LEARNABLE LOW RANK SPARSE MODELS FOR SPEECH DENOISING

Pablo Sprechmann,[1] *Alex Bronstein*,[2] *Michael Bronstein*[3] *and Guillermo Sapiro*[1]

[1] Duke University,[2] Tel Aviv University,[3] Università della Svizzera Italiana

ABSTRACT

In this paper we present a framework for real time enhancement of speech signals. Our method leverages a new process-centric approach for sparse and parsimonious models, where the representation pursuit is obtained applying a deterministic function or process rather than solving an optimization problem. We first propose a rank-regularized robust version of non-negative matrix factorization (NMF) for modeling time-frequency representations of speech signals in which the spectral frames are decomposed as sparse linear combinations of atoms of a low-rank dictionary. Then, a parametric family of pursuit processes is derived from the iteration of the proximal descent method for solving this model. We present several experiments showing successful results and the potential of the proposed framework. Incorporating discriminative learning makes the proposed method significantly outperform exact NMF algorithms, with fixed latency and at a fraction of it's computational complexity.

Index Terms— Audio denoising, source separation, parsimonious models, neural networks.

1. INTRODUCTION

The problem of isolating or enhancing a speech signal recorded in a noisy environment has been widely studied in the audio processing community [1, 2]. It becomes particularly challenging when the background noise is non-stationary, which is a very common situation in many applications encountered in telephony. We approach this problem as a monaural source separation method by modeling the speech as one source, and the noise as the other. This is a natural approach when the characteristics of both the source of interest and the noise vary throughout time [3, 4, 5, 6].

The decomposition of time-frequency representations, such as the power or magnitude spectrogram in terms of elementary atoms of a dictionary, has become a popular tool in audio processing. In particular, non-negative matrix factorization (NMF) [7], and its probabilistic counterpart probabilistic latent component analysis (PLCA) [8], were shown effective for various speech processing tasks as speech separation [9, 10], denoising [4, 6, 11], robust automatic speech recognition [12, 13], bandwidth extension [14, 15] and speaker recognition [16, 17].

NMF and PLCA produce high quality separation results when the dictionaries for different sources are sufficiently distinct. There is naturally a compromise between the approximation of the training data and tightness of the model: the more general is the dictionary the higher is the chance it will include elements that match spectral patterns in the competing sources. In order to mitigate this problem, recent approaches have proposed alternative models in order to constrain the solution in meaningful ways, as adding sparsity constrains to the activations [10, 18]. However, there is much additional structure in speech (and noise). For example, speech signals are monophonic, they will never generate simultaneously two different harmonic sounds with harmonically unrelated pitches. Using standard NMF reconstruction however, this combination would be allowed. Different works have proposed to regularize versions of NMF or PLCA with this motivation, including co-occurrence statistics of the basis functions [3], smoothness of the activation coefficients [19] and learned temporal dynamics [5, 15]. In all these methods the model is expressed as the minimization of a cost with a data fitting term and some structure-promoting penalties.

In contrast to these ideas, in this work we propose a *process-centric* approach. Instead of trying to design a regularized optimization problem capturing all the variability of speech signals, we propose to use a simplified model and bridge the gap between the model and the real signals via learning. Having a deterministic pursuit process in lieu of iterative optimization brings significant additional advantages. The latency and computational complexity is fixed instead of being data-dependent. While capturing the great advantages of the NMF paradigm, it allows the inclusion of the model into higher-level training objective functions without falling into bi-level optimization problems. Our work builds upon recent developments of fast sparse encoders [20, 21] that were successfully extended to solve audio classification [22] and music source separation tasks [23]. For a general presentation of this appoach refer to [24].

In Secion 2 we briefly introduce NMF. In Section 3 we introduce a new regularized version of NMF for representing speech signals that includes a regularization term related to the nuclear norm of the reconstructed spectrogram. This new setting promotes solutions with low rank and improves robustness with respect to the size of the dictionaries. Section 4 adapts this model for the speech denoising problem. Next, in Section 5 we take this model one step further to a process-centric approach. The process is chosen from a parametric family of pursuit processes derived from the iteration of the proximal descent method for the proposed NMF model. Detailed experimental evaluation on real data is presented in Section 6.

