation, will be referred to as VNCP-BC. A simplified block diagram of the proposed method is illustrated in Figure 1, while the algorithm is summarized in Table 1. Recall that the terms $\bar{\boldsymbol{\alpha}}_i = [\bar{\alpha}^i_{km}] \in \mathbb{R}^{K \times M_i}$ and $\bar{\boldsymbol{\beta}}_i = [\bar{\beta}^i_{km}] \in \mathbb{R}^{K \times M_i}$ represent the parameters of the variational distribution in (20), and the sets $\boldsymbol{\theta}_C$ and $\{a^i\}$ respectively denote the PGM parameters for classification and the shape parameter in the activation prior.

Table 1: Algorithm summary of the proposed enhancement stage

for $l_b = 1, 2, ...$
  Estimate $\bar{\mathbf{S}}_{l_b}$ and $\bar{\mathbf{N}}_{l_b} = \mathbf{Y}_{l_b} - \bar{\mathbf{S}}_{l_b}$
  if $l_b = 1$
    Initialize $\hat{p}^S_{k,0} = \sum_l |\bar{S}_{kl}|^2 / L_b$ and $\hat{p}^N_{k,0} = \sum_l |\bar{N}_{kl}|^2 / L_b$
    Initialize $q(\mathbf{W}^{SF}_{l_b-1})$ parameters by applying sparse NMF to $|\bar{\mathbf{S}}_{l_b}|$
  end
  Compute $\tilde{\mathbf{w}}_C$ by averaging and normalizing $|\mathbf{Y}_{l_b} \otimes \bar{\mathbf{N}}_{l_b}|^{1/2}$
  Select noise type $i \in \{1, ..., I-1\}$ via (9) and (30)
  Initialize $q(\mathbf{W}^{SF}_{l_b})$ parameters by the one estimated at $l_b - 1$
  Initialize $q(\mathbf{W}^{NF}_{l_b})$ parameters by applying sparse NMF to $|\bar{\mathbf{N}}_{l_b}|$
  Initialize $q(\mathbf{H}_{l_b})$ parameters by generating positive random numbers
  for iter = 1:itermax
    Estimate $q(\mathbf{W}^{SF}_{l_b})$ and $q(\mathbf{W}^{NF}_{l_b})$ and normalize
    Estimate $q(\mathbf{H}_{l_b})$
    Update b_{l_b} via (31)
  end
  Compute $\hat{\mathbf{P}}^S_{l_b} = [\hat{p}^S_{kl}]$ and $\hat{\mathbf{P}}^N_{l_b} = [\hat{p}^N_{kl}]$
  Compute ν_{l_b} via (33) and estimate $\hat{\mathbf{S}}_{l_b}$ via (32)
end

6. Experiments

The enhancement performance of the proposed method was assessed through objective experiments. Below, after describing the general methodology and benchmark algorithms, we present and discuss the experimental results.

6.1. Methodology

We conducted the experiments using the 4th CHiME challenge corpus [43]. The speech and noise files were divided into two disjoint groups: i) *training data*, used for estimating the basis matrix for each class i during the training stage, and ii) *test data*, used during the enhancement stage to evaluate the enhancement performance. The clean speech training data of the CHiME database are from the Wall Street Journal (WSJ0) corpus, which consists of 101 speakers. We considered a speaker-independent (SI) application, where one *universal* basis matrix covering all speakers is estimated during the training stage. To this end, we randomly selected 40 utterances from each speaker and concatenated them to construct the clean speech training data ($i = 0$), resulting in a total of 8 hours long signal. Regarding the noise training data, we selected the Bus ($i = 1$), Pedestrian ($i = 2$) and Street ($i = 3$) noises, where each noise type consists of 2 hours long signal.

We used the reference clean speech from the test set of the CHiME corpus, which consists of 330 utterances. Regarding the test data for the noise signals, we categorized them into two groups, referred to as: *matched* and *mismatched* cases. The matched case assumes that the training data is available, whereas the purpose of the mismatched case is to evaluate the enhancement performance for an *unseen* noise type, i.e., when no training data is available. For both the matched and mismatched cases, we performed noise classification to select a single noise type which has characteristics similar to the actual noise included in the noisy speech.

We considered two types of the noisy speech signals for the test: *additive noise* and *simulated noisy speech*. The noisy

Table 2: Summary of the test noise types

	Additive	Simulated
Matched	Bus, Pedestrian, Street (from CHiME)	
Mismatched	Cafe (from CHiME), Factory 1, Babble (from NOISEX)	Cafe (from CHiME)

speech signals for the former type were generated by scaling and adding the noise to the clean speech to obtain input SNRs of -5, 0, 5, and 10 dB. The simulated test set, provided by the CHiME organization for the challenge, contains the noisy speech signals which were generated by artificially mixing the clean speech and noises. Specifically, the clean speech signals were filtered by the impulse responses (IR) between the speaker and microphone, estimated from the real recorded signals and hence, the simulated data exhibit a more realistic nature of the noisy speech (see [43] for more details about the database).

For both the additive and simulated data types, we considered the Bus ($i = 1$), Pedestrian ($i = 2$) and Street ($i = 3$) noises for the matched noise case and used the Cafe noise from the CHiME database for the mismatched noise case. Regaring the additive noise, we additionally selected the Factory 1 and Babble noises from the NOISEX database [44] for the mismatched noise case. The sampling rate of all signals was set to 16 kHz. The noise types used for the test are summarized in Table 2.