2. NON-NEGATIVE MATRIX FACTORIZATION

Given a non-negative matrix $\mathbf{V} \in \mathbb{R}^{F\times N}$, NMF aims at finding a factorization $\mathbf{V} \approx \mathbf{WH}$ into non-negative matrices $\mathbf{W} \in \mathbb{R}^{F\times Q}$, $\mathbf{H} \in \mathbb{R}^{Q\times N}$ such that $FQ + QN \ll FN$. In audio processing, $\mathbf{V}$ is a non-negative representation of the time-frequency domain, with F frequency samples and N time frames. $\mathbf{W}$ is interpreted as a dictionary with each column representing an elementary spectral atom while $\mathbf{H}$ codes the activation of each atom in the dictionary throughout the time frames. When Q is small, $\mathbf{WH}$ is a low-rank approximation of $\mathbf{V}$. The factorization is obtained by solving

$$\min_{\mathbf{W}\geq\mathbf{0},\mathbf{H}\geq\mathbf{0}} \sum_{i=1}^{N} d(\mathbf{v}_i|[\mathbf{WH}]_i), \tag{1}$$

Work partially supported by BSF, ONR, NSF, DARPA, NGA, ARO, and NSSEFF.

where d is a scalar cost function. Significant attention has been devoted in the literature to the case in which d belongs to the family of the so-called β-divergences [25], defined as

$$d_\beta(\mathbf{a}|\mathbf{b}) = \sum_{j=1}^{F} \begin{cases} \frac{a_i}{b_i} - \log \frac{a_i}{b_i} - 1 & \text{if } \beta = 0 \\ a_i \log a_i/b_i + (a_i - b_i) & \text{if } \beta = 1 \\ \frac{1}{\beta(\beta-1)}(a_i^\beta + (\beta-1)b_i^\beta - \beta a_i b_i^{\beta-1}) & \text{else.} \end{cases}$$

This family includes the three most widely used cost functions in NMF: the Euclidean distance ($\beta = 2$), the Kullback-Leibler divergence ($\beta = 1$), and the Itakura-Saito divergence ($\beta = 0$). One of the most popular ways of solving (1) is the multiplicative gradient descent approach introduced in [7] that alternates a descent over $\mathbf{W}$ and $\mathbf{H}$,

$$\mathbf{W} \leftarrow \mathbf{W}.\frac{\mathbf{W}^{\mathrm{T}}((\mathbf{WH})^{\cdot(\beta-2)}.\mathbf{V})}{\mathbf{W}^{\mathrm{T}}(\mathbf{WH})^{\cdot(\beta-2)}}, \mathbf{H} \leftarrow \mathbf{H}.\frac{((\mathbf{WH})^{\cdot(\beta-2)}.\mathbf{V})\mathbf{H}^{\mathrm{T}}}{(\mathbf{WH})^{\cdot(\beta-2)}\mathbf{H}^{\mathrm{T}}}$$

where . and division represent element-wise operations.

Online NMF algorithms aim at computing the factorization as the data comes in. When the dictionary is available *a priori*, problem (1) is separable in the columns and then the projection is given by

$$\mathbf{v}_t = \underset{\mathbf{h}\geq\mathbf{0}}{\operatorname{argmin}}\, d_\beta(\mathbf{v}_t|\mathbf{Wh}). \tag{2}$$

In many situations, in order to save computational time, (2) is not solved exactly, as discussed in [26]. [6] set the maximum number of iterations of the multiplicative algorithm to a value that empirically provides a good approximation. The update of the dictionary is performed by minimizing (1) w.r.t. $\mathbf{W}$ [26].

3. LOW-RANK SPARSE MODELS

In this section we present a new model for representing speech signals that builds upon NMF. It is well established that speech signals can be accurately represented by a low rank model. For example, clean speech can be reconstructed with perceptually good quality with as few as 20 dictionary atoms as reported by [6]. However, the size of the dictionary that gives the best reconstruction is application dependent. Having more dictionary atoms improves the approximation of the training data but also allows the dictionary to include elements that match spectral patterns in the competing sources.

A way to regularize (1) in order to obtain a factorization of $\mathbf{WH}$ that penalizes high rank reconstructions was proposed in [23]. Recent advances in convex optimization have shown that rank-regularization can be obtained by minimizing the nuclear norm $\|\mathbf{WH}\|_*$, defined as the sum of the singular values of the matrix, which is the tightest convex surrogate for the rank. Akin the ℓ_1-norm that encourages sparsity of vectors, the nuclear norm promotes low rank of matrices. In [23] it was shown that the sum of the Frobenius norms of the non-negative matrices $\mathbf{W}$ and $\mathbf{H}$ gives an upper bound on the nuclear norm of their product,

$$||\mathbf{WH}||_* \leq \frac{1}{2}\|\mathbf{W}\|_{\mathrm{F}}^2 + \frac{1}{2}\|\mathbf{H}\|_{\mathrm{F}}^2. \tag{3}$$

By adding these two terms as regularizers of (1) one can obtain a solution which has a rank better adapted to the data and less dependent on the exact selection of the number of atoms Q. As an example, we show in Figure 1 a toy supervised speech separation experiment. For each of the two speakers we learn dictionaries using training data and then decompose the test signal in a combined dictionary. The performance of the separation does not degrade with the increase of the rank bound Q. This is sharply different in the standard NMF problem, where the choice of the rank is a delicate issue greatly affecting the performance of the method.