Regarding the implementation, a Hanning window of 512 samples with 50% overlap was employed for the STFT analysis. We used $M_i = 60$ (for all i) and $M_{SF} = M_{NF} = 20$ basis vectors. The values of $(\tau_S, \tau_N) = (0.4, 0.9)$ were chosen as the temporal smoothing factors in (3) and (4). We used $L_b = 16$ for the mini-batch size. For the pre-processor, the value of 0.9 was used as the smoothing factor in the decision-directed (DD) method for the *a priori* SNR estimation in [2], whereas 0.85 was used as the smoothing factor for the noise PSD estimation in [40]. Regarding the shape parameters a^i, we obtained values around 0.02 using the training data (similar results were found when using different initial values, e.g., $a^i = 0.1$). Although we can use such values during the enhancement stage, we found that instead using larger values resulted in slightly better enhancement performance, where we ultimately chose 0.1 for the speech and 0.2 for the noises in the experiments. The reason for this phenomenon can be explained as follows. The basis vectors in the proposed framework are estimated within a restricted decision boundary for each class, which may prevent them from properly representing the target magnitude spectrum. This becomes severe when the number of sources increases (i.e., resulting in smaller decision regions) and hence, may further limit the enhancement performance. Fortunately, the extra free basis vectors can handle such limitation by supporting the class-conditioned basis vectors to better represent the target observation $\mathbf{V}_{l_b}$. In particular, for a given class i, it is necessary to relax the dependency of the free basis vectors on their prior distribution so that they are able to be estimated beyond the decision boundaries. This can be specified by lowering the degree of sparsity of the activations, which corresponds to using a larger value of a^i [28].

We considered the PESQ [30], SDR [31] and SSNR as the objective measures of performance. The PESQ attempts to predict overall perceptual quality in mean opinion scores (MOS) and the SDR measures the overall quality of the enhanced speech in dB by considering both the aspects of speech distortion and noise reduction. For all the measures, a higher value indicates a better result.

6.2. Benchmark algorithms

To evaluate the enhancement performance of the proposed VNCP-BC method, we implemented several benchmark algorithms, which are summarized below. Basic settings, such as the STFT analysis and synthesis, the mini-batch size L_b and the reconstruction method introduced in Section II, were kept identical when applicable.

1) MMSE-STSA estimator: We implemented the MMSE-STSA estimator [2], where the noise PSD was estimated based on [40]. A smoothing factor of 0.85 in the DD method and 0.9 in the noise PSD estimation were used.

2) NMF: The standard NMF algorithm based on KL-divergence introduced in Section II was evaluated.

3) NMF model with distinct basis vectors: Among several NMF algorithms aiming at estimating the distinct basis vectors, we implemented two algorithms as representative benchmarks. The first one is estimating the basis vectors based on the cross-coherence penalty (NCC) introduced in [14]. The second one is our previous work in [25], where the class-conditioned basis vectors are obtained via the MAP estimator.

4) NMF with basis compensation (NBC): The NMF algorithm with basis compensation proposed in [26] was evaluated, as a representative benchmark among several NMF algorithms proposed for handling the mismatch problem. We examined with three different types of basis vectors, i.e., obtained via the conventional NMF, NCC and NCP methods. We used identical settings for the pre- and post-processor as in the proposed VNCP-BC method.

5) Bayesian NMF model (BNMF): To compare with a VBEM-based NMF algorithm, We implemented the BNMF method in [28]. The difference with the proposed VNCP (-BC) method is that the BNMF method estimates the basis matrix for each source independently as in the typical supervised NMF-based framework, whereas the proposed method estimates the basis matrices for all sources jointly.

In addition to the above mentioned benchmarks, we implemented the proposed method without employing the free basis vectors and pre-processing, which will be referred to as VNCP.

We used $M_i = 80$ basis vectors for all NMF-based benchmark algorithms (including the VNCP method) except for the NBC method, where we used $M_i = 60$ and $M_{SF} = M_{NF} = 20$. Hence, the same total number of basis vectors was employed for fair comparison. To perform the noise classification for the benchmark algorithms, we estimated the set $\boldsymbol{\theta}_C$ based on the Gaussian-distributed class-conditional density [25, 27]. For the NMF, NBC and BNMF methods, we first estimated the basis vectors for each class i, then we applied the ML criterion to the basis vectors [25]. The set $\boldsymbol{\theta}_C$ for the NCP method was jointly obtained with the NMF parameters. The noise classification was performed by following a strategy similar to the one

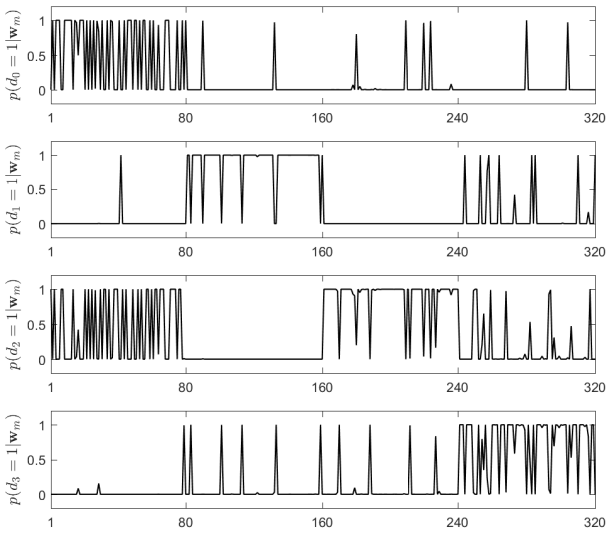

Figure 2: The posterior class probabilities $p(d_i = 1|\mathbf{w}_m)$.

introduced in Section V-A. Note that the pre-processing was performed only for the noise classification in the NMF, DCP and VNCP methods, since these methods do not employ the pre-processed noisy speech during the reconstruction.

6.3. Results

Figure 2 shows the PCPs of the estimated basis vectors $\mathbb{E}_q[\mathbf{w}_m^i]$ (which will be simply denoted by $\mathbf{w}_m^i$). The x-axis indicates the m-th column vector of the matrix $[\mathbf{W}^0, ..., \mathbf{W}^3] = [\mathbf{w}_m]$, where each submatrix $\mathbf{W}^i$ consists of 80 basis vectors, i.e., $M_i = 80$ for all i. For each class i, the PCP values $p(d_i = 1|\mathbf{w}_m)$ should be close to one for the interval $iM_i+1 \leq m \leq (i+1)M_i$, whereas the PCPs for the other intervals should be close to zero. Regarding the class $i = 0$, for example, the PCPs $p(d_0 = 1|\mathbf{w}_m)$ for the interval $1 \leq m \leq 80$ should be close to one, whereas the PCPs for the interval $81 \leq m \leq 320$ should be close to zero. We can see that the basis vectors are estimated to be distinct in terms of the PCP, which implies that the basis vectors of each source will be less likely to represent each other. Similar patterns were found when using $M_i = 60$.

Figure 3 shows an example of the noise classification results using the method introduced in Section V-A. In this particular example, a male speech signal was degraded with a noise at 0 dB input SNR. Specifically, the noise was generated by concatenating the Bus ($i = 1$), Street ($i = 3$) and Pedestrian ($i = 2$) noises where each noise signal was 3 seconds in duration. As we can see, the noise type is well estimated. The magnitude spectra of the clean speech, noisy speech and the enhanced speech using the proposed VNCP-BC method, for this particular example, are illustrated in Figure 4. As it can be observed, the background noise has been significantly reduced.

The average results over all utterances for the additive noises are shown in Tables 3 to 8, where the values in bold indicate the best performance along the row. Most of all, we can see that the proposed VNCP-BC method provided better enhancement performance than the benchmark algorithms in general

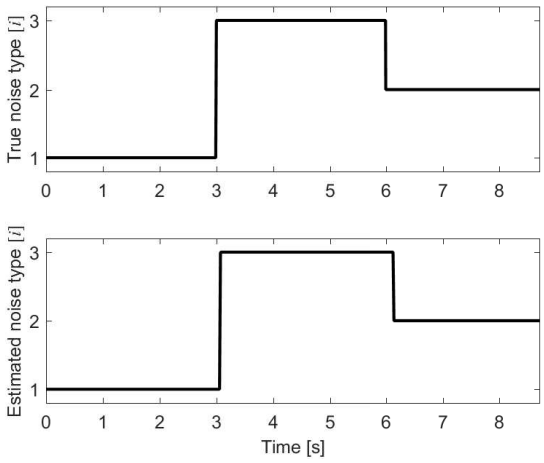

Figure 3: An example of noise classification. Top shows the true noise type and bottom shows the estimated noise type using the proposed method.

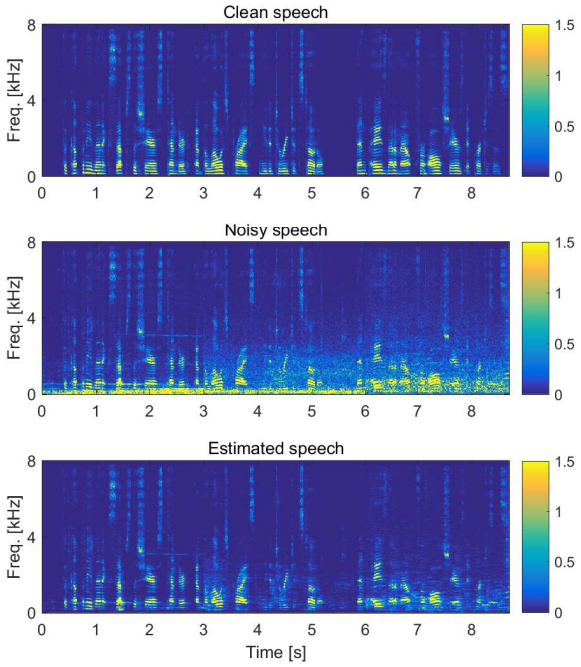

Figure 4: Examples of magnitude spectrograms of the clean, noisy and estimated clean speech using the VNCP-BC method. A male speech is degraded by a noise consisting of different types as shown in Figure 3, at 0 dB input SNR.

for both the matched and mismatched noise cases. Specifically, the proposed VNCP-BC method resulted in better performance compared to using the algorithms introduced in our previous works, i.e., the NCP and NBC methods. Moreover, the VNCP-BC method provided better results than the VNCP method, which further validates that implementing the basis compensation scheme improves the enhancement performance.

Regarding the matched noises, the results of the VBEM-based VNCP method were found to be better than the MAP-based NCP method. Comparing between the VBEM-based methods, the class-conditioned model-based VNCP method exhibited better performance than the independent source training-based BNMF method in general, whereas the BNMF

Table 3: Average results for additive Bus noise (matched)