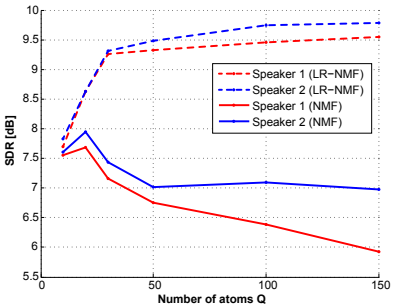

Figure 1. Results for speaker separation performed in the supervised setting described in Section 3.1 and measured in terms of the signal-to-distortion ratio (SDR). Low-rank (LR) NMF and standard NMF varying the number of atoms, Q.

Even if the spectral components of speech can be well represented with low rank models, more robust estimations can be obtained by constraining $\mathbf{H}$ to be sparse. Based on this, we propose to model the time-frequency representations of the speech signals as a sparse linear combination of the atoms of an under-complete dictionary,

$$\min_{\mathbf{W}\geq 0, \mathbf{H}\geq 0} \frac{1}{2} d_\beta(\mathbf{V}|\mathbf{WH}) + \frac{\lambda_*}{2}(\|\mathbf{W}\|_{\mathrm{F}}^2 + \|\mathbf{H}\|_{\mathrm{F}}^2) + \lambda \|\mathbf{H}\|_1, \tag{4}$$

where λ_* and λ are parameters controlling, respectively, the low-rank and the sparsity constraints. A multiplicative gradient descent algorithm can still be used to solve (4).

4. SPEECH DENOISING

We assume that the speech signal is affected by additive non-stationary noise. In line with the literature on NMF-based denoising [4, 6, 11], we propose to model the speech and the noise by a pair of pre-trained dictionaries $\mathbf{W}_{\mathrm{s}}$ and $\mathbf{W}_{\mathrm{n}}$ respectively. Given a degraded signal $\mathbf{V}$, we decompose it into speech and noise signals by finding the activation matrices $\mathbf{H}_{\mathrm{s}}$ and $\mathbf{H}_{\mathrm{n}}$ minimizing

$$\min_{\mathbf{H}_{\mathrm{s}}\geq\mathbf{0}, \mathbf{H}_{\mathrm{n}}\geq\mathbf{0}} \frac{1}{2} d_\beta(\mathbf{V}|\mathbf{W}_{\mathrm{n}}\mathbf{H}_{\mathrm{n}} + \mathbf{W}_{\mathrm{s}}\mathbf{H}_{\mathrm{s}}) + \frac{\lambda_*}{2}\|\mathbf{H}_{\mathrm{s}}\|_{\mathrm{F}}^2 + \frac{\lambda_*}{2}\|\mathbf{H}_{\mathrm{n}}\|_{\mathrm{F}}^2 + \lambda \|\mathbf{H}_{\mathrm{s}}\|_1. \tag{5}$$

Note that the ℓ_2 regularization terms on the dictionaries are superfluous, since they are assumed fixed. Observe that we impose sparsity of the activation corresponding to the speech signal only, as noise is poorly described by sparse activation. The model can be easily adapted to source separation: $\mathbf{H}_n$ would correspond to another speaker and its sparsity would be enforced through an ℓ_1 term.

Problem (5) is column-wise separable and can be solved using a simple adaptation of the multiplicative algorithms described in Section 2. Once $\mathbf{H}_{\mathrm{s}}$ and $\mathbf{H}_{\mathrm{n}}$ are obtained, the time-frequency representations of the speech and the noise are estimated as $\mathbf{W}_{\mathrm{s}}\mathbf{H}_{\mathrm{s}}$ and $\mathbf{W}_{\mathrm{n}}\mathbf{H}_{\mathrm{n}}$, respectively. Next, a time-frequency mask is constructed and the speech is recovered from the mixture by Wiener filtering, as is standard in NMF-based source separation.