Input SNR	Eval.	Noisy	STSA	NMF	NCC	NCP	NBC -NMF	NBC -NCC	NBC -NCP	BNMF	VNCP	VNCP -BC
-5 dB	PESQ	1.83	2.08	2.07	2.08	2.08	2.17	2.17	2.16	2.11	2.11	**2.27**
	SDR	-4.89	0.17	2.83	2.63	2.80	6.60	6.50	6.80	3.43	3.67	**7.48**
	SSNR	-13.57	-7.10	-4.50	-4.58	-4.49	-1.24	-1.22	-1.00	-3.54	-3.31	**-0.35**
0 dB	PESQ	2.20	2.43	2.41	2.42	2.42	2.49	2.49	2.48	2.42	2.42	**2.57**
	SDR	0.05	5.25	7.70	7.51	7.67	10.39	10.33	10.51	8.13	8.34	**11.11**
	SSNR	-8.56	-2.75	-0.78	-0.85	-0.85	1.57	1.58	1.73	0.14	0.33	**2.29**
5 dB	PESQ	2.55	2.76	2.74	2.74	2.74	2.78	2.78	2.77	2.74	2.75	**2.87**
	SDR	5.03	9.99	11.96	11.79	11.98	13.62	13.56	13.55	12.61	12.75	**14.27**
	SSNR	-3.56	1.43	2.38	2.31	2.21	4.12	4.13	4.20	3.70	3.82	**4.76**
10 dB	PESQ	2.90	3.07	3.04	3.04	3.05	3.04	3.04	3.03	3.07	3.07	**3.15**
	SDR	10.03	14.33	15.05	15.04	15.19	16.22	16.12	15.96	16.56	16.64	**17.06**
	SSNR	1.45	5.54	4.87	4.84	4.65	6.36	6.39	6.34	6.85	6.90	**7.07**

Table 4: Average results for additive Pedestrian noise (matched)

Input SNR	Eval.	Noisy	STSA	NMF	NCC	NCP	NBC -NMF	NBC -NCC	NBC -NCP	BNMF	VNCP	VNCP -BC
-5 dB	PESQ	1.22	1.34	1.33	1.35	1.33	1.30	1.33	1.32	1.36	1.36	**1.39**
	SDR	-4.88	-3.61	-4.19	-3.96	-3.93	-3.55	-3.44	-3.58	-3.70	-3.73	**-3.17**
	SSNR	-13.88	-9.02	-9.22	-9.21	-9.25	-6.51	-6.53	-6.50	-9.07	-9.13	**-6.11**
0 dB	PESQ	1.51	1.70	1.70	1.71	1.70	1.73	1.75	1.74	1.77	1.77	**1.85**
	SDR	0.06	1.95	1.11	1.31	1.33	2.09	2.16	2.05	2.01	1.96	**2.75**
	SSNR	-8.87	-4.70	-4.78	-4.80	-4.82	-2.67	-2.65	-2.66	-3.99	-4.06	**-1.79**
5 dB	PESQ	1.85	2.09	2.09	2.10	2.09	2.13	2.15	2.14	2.18	2.18	**2.26**
	SDR	5.04	7.07	6.02	6.18	6.26	6.86	6.94	6.85	7.45	7.38	**7.85**
	SSNR	-3.87	-0.43	-0.80	-0.86	-0.86	0.92	0.95	0.90	1.08	0.98	**1.92**
10 dB	PESQ	2.20	2.44	2.46	2.48	2.46	2.45	2.47	2.45	2.55	2.55	**2.61**
	SDR	10.03	11.88	10.00	10.16	10.28	10.69	10.76	10.59	**12.36**	12.23	12.30
	SSNR	1.14	3.87	2.47	2.38	2.39	4.02	4.05	3.91	5.16	5.03	**5.21**

Table 5: Average results for additive Street noise (matched)

Input SNR	Eval.	Noisy	STSA	NMF	NCC	NCP	NBC -NMF	NBC -NCC	NBC -NCP	BNMF	VNCP	VNCP -BC
-5 dB	PESQ	1.39	1.68	1.63	1.64	1.64	1.84	1.86	1.86	1.77	1.80	**2.04**
	SDR	-4.89	-0.35	0.76	1.06	0.89	4.07	3.80	4.72	3.99	4.49	**7.04**
	SSNR	-13.72	-6.80	-6.11	-6.05	-6.16	-2.73	-2.89	-2.40	-3.16	-2.71	**-0.34**
0 dB	PESQ	1.67	2.02	1.98	1.99	1.98	2.20	2.21	2.21	2.11	2.15	**2.39**
	SDR	0.05	4.87	5.77	6.06	5.91	8.37	8.18	8.83	8.34	8.70	**10.28**
	SSNR	-8.72	-2.61	-1.97	-1.89	-2.02	0.42	0.36	0.65	0.53	0.80	**2.06**
5 dB	PESQ	2.00	2.37	2.35	2.36	2.36	2.52	2.53	2.53	2.46	2.48	**2.66**
	SDR	5.03	9.63	10.17	10.43	10.37	11.83	11.81	12.13	12.36	12.58	**13.25**
	SSNR	-3.72	1.43	1.58	1.69	1.54	3.30	3.38	3.47	3.86	3.93	**4.31**
10 dB	PESQ	2.36	2.70	2.71	2.72	2.72	2.77	2.79	2.78	2.77	2.78	**2.91**
	SDR	10.03	14.03	13.49	13.76	13.80	14.46	14.59	14.59	16.03	16.13	**16.16**
	SSNR	1.29	5.41	4.39	4.57	4.38	5.67	5.94	5.80	6.67	6.62	**6.69**

method provided slightly better results for the Pedestrian noise. Among the NBC methods with different types of basis vectors, we can see that using the basis vectors obtained via the NCP method provided better results. We also conducted experiments for all benchmarks and proposed algorithms assuming that the noise type is known *a priori*, for the matched noise case. Although we did not report their objective results in this paper, we have seen that there were no significant differences with the results obtained by including the noise classification. That is, the results increased by about 0.01 in PESQ and SDR for all methods when assuming that the noise type is known *a priori*.

The effectiveness of using the basis compensation scheme can be better verified from the results of the mismatched noises. In general, we can see that some NMF-based benchmark algorithms showed even worse performance than using the STSA estimator, whereas the NBC-based methods provided reasonable results. Specifically, although the NBC methods gave acceptable SDR and SSNR values for the Cafe and Babble noises under low input SNRs, the proposed VNCP-BC method exhibited better than all benchmark algorithms in most cases.

The average results over all utterances for the simulated data set are shown in Table 9. Although the results showed slightly different pattern from the additive noise case (e.g., the STSA estimator gave even better results than some of the benchmarks for the Pedestrian noise), mainly due to the effect of the IR-filtered clean speech, we can see that the proposed VNCP-BC method provided the best results for all types of noises. Hence, it is verified that the proposed VNCP-BC method performs well under a more realistic environment.

7. Conclusion and future works

We introduced a training and compensation algorithm of the class-conditioned basis vectors in the NMF model for single-channel speech enhancement. We considered the PGM for both the NMF and classification models. The former is specified by a Poisson observation model, whereas the latter is specified by Gamma class conditional densities, which are used as *a priori*

Table 6: Average results for additive Cafe noise (mismatched)

Input SNR	Eval.	Noisy	STSA	NMF	NCC	NCP	NBC -NMF	NBC -NCC	NBC -NCP	BNMF	VNCP	VNCP -BC
-5 dB	PESQ	1.30	1.38	1.38	1.39	1.38	1.29	1.32	1.32	1.38	1.37	**1.39**
	SDR	-4.89	-3.40	-2.98	-2.99	-2.78	-2.22	-2.23	**-1.86**	-3.23	-3.14	-1.93
	SSNR	-14.48	-10.93	-10.08	-10.19	-9.75	**-7.99**	-8.37	-8.09	-10.68	-10.53	-8.13
0 dB	PESQ	1.56	1.68	1.69	1.70	1.71	1.67	1.69	1.68	1.71	1.72	**1.76**
	SDR	0.06	2.07	2.13	2.17	2.37	3.32	3.33	3.61	2.45	2.53	**4.01**
	SSNR	-9.47	-6.26	-5.56	-5.65	-5.27	-3.74	-4.06	-3.90	-5.54	-5.44	**-3.34**
5 dB	PESQ	1.87	2.00	2.03	2.04	2.06	2.04	2.05	2.04	2.11	2.12	**2.15**
	SDR	5.04	7.18	6.88	6.93	7.13	7.99	8.07	8.25	8.14	8.17	**9.13**
	SSNR	-4.47	-1.72	-1.46	-1.54	-1.25	0.06	-0.11	-0.09	-0.13	-0.10	**0.99**
10 dB	PESQ	2.20	2.35	2.37	2.39	2.41	2.39	2.40	2.38	2.52	**2.54**	**2.54**
	SDR	10.03	11.96	10.72	10.81	10.94	11.63	11.85	11.80	13.20	13.19	**13.31**
	SSNR	0.54	2.74	1.96	1.91	2.08	3.35	3.34	3.21	4.69	4.66	**4.71**

Table 7: Average results for additive Factory 1 noise (mismatched)

Input SNR	Eval.	Noisy	STSA	NMF	NCC	NCP	NBC -NMF	NBC -NCC	NBC -NCP	BNMF	VNCP	VNCP -BC
-5 dB	PESQ	1.23	1.44	1.33	1.34	1.33	1.47	1.49	1.48	1.40	1.42	**1.56**
	SDR	-4.90	-1.44	-3.33	-2.59	-3.17	0.56	0.51	0.83	-1.07	-0.88	**2.01**
	SSNR	-14.33	-8.33	-9.82	-9.22	-9.76	-5.36	-5.49	-5.13	-8.40	-8.48	**-4.25**
0 dB	PESQ	1.50	1.77	1.67	1.68	1.67	1.84	1.86	1.85	1.74	1.75	**1.96**
	SDR	0.05	3.96	1.92	2.64	2.12	5.53	5.51	5.73	4.33	4.43	**7.08**
	SSNR	-9.32	-4.05	-5.31	-4.75	-5.25	-1.81	-1.87	-1.64	-3.77	-3.87	**-0.63**
5 dB	PESQ	1.83	2.12	2.05	2.06	2.04	2.20	2.22	2.21	2.13	2.14	**2.34**
	SDR	5.03	8.83	6.68	7.34	6.90	9.52	9.57	9.59	9.58	9.65	**11.11**
	SSNR	-4.32	0.14	-1.32	-0.83	-1.28	1.50	1.53	1.58	0.92	0.77	**2.62**
10 dB	PESQ	2.18	2.48	2.43	2.44	2.42	2.51	2.53	2.51	2.52	2.52	**2.67**
	SDR	10.03	13.40	10.42	10.94	10.72	12.38	12.56	12.35	14.11	14.24	**14.51**
	SSNR	0.68	4.40	1.88	2.27	1.90	4.28	4.41	4.27	4.97	4.77	**5.46**

Table 8: Average results for additive Babble noise (mismatched)

Input SNR	Eval.	Noisy	STSA	NMF	NCC	NCP	NBC -NMF	NBC -NCC	NBC -NCP	BNMF	VNCP	VNCP -BC
-5 dB	PESQ	1.33	1.45	1.44	1.44	1.44	1.40	1.43	1.43	1.46	1.46	**1.53**
	SDR	-4.89	-2.72	-3.75	-3.67	-3.68	-1.94	-1.91	-1.71	-3.95	-3.97	**-1.66**
	SSNR	-14.26	-9.90	-10.85	-10.90	-10.82	**-7.53**	-7.85	-7.70	-11.88	-11.91	-8.15
0 dB	PESQ	1.63	1.79	1.78	1.78	1.78	1.77	1.78	1.79	1.82	1.82	**1.90**
	SDR	0.05	2.77	1.48	1.55	1.54	3.54	3.50	3.67	1.60	1.57	**4.38**
	SSNR	-9.25	-5.39	-6.17	-6.17	-6.12	-3.49	-3.69	-3.61	-6.55	-6.61	**-3.24**
5 dB	PESQ	1.96	2.12	2.14	2.15	2.14	2.12	2.13	2.14	2.21	2.21	**2.29**
	SDR	5.03	7.78	6.47	6.58	6.51	7.97	8.01	8.11	7.42	7.35	**9.51**
	SSNR	-4.24	-1.02	-1.84	-1.78	-1.81	0.18	0.12	0.10	-0.86	-0.97	**1.34**
10 dB	PESQ	2.31	2.46	2.50	2.50	2.50	2.45	2.46	2.46	2.61	2.60	**2.64**
	SDR	10.03	12.41	10.68	10.92	10.69	11.27	11.45	11.43	12.91	12.80	**13.37**
	SSNR	0.77	3.31	1.82	1.94	1.80	3.32	3.38	3.24	4.47	4.29	**4.98**