The proposed model can also be used for speaker identification in the presence of noise. When $\mathbf{W}_{\mathrm{n}}$ matches the noise and $\mathbf{W}_{\mathrm{s}}$ is tuned to a particular speaker, the minimal cost attained in (5) is likely to be small. On the other hand, using a dictionary $\mathbf{W}'_{\mathrm{s}}$ tuned to another speaker, the cost is likely to be higher as the dictionary is less suitable for the given data. This suggests a very commonly used classification scheme: a collection of dictionaries is trained,

input : Data $\mathbf{v}$, dictionary $\mathbf{W} = (\mathbf{W}_{\rm s}, \mathbf{W}_{\rm n})$.
output: Nonnegative coefficient vector $\mathbf{h} = (\mathbf{h}_{\rm s}; \mathbf{h}_{\rm n})$.
Define $\mathbf{B} = \mathbf{I} - \frac{1}{\alpha}\begin{pmatrix} \hat{\mathbf{W}}_{\rm s}^{\rm T}\hat{\mathbf{W}}_{\rm s} + \lambda_* \mathbf{I} & \hat{\mathbf{W}}_{\rm s}^{\rm T}\hat{\mathbf{W}}_{\rm n} \\ \hat{\mathbf{W}}_{\rm n}^{\rm T}\hat{\mathbf{W}}_{\rm s} & \mathbf{W}_{\rm n}^{\rm T}\hat{\mathbf{W}}_{\rm n} + \lambda_* \mathbf{I} \end{pmatrix}$,
$\mathbf{A} = \frac{1}{\alpha}\begin{pmatrix} \hat{\mathbf{W}}_{\rm s}^{\rm T} \\ \hat{\mathbf{W}}_{\rm n}^{\rm T} \end{pmatrix}$, $\mathbf{t} = \frac{\lambda}{\alpha}\begin{pmatrix} \mathbf{1} \\ \mathbf{0} \end{pmatrix}$ and $\hat{\mathbf{W}} = \mathbf{Q}\mathbf{W}$.
Initialize $\mathbf{h} = \mathbf{0}$, $\mathbf{b} = \mathbf{A}\mathbf{v}$.
repeat
  $\mathbf{y} = \max\{\mathbf{b} - \mathbf{t}, 0\}$
  $\mathbf{b} = \mathbf{b} + \mathbf{B}(\mathbf{y} - \mathbf{h})$
  $\mathbf{h} = \mathbf{y}$
until *convergence*;
Output $(\mathbf{h}_{\rm s}, \mathbf{h}_{\rm n}) = \mathbf{h}$.

Algorithm 1: Proximal methods for solving the low-rank NMF problem with $\beta = 2$, given the dictionaries $\mathbf{W}_{\rm n}$ and $\mathbf{W}_{\rm s}$.

one per individual speaker. At testing, a collection of data vectors is encoded in each of the dictionaries by solving (5). The class assignment is made based on the minimum cost attained by the solutions, sometimes with the help of pooling or voting in time.

4.1. Low-rank sparse NMF via convex optimization

When $\beta = 2$, the proposed low-rank NMF problem (4) with fixed dictionaries can be solved using a first order convex optimization algorithm. The solution is obtained via proximal methods [27], which split the objective function (5) into a smooth part (the fitting and the low-rank terms), and a non-differentiable part (the ℓ_1 norm of the activation vector). The algorithm iterates between a gradient descent step on the smooth function and an application of the proximal operator (which assumes a closed form of one-sided soft-thresholding), as detailed in Algorithm 1. This algorithm is conceptually very similar to the iterative shrinkage and thresholding algorithm (ISTA) [28]. We do not use this algorithm as an explicit tool, but rather as a motivation of the architecture of a learnable deterministic process.

5. LEARNABLE PURSUIT PROCESSES

In the model-centric setting described in Section 4, the time-frequency representation of the enhanced signal is obtained by solving the pursuit problem (5). Note that this optimization problem implicitly defines a deterministic mapping that assigns to each input vector $\mathbf{v} \in \mathbb{R}^F$ a pair of codes $\mathbf{h}_{\rm s} \in \mathbb{R}^Q$ and $\mathbf{h}_{\rm n} \in \mathbb{R}^Q$.

We propose a process-centric approach for speech enhancement in which we aim at explicitly construct a parametric regressors $(\mathbf{h}_{\rm s}, \mathbf{h}_{\rm n}) = \boldsymbol{h}_{\boldsymbol{\Theta}}(\mathbf{v})$, with some set of parameters, collectively denoted as $\boldsymbol{\Theta}$, capable of accurately separating the speech from the background noise for a given training sample $\mathcal{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_N\}$. We denote by $\mathcal{F}$ the family of the parametric functions $\boldsymbol{h}_{\boldsymbol{\Theta}}$. Here, each $\mathbf{v}_i$ represents the magnitude spectrum of a mixture of speech and noise; training samples may come from many different speakers and noises, or be specific to a single speaker, or single noise category, or both.