Table 9: Average results for simulated noisy speech

Input SNR	Eval.	Noisy	STSA	NMF	NCC	NCP	NBC -NMF	NBC -NCC	NBC -NCP	BNMF	VNCP	VNCP -BC
BUS (mat.)	PESQ	1.70	1.97	1.94	1.95	1.94	2.05	2.06	2.05	1.99	2.00	**2.16**
	SDR	-1.34	2.79	3.62	4.18	4.00	6.45	6.39	6.66	5.59	5.63	**7.92**
	SSNR	-10.75	-7.35	-6.61	-6.36	-6.44	-4.75	-4.88	-4.67	-5.52	-5.49	**-3.65**
PED. (mat.)	PESQ	1.50	1.72	1.67	1.67	1.67	1.76	1.78	1.76	1.73	1.73	**1.86**
	SDR	0.13	3.26	1.47	1.58	0.89	4.33	4.37	4.41	2.39	2.43	**5.35**
	SSNR	-10.58	-7.54	-7.32	-7.30	-7.36	-5.48	-5.60	-5.53	-6.95	-6.90	**-4.64**
STR. (mat.)	PESQ	1.51	1.76	1.73	1.74	1.74	1.85	1.86	1.85	1.82	1.83	**2.00**
	SDR	-1.76	2.08	1.77	2.10	1.98	4.69	4.69	4.96	3.77	3.80	**6.39**
	SSNR	-10.81	-7.40	-6.96	-6.89	-6.92	-5.10	-5.30	-5.02	-5.74	-5.79	**-3.74**
CAF. (mis.)	PESQ	1.52	1.71	1.68	1.69	1.67	1.72	1.74	1.73	1.74	1.73	**1.82**
	SDR	-0.18	2.54	1.02	0.80	0.74	3.41	3.52	3.56	2.36	2.12	**4.68**
	SSNR	-10.64	-7.80	-7.48	-7.57	-7.53	-5.84	-5.99	-5.91	-7.11	-7.19	**-4.95**

distribution for the basis vectors. During the training stage, the basis matrices for the clean speech and noises were estimated jointly by constraining them to belong to different classes. The parameters of the NMF model and PGM of classification were obtained by using the VBEM algorithm, which guarantees convergence to a stationary point. During the enhancement stage, we performed a noise classification followed by a basis compensation. The latter was implemented by using extra free basis vectors to capture features which are not included in the training data. The PGM parameters for classification were employed while estimating the free basis vectors as well as during the noise classification. Experiments showed that the proposed

VNCP-BC method provided better enhancement performance than the benchmark algorithms for all types of noises and input SNRs.

Finally, we comment on some interesting research avenues for further improving the enhancement performance of our proposed method. Firstly, we can consider modeling the basis vectors using a more accurate multimodal distribution, e.g., the Gamma mixture model [22]. This extended prior modeling may also offer the potential of a noise-independent application by handling highly correlated noise sources (i.e., one universal basis matrix covering all noise types). Secondly, we can take into account the convolutive nature of the acoustic medium (e.g., room impulse response) between the sound source and the microphone, in order to deal with more realistic reverberant environments. A possible approach to this end is to model the latent variables in the NMF model via auto-regressive moving average (ARMA) processes [36].

Appendix A. Variational lower bound

Based on (5), (6), (10) and (12), the logarithm of the full joint distribution is given by

$$
\begin{aligned}
&\ln p(\mathbf{V},\mathbf{C},\mathbf{W},\mathbf{H};\boldsymbol{\theta}_R) \qquad \text{(A.1)}\\
&= \ln p(\mathbf{V}|\mathbf{C}) + \ln p(\mathbf{C}|\mathbf{W},\mathbf{H}) + \ln p(\mathbf{W};\boldsymbol{\theta}_C) + \ln p(\mathbf{H};\mathbf{a},\mathbf{b})\\
&= \sum_{i=0}^{I-1}\sum_{k=1}^{K}\sum_{l=1}^{L_i} \ln \delta\Big(v_{kl}^i - \sum_{m=1}^{M_i} c_{kl}^{m,i}\Big)\\
&+ \sum_{i=0}^{I-1}\sum_{k=1}^{K}\sum_{m=1}^{M_i}\sum_{l=1}^{L_i} \Big(c_{kl}^{m,i}\ln(w_{km}^i h_{ml}^i) - w_{km}^i h_{ml}^i - \ln(c_{kl}^{m,i}!)\Big)\\
&+ \sum_{i=0}^{I-1}\sum_{m=1}^{M_i}\sum_{k=1}^{K}\Big((\alpha_k^i - 1)\ln w_{km}^i - \frac{w_{km}^i}{\beta_k} - \ln\Gamma(\alpha_k^i) - \alpha_k^i \ln\beta_k\Big)\\
&+ \sum_{i=0}^{I-1}\sum_{m=1}^{M_i}\sum_{l=1}^{L_i}\Big((a^i - 1)\ln h_{ml}^i - \frac{a^i}{b^i} h_{ml}^i - \ln\Gamma(a^i) - a^i \ln\Big(\frac{b^i}{a^i}\Big)\Big).
\end{aligned}
$$

The energy $\mathcal{L}_V(q(\boldsymbol{\theta}_L);\boldsymbol{\theta}_R)$ in (13) is simply found by evaluating the expectations of (A.1) with respect to $q(\mathbf{C},\mathbf{W},\mathbf{H})$ in (15)-(17), where the sufficient statistics are given by (23)-(25).

Based on (18), (20) and (22), and using the sufficient statistics in (23)-(25), the entropy $\mathcal{L}_E(q(\boldsymbol{\theta}_L)) = -\mathbb{E}_q[\ln q(\boldsymbol{\theta}_L)]$ can be written as

$$
\begin{aligned}
&\mathcal{L}_E(q(\boldsymbol{\theta}_L)) \qquad \text{(A.2)}\\
&= \sum_{i=0}^{I-1}\sum_{k=1}^{K}\sum_{l=1}^{L_i}\Big(-\ln(v_{kl}^i!) - \sum_{m=1}^{M_i} v_{kl}^i \bar{p}_{kl}^{m,i} \ln \bar{p}_{kl}^{m,i}\\
&\qquad -\mathbb{E}_q\Big[\ln\delta\Big(v_{kl}^i - \sum_{m=1}^{M_i} c_{kl}^{m,i}\Big)\Big] + \sum_{m=1}^{M_i}\mathbb{E}_q[\ln(c_{kl}^{m,i}!)]\Big)\\
&- \sum_{i=0}^{I-1}\sum_{k=1}^{K}\sum_{m=1}^{M_i}\Big((\bar{\alpha}_{km}^i - 1)\Psi(\bar{\alpha}_{km}^i) - \ln\bar{\beta}_{km}^i - \bar{\alpha}_{km}^i - \ln\Gamma(\bar{\alpha}_{km}^i)\Big)\\
&- \sum_{i=0}^{I-1}\sum_{m=1}^{M_i}\sum_{l=1}^{L_i}\Big((\bar{a}_{ml}^i - 1)\Psi(\bar{a}_{ml}^i) - \ln\bar{b}_{ml}^i - \bar{a}_{ml}^i - \ln\Gamma(\bar{a}_{ml}^i)\Big).
\end{aligned}
$$

The lower bound on the marginal LLF, $\ln p(\mathbf{V};\boldsymbol{\theta}_R)$, is obtained by summing the energy and entropy terms as given by (13). Note that the terms in $\mathbb{E}_q[\cdot]$ in the third line in (A.2), which are analytically intractable, are canceled by their corresponding terms in the energy $\mathcal{L}_V(q(\boldsymbol{\theta}_L);\boldsymbol{\theta}_R)$ [28].

References

[1] N. Virag, Single channel speech enhancement based on masking properties of the human auditory system, IEEE Transactions on speech and audio processing 7 (2) (1999) 126–137.

[2] Y. Ephraim, D. Malah, Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator, IEEE Transactions on Acoustics, Speech, and Signal Processing 32 (6) (1984) 1109–1121.

[3] E. Plourde, B. Champagne, Auditory-based spectral amplitude estimators for speech enhancement, IEEE transactions on audio, speech, and language processing 16 (8) (2008) 1614–1623.

[4] F. Jabloun, B. Champagne, Incorporating the human hearing properties in the signal subspace approach for speech enhancement, IEEE Transactions on Speech and Audio Processing 11 (6) (2003) 700–708.

[5] N. Bertin, R. Badeau, E. Vincent, Enforcing harmonicity and smoothness in bayesian non-negative matrix factorization applied to polyphonic music transcription, IEEE Transactions on Audio, Speech, and Language Processing 18 (3) (2010) 538–549.

[6] T. Virtanen, Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria, IEEE transactions on audio, speech, and language processing 15 (3) (2007) 1066–1074.

[7] N. Mohammadiha, P. Smaragdis, A. Leijon, Supervised and unsupervised speech enhancement using nonnegative matrix factorization, IEEE Transactions on Audio, Speech, and Language Processing 21 (10) (2013) 2140–2151.

[8] S. Zafeiriou, A. Tefas, I. Buciu, I. Pitas, Exploiting discriminant information in nonnegative matrix factorization with application to frontal face verification, IEEE Transactions on Neural Networks 17 (3) (2006) 683–695.

[9] D. D. Lee, H. S. Seung, Algorithms for non-negative matrix factorization, in: Advances in neural information processing systems, 2001, pp. 556–562.

[10] C. Févotte, N. Bertin, J.-L. Durrieu, Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis, Neural computation 21 (3) (2009) 793–830.

[11] Y. Xu, J. Du, L.-R. Dai, C.-H. Lee, A regression approach to speech enhancement based on deep neural networks, IEEE/ACM Transactions on Audio, Speech and Language Processing 23 (1) (2015) 7–19.

[12] K. Han, Y. Wang, D. Wang, W. S. Woods, I. Merks, T. Zhang, Learning spectral mapping for speech dereverberation and denoising, IEEE/ACM Transactions on Audio, Speech, and Language Processing 23 (6) (2015) 982–992.

[13] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, P. Smaragdis, Joint optimization of masks and deep recurrent neural networks for monaural source separation, IEEE/ACM Transactions on Audio, Speech and Language Processing 23 (12) (2015) 2136–2147.

[14] E. M. Grais, H. Erdogan, Discriminative nonnegative dictionary learning using cross-coherence penalties for single channel source separation., in: Interspeech, 2013, pp. 808–812.