Following [20, 21, 23], we define $\mathcal{F}$ as the family of pursuit processes derived from a *truncated* proximal descent method, in this case Algorithm 1. Each iteration can be described as a function receiving the current state $(\mathbf{b}_{\rm in}, \mathbf{h}_{\rm in})$ and producing the next state $(\mathbf{b}_{\rm out}, \mathbf{h}_{\rm out})$ by applying the non-linear transformation $\mathbf{h}_{\rm out} = \max\{\mathbf{b}_{\rm in} - \mathbf{t}, 0\}$, and the linear transformation $\mathbf{b}_{\rm out} = \mathbf{b}_{\rm in} + \mathbf{B}(\mathbf{h}_{\rm out} - \mathbf{h}_{\rm in})$. This can be described by the function $(\mathbf{b}_{\rm out}, \mathbf{h}_{\rm out}) = \boldsymbol{f}_{\mathbf{B},\mathbf{t}}(\mathbf{b}_{\rm in}, \mathbf{h}_{\rm in})$ parametrized by the matrix $\mathbf{B}$ describing the linear transformation, and the vector $\mathbf{t}$ describing the thresholding parameters. Then for a given number of layers T we get a family of functions defined as,

$$\mathcal{F}_T = \left\{ \boldsymbol{h}_{\boldsymbol{\Theta}}(\mathbf{v}) = \boldsymbol{f}_{\mathbf{B},\mathbf{t}} \circ \cdots \circ \boldsymbol{f}_{\mathbf{B},\mathbf{t}}(\mathbf{A}\mathbf{v}, \mathbf{0}) : \boldsymbol{\Theta} = \{\mathbf{A}, \mathbf{B}, \mathbf{t}\} \right\}.$$

The functions in $\mathcal{F}_T$ can be thought as feed-forward multi-layer artificial neural networks with T identical layers. These encoder architectures are continuous and almost everywhere $\mathcal{C}^1$ with respect to the parameters, allowing the use of (sub)gradient descent methods for training. Learning these parameters produces much better approximations than simply truncating the algorithms as mentioned in Section 2 [20, 21]. We train the encoders by minimizing over $\mathcal{V}$ functions of the form

$$\mathcal{L}(\boldsymbol{\Theta}) \quad = \quad \frac{1}{|\mathcal{V}|} \sum_{\mathbf{v}_i \in \mathcal{V}} L(\boldsymbol{\Theta}, \mathbf{v}_i), \tag{6}$$

where $L(\boldsymbol{\Theta}, \mathbf{v}_i)$ is a function that measures the quality of the estimated speech spectrum $\mathbf{h}_{\rm s} = \mathbf{h}(\mathbf{v}_i, \boldsymbol{\Theta})$. We minimize (6) using stochastic gradient descent. We initialize $\boldsymbol{\Theta}$ with the parameters given by Algorithm 1 and then iteratively select a random subset of $\mathcal{V}$ and update the network parameters as $\boldsymbol{\Theta} \leftarrow \boldsymbol{\Theta} - \mu \frac{\partial \mathcal{L}(\boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}}$, where μ is a decaying step, repeating the process until convergence.

Once trained, the parameters $\boldsymbol{\Theta}$ and the dictionaries $\mathbf{W} = (\mathbf{W}_{\rm s}, \mathbf{W}_{\rm n})$ are fixed, and the network is used to sequentially process new data. The latency of the NMF networks (referred henceforth as *NMF encoders*) is of the order of a single frame (hundreds of milliseconds). In the next section we describe several choices of the objective function for training the encoders.

5.1. Training regimes

Training of the proposed NMF encoders is possible under different regimes. We refer as *supervised* to the setting where the training set consists of the noisy speech signal $\mathbf{v}_i$, and the synchronized ground-truth clean speech component $\mathbf{s}_i^*$ (each vector corresponding to the magnitude spectrogram). In that case, we set

$$L_{\rm sup}(\boldsymbol{\Theta}, \mathbf{v}_i) = d_\beta(\mathbf{s}_i^* | \mathbf{W}_{\rm s} \mathbf{h}_{\boldsymbol{\Theta}_{\rm s}}(\mathbf{v}_i)).$$