[15] K. Kwon, J. W. Shin, N. S. Kim, Target source separation based on discriminative nonnegative matrix factorization incorporating cross-reconstruction error, IEICE Transactions on Information and Systems E98-D (11) (2015) 2017–2020.

[16] Z. Wang, F. Sha, Discriminative non-negative matrix factorization for single-channel speech separation, in: Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on, IEEE, 2014, pp. 3749–3753.

[17] F. Weninger, J. Le Roux, J. R. Hershey, S. Watanabe, Discriminative nmf and its application to single-channel source separation., in: Interspeech, 2014, pp. 865–869.

[18] P. Sprechmann, A. M. Bronstein, G. Sapiro, Supervised non-euclidean sparse nmf via bilevel optimization with applications to speech enhance-

ment, in: Hands-free Speech Communication and Microphone Arrays (HSCMA), 2014 4th Joint Workshop on, IEEE, 2014, pp. 11–15.
[19] R. Badeau, N. Bertin, E. Vincent, Stability analysis of multiplicative update algorithms and application to nonnegative matrix factorization, IEEE Transactions on Neural Networks 21 (12) (2010) 1869–1881.
[20] G. J. Mysore, P. Smaragdis, A non-negative approach to semi-supervised separation of speech from noise with the use of temporal dynamics, in: Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on, IEEE, 2011, pp. 17–20.
[21] H. Chung, E. Plourde, B. Champagne, Regularized non-negative matrix factorization with gaussian mixtures and masking model for speech enhancement, Speech Communication 87 (2017) 18–30.
[22] T. Virtanen, A. T. Cemgil, Mixtures of gamma priors for non-negative matrix factorization based speech separation, in: Independent Component Analysis and Signal Separation, Springer, 2009, pp. 646–653.
[23] E. M. Grais, H. Erdoğan, Adaptation of speaker-specific bases in non-negative matrix factorization for single channel speech-music separation, in: Interspeech, 2011, pp. 569–572.
[24] K. Kwon, J. W. Shin, N. S. Kim, NMF-based speech enhancement using bases update, IEEE Signal Processing Letters 22 (4) (2015) 450–454.
[25] H. Chung, E. Plourde, B. Champagne, Discriminative training of nmf model based on class probabilities for speech enhancement, IEEE Signal Processing Letters 23 (4) (2016) 502–506.
[26] H. Chung, E. Plourde, B. Champagne, Basis compensation in non-negative matrix factorization model for speech enhancement, in: Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on, IEEE, 2016, pp. 2249–2253.
[27] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
[28] A. T. Cemgil, Bayesian inference for nonnegative matrix factorisation models, Computational Intelligence and Neuroscience (4) (2009) 1–17.
[29] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, L. K. Saul, An introduction to variational methods for graphical models, Machine learning 37 (2) (1999) 183–233.
[30] I. Recommendation, Perceptual evaluation of speech quality (PESQ), an objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs , ITU-T Recommendation (2001) 862.
[31] E. Vincent, R. Gribonval, C. Févotte, Performance measurement in blind audio source separation, IEEE Trans. Audio, Speech, and Language Process. 14 (4) (2006) 1462–1469.
[32] J.-T. Chien, P.-K. Yang, Bayesian factorization and learning for monaural source separation, IEEE/ACM Transactions on Audio, Speech, and Language Processing 24 (1) (2016) 185–195.
[33] M. D. Hoffman, Poisson-uniform nonnegative matrix factorization, in: Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on, IEEE, 2012, pp. 5361–5364.
[34] I. Ulusoy, C. M. Bishop, Generative versus discriminative methods for object recognition, in: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, Vol. 2, IEEE, 2005, pp. 258–265.
[35] L. K. Saul, D. D. Lee, Multiplicative updates for classification by mixture models, in: Advances in neural information processing systems, Vol. 14, 2001, pp. 897–904.
[36] R. Badeau, M. D. Plumbley, Multichannel high-resolution nmf for modeling convolutive mixtures of non-stationary signals in the time-frequency domain, IEEE/ACM Transactions on Audio, Speech and Language Processing 22 (11) (2014) 1670–1680.
[37] J. Eggert, E. Korner, Sparse coding and nmf, in: Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on, Vol. 4, IEEE, 2004, pp. 2529–2533.
[38] S. M. Kim, J. H. Park, K. H. Kook, S. J. Lee, Y. K. Lee, Non-negative matrix factorization based noise reduction for noise robust automatic speech recognition, in: Lecture Notes in Computer Science, Vol. 7191, Springer, 2012, pp. 338–346.
[39] M. Sun, Y. Li, J. F. Gemmeke, X. Zhang, Speech enhancement under low snr conditions via noise estimation using sparse and low-rank nmf with kullback–leibler divergence, IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP) 23 (7) (2015) 1233–1242.
[40] T. Gerkmann, R. C. Hendriks, Unbiased mmse-based noise power estimation with low complexity and low tracking delay, IEEE Transactions on Audio, Speech, and Language Processing 20 (4) (2012) 1383–1393.
[41] Z. Duan, G. J. Mysore, P. Smaragdis, Speech enhancement by online non-negative spectrogram decomposition in nonstationary noise environments, in: Interspeech, 2012, pp. 595–598.
[42] D. O'Shaughnessy, Speech Communication: Human and Machine, IEEE press, 1987.
[43] E. Vincent, S. Watanabe, A. A. Nugraha, J. Barker, R. Marxer, An analysis of environment, microphone and data simulation mismatches in robust speech recognition, Computer Speech & Language - to appear.
[44] A. Varga, H. J. Steeneken, Assessment for automatic speech recognition. II. NOISEX-92: a database and an experiment to study the effect of additive noise on speech recognition systems, Speech Communication 12 (3) (1993) 247–251.