The reconstructed noise is discarded, as we are only interested in reconstructing the speech signal. In the *unsupervised* setting we only have access to noisy signals as the training data and the objective is used to directly minimize the cost in (5),

$$L_{\rm uns}(\boldsymbol{\Theta}, \mathbf{v}_i) = d_\beta(\mathbf{v}_i | \mathbf{W}\mathbf{h}_{\boldsymbol{\Theta}}(\mathbf{v}_i)) + \frac{\lambda_*}{2} \|\mathbf{h}_{\boldsymbol{\Theta}}(\mathbf{v}_i)\|_2^2 + \lambda \|\mathbf{h}_{\boldsymbol{\Theta}_{\rm s}}(\mathbf{v}_i)\|_1$$

Finally, when the NMF encoders are used for speaker identification, classification performance can be further improved by using a *discriminative* loss. Let $\mathcal{V}^+$ be the set of noisy training examples corresponding to the speaker for which the model is being built, and let $\mathcal{V}^-$ include examples of other speakers. We would like the encoder to minimize the loss $L_{\rm uns}$ on $\mathcal{V}^+$ while simultaneously maximizing it on $\mathcal{V}^-$. The aggregate loss function has the form

$$\begin{aligned} \mathcal{L}_{\rm dis}(\boldsymbol{\Theta}) = \quad & \tfrac{1}{|\mathcal{V}^+|} \textstyle\sum_{\mathbf{v}_i^+ \in \mathcal{V}^+} L_{\rm uns}(\boldsymbol{\Theta}, \mathbf{v}_i^+) + \\ & \tfrac{\gamma}{|\mathcal{V}^-|} \textstyle\sum_{\mathbf{v}_i^- \in \mathcal{V}^-} \max\left\{0, \mu - L_{\rm uns}(\boldsymbol{\Theta}, \mathbf{v}_i^-)\right\}, \end{aligned}$$

where parameter γ governs the relative importance of the positive and negative examples, and the hinge function with the margin parameter μ is used to counter excessive influence of the negatives.

Method	Exact NMF (noise only)		Exact NMF (noise+voice)		NMF Enc. (Untrained)		NMF Enc. (Sup. $\beta = 2$)		NMF Enc. (Sup. $\beta = 0$)		NMF Enc. (Unsup.)	
street	4.67	2.57	6.90	6.77	6.07	6.37	7.21	7.21	8.08	7.70	6.22	6.50
restaurant	3.37	2.52	6.20	6.18	4.92	5.42	6.45	6.27	7.49	7.33	5.14	5.57
car	6.57	3.13	7.89	7.02	6.80	6.68	8.13	7.61	8.95	8.23	7.00	6.84
exhibition	7.38	3.14	8.85	7.95	7.78	7.79	9.15	8.60	10.07	9.46	7.95	7.88
train	6.53	3.24	8.48	7.21	7.55	6.95	8.71	7.78	9.22	8.01	7.70	7.06
airport	4.07	2.86	6.71	6.47	5.40	5.77	7.07	6.88	7.63	7.13	5.60	5.92
average	5.43	2.91	7.51	6.93	6.42	6.50	7.79	7.39	8.57	7.98	6.60	6.63

Table 1. Performance of denoising methods on the GRID dataset with different background noises, in terms of GSDR in dB. For each method, two numbers are given corresponding to the noise-specific (first) and noise-agnostic (second) settings.

The dictionaries are trained by executing the exact NMF algorithm solving (4) independently on the clean speech and noise training examples, producing $\mathbf{W}_{\mathrm{s}}$ and $\mathbf{W}_{\mathrm{n}}$, respectively. The performance of the encoders can be further improved if the dictionaries are updated during the training to match the parameters of the encoder.

In the speech denoising setting, often a semi supervised learning is considered: [4] proposed an algorithm for speech denosing assuming that a model of the speaker is available and adaptively learning the one for the noise, while [6] propose a dual approach in which the model of the noise is the one known. The proposed framework can be used in a semi-supervised setting as well, by updating the noise dictionary in an online manner as discussed in [23].

While the encoder architectures were constructed for an Euclidean data term ($\beta = 2$), they still present sufficient flexibility to be trained with a general divergence.

6. EXPERIMENTAL RESULTS

We evaluated the separation performance of the proposed methods on a subset of the GRID dataset [29] containing ten distinct speakers; each speaker comprising 1000 short clips. Three sets of 200 distinct clips each were used for training, validation, and testing. The GRID clips were resampled to 8 KHz and artificially contaminated by six categories of noise recorded from different real environments (street, restaurant, car, exhibition, train, and airport) taken from the AURORA corpus [30]. The voice and the noise clips were mixed linearly with equal energy (0 dB SNR).

As the evaluation criteria, we used the *source-to-distortion ratio* (SDR) from the BSS-EVAL metrics [31]. Following [32], we computed the global SDR (GSDR) by averaging the SDR over all test clips from the same speaker and noise weighted by the clip duration.

6.1. Comparison of denoising methods

We evaluated the proposed NMF encoders with the different training settings discussed in Section 5.1. In all our examples we used $T = 10$ layers and $q = 50$. As a reference, we also evaluate untrained networks with parameters initialized according to Algorithm 1. We compare these result against exact low-rank NMF with noise model only, and that involving both noise and voice models; both with the Euclidean data term ($\beta = 2$). We used $\lambda = \sqrt{2N}\sigma$ and $\lambda_* = \sqrt{2}\sigma$ with $\sigma = 0.3$ set following [33]; such a setting guarantees that if the data $\mathbf{V}$ consist of n frames of zero-mean white noise of variance σ^2, then both $\mathbf{W}_{\mathrm{n}}\mathbf{H}_{\mathrm{n}}$ and $\mathbf{W}_{\mathrm{s}}\mathbf{H}_{\mathrm{s}}$ are zero.

In all experiments, the spectrogram of each mixture was computed using a window of size 512 and a step size of 128 samples (at 8 KHz sampling rate). Training was performed using 1500 safeguarded gradient descent iterations on a random selection of $10K$ spectral frames for training and the same amount of distinct frames for cross-validation. All methods were trained in two distinct settings: the *noise-specific* setting in which the noise category is assumed to be known as the training is performed only on that noise; and the *noise-agnostic* setting, in which the noise is only known to belong to one of the six categories and the training is performed on a random selection of all the noises.

Table 2. Performance of speaker identification methods with different background noises in terms of classification rate.

Method	Exact RNMF		NMF Enc. (Super.)		NMF Enc. (Discrim.)	
street	0.86	0.93	0.91	0.63	0.91	0.94
restaurant	0.91	0.90	0.89	0.83	0.90	0.97
car	0.90	0.94	0.91	0.65	0.96	0.87
exhibition	0.93	0.94	0.91	0.65	0.95	0.96
train	0.93	0.94	0.88	0.77	0.96	0.95
airport	0.92	0.94	0.85	0.65	0.96	0.95
average	0.91	0.93	0.89	0.69	0.94	0.94

Table 1 summarizes the performance of the compared methods. We observe that the introduction of a low-rank sparse voice model improves the quality of denoising by exact NMF algorithms by over 2 dB GSDR in the noise-specific setting, and over 4 dB GSDR in the noise-agnostic one. The NMF encoder trained in the supervised regime to produce the best approximation of the voice and noise tracks known at training consistently outperforms the exact NMF algorithms and NMF encoders trained in other regimes, achieving over 7 dB GSDR in both the noise-specific and noise-agnostic settings. The use of the Itakura-Saito divergence ($\beta = 0$) in the supervised setting brings further improvement by about 1 dB.

The complexity of the proposed NMF encoders is significantly lower than the one of exact algorithms: a preliminary implementation is over four times faster than real time. The latency of our implementation is in the order of hundreds of milliseconds, while the exact algorithms require a significant amount of data to be observed.

6.2. Comparison of speaker identification methods

We evaluated the classification capabilities of different low rank NMF architectures in combination with two supervised training regimes discussed in Section 5.1, one aimed to produce a good reconstruction of the speech signal and another one optimized to produce the best classification. Table 2 summarizes the classification rates of the compared methods with different noise and voice models. In the noisy case (for which all training was performed), the best performance is achieved when using the NMF encoders with the discriminative loss. This simple example shows the potential of using process-centric NMF for discriminative tasks.

7. REFERENCES

[1] P. C. Loizou, *Speech enhancement: theory and practice*, vol. 30, CRC, 2007.

[2] E. Hänsler and G. Schmidt, *Speech and audio processing in adverse environments*, Springer, 2008.

[3] K. W. Wilson, B. Raj, P. Smaragdis, and A. Divakaran, "Speech denoising using nonnegative matrix factorization with priors," in *ICASSP*, 2008, pp. 4029–4032.

[4] C. Joder, F. Weninger, F. Eyben, D. Virette, and B. Schuller, "Real-time speech separation by semi-supervised nonnegative matrix factorization," in *LVA/ICA*, 2012, pp. 322–329.

[5] G. J. Mysore and P. Smaragdis, "A non-negative approach to semi-supervised separation of speech from noise with the use of temporal dynamics," in *ICASSP*, 2011, pp. 17–20.

[6] Z. Duan, G. J. Mysore, and P. Smaragdis, "Online plca for real-time semi-supervised source separation," in *LVA/ICA*, 2012, pp. 34–41.

[7] D.D. Lee and H.S. Seung, "Learning parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[8] P. Smaragdis, B. Raj, and M. Shashanka, "A probabilistic latent variable model for acoustic modeling," *Advances in models for acoustic processing, NIPS*, vol. 148, 2006.

[9] M. N. Schmidt and R. K. Olsson, "Single-channel speech separation using sparse non-negative matrix factorization," in *INTERSPEECH*, Sep 2006.

[10] M. V. S. Shashanka, B. Raj, and P. Smaragdis, "Sparse Overcomplete Decomposition for Single Channel Speaker Separation," in *ICASSP*, 2007.

[11] M. N. Schmidt, J. Larsen, and F.-T. Hsiao, "Wind noise reduction using non-negative sparse coding," in *MLSP*, Aug 2007, pp. 431–436.

[12] J. F. Gemmeke, T. Virtanen, and A. Hurmalainen, "Exemplar-based sparse representations for noise robust automatic speech recognition," *IEEE Trans. on Audio, Speech, and Lang. Proc.*, vol. 19, no. 7, pp. 2067–2080, 2011.

[13] F. Weninger, M. Wöllmer, J. T. Geiger, B. Schuller, J. F. Gemmeke, A. Hurmalainen, T. Virtanen, and G. Rigoll, "Non-negative matrix factorization for highly noise-robust asr: To enhance or to recognize?," in *ICASSP*, 2012, pp. 4681–4684.

[14] D. Bansal, B. Raj, and P. Smaragdis, "Bandwidth expansion of narrowband speech using non-negative matrix factorization," in *INTERSPEECH*, 2005, pp. 1505–1508.

[15] J. Han, G. J. Mysore, and B. Pardo, "Audio imputation using the non-negative hidden markov model," in *LVA/ICA*, 2012, pp. 347–355.

[16] Q. Wu, L.Q. Zhang, and G.C. Shi, "Robust feature extraction for speaker recognition based on constrained nonnegative tensor factorization," *Journal of Computer Science and Technology*, vol. 25, no. 4, pp. 783–792, 2010.

[17] C. Joder and B. Schuller, "Exploring nonnegative matrix factorization for audio classification: Application to speaker recognition," in *Speech Communication*. VDE, 2012, pp. 1–4.

[18] P.O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *The Journal of Machine Learning Research*, vol. 5, pp. 1457–1469, 2004.

[19] C. Févotte, "Majorization-minimization algorithm for smooth itakura-saito nonnegative matrix factorization," in *ICASSP*. IEEE, 2011, pp. 1980–1983.

[20] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *ICML*, 2010, pp. 399–406.

[21] P. Sprechmann, A. M. Bronstein, and G. Sapiro, "Learning efficient structured sparse models," in *ICML*, 2012.

[22] M. Henaff, K. Jarrett, K. Kavukcuoglu, and Y. LeCun, "Unsupervised learning of sparse features for scalable audio classification," in *ISMIR*, 2011.

[23] P. Sprechmann, A. M. Bronstein, and G. Sapiro, "Real-time online singing voice separation from monaural recordings using robust low-rank modeling," in *ISMIR*, 2012.

[24] P. Sprechmann, A. M. Bronstein, and G. Sapiro, "Learning efficient sparse and low rank models," *arXiv preprint arXiv:1212.3631*, 2012.

[25] C. Févotte and J. Idier, "Algorithms for nonnegative matrix factorization with the β-divergence," *Neural Computation*, vol. 23, no. 9, pp. 2421–2456, 2011.

[26] A. Lefèvre, F. Bach, and C. Févotte, "Online algorithms for nonnegative matrix factorization with the Itakura-Saito divergence," in *WASPAA*, Mohonk, NY, Oct. 2011.

[27] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, "Convex optimization with sparsity-inducing norms," in *Optimization for Machine Learning*. MIT Press, 2011.

[28] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Img. Sci.*, vol. 2, pp. 183–202, March 2009.

[29] M. Cooke, J. Barker, S. Cunningham, and X. Shao, "An audio-visual corpus for speech perception and automatic speech recognition," *J. of the Acoustical Society of America*, vol. 120, pp. 2421, 2006.

[30] D. Pearce and H.-G. Hirsch, "The aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *INTERSPEECH*, 2000, pp. 29–32.

[31] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Trans. on Audio, Speech, and Lang. Proc.*, vol. 14, no. 4, pp. 1462–1469, 2006.

[32] P.-S. Huang, S. D. Chen, P. Smaragdis, and M. Hasegawa-Johnson, "Singing-voice separation from monaural recordings using robust principal component analysis," in *ICASSP*, 2012.

[33] E. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," *Journal of the ACM*, vol. 58, no. 3, May 2011.