Discriminative Nonnegative Dictionary Learning using Cross-Coherence Penalties for Single Channel Source Separation

Emad M. Grais and Hakan Erdogan

Faculty of Engineering and Natural Sciences
Sabanci University, Orhanli Tuzla, 34956, Istanbul, Turkey.
grais@sabanciuniv.edu, haerdogan@sabanciuniv.edu

10.21437/Interspeech.2013-232

Abstract

In this work, we introduce a new discriminative training method for nonnegative dictionary learning. The new method can be used in single channel source separation (SCSS) applications. In SCSS, nonnegative matrix factorization (NMF) is used to learn a dictionary (a set of basis vectors) for each source in the magnitude spectrum domain. The trained dictionaries are then used in decomposing the mixed signal to find the estimate for each source. Learning discriminative dictionaries for the source signals can improve the separation performance. To achieve discriminative dictionaries, we try to avoid the bases set of one source dictionary from representing the other source signals. We propose to minimize cross-coherence between the dictionaries of all sources in the mixed signal. We incorporate a simplified cross-coherence penalty using a regularized NMF cost function to simultaneously learn discriminative and reconstructive dictionaries. The new regularized NMF update rules that are used to discriminatively train the dictionaries are introduced in this work. Experimental results show that using discriminative training gives better separation results than using conventional NMF.

Index Terms: Single channel source separation, nonnegative matrix factorization, discriminative training, dictionary learning.

1. Introduction

In single channel source separation problems, only one observation of the mixed signal is available. The solution of this problem usually relies on training data for each source signal. Nonnegative matrix factorization (NMF) [1] is usually used to train a set of basis vectors (dictionary) for each source signal in the magnitude spectrum domain. NMF is then used to decompose the mixed signal magnitude spectrogram as a weighted linear combination of the trained dictionary entries for all sources in the mixed signal. The estimate for each source is found by summing the decomposition terms that include its corresponding trained basis vectors [2, 3, 4].

One of the main problems of this framework is that the basis vectors for each source dictionary can represent the other source signals. When a dictionary of one source is able to represent the other source signals, the estimated separated signal for this source will contain signals from the other sources that are in the mixed signal. The solution for this problem is to learn the entries for each source dictionary to be more discriminative from the entries of the other sources' dictionaries. Discriminative learning for NMF dictionaries here is not related to discriminative NMF which aims to train discriminative basis vectors for a single source [5]. Discriminative NMF is out of the scope of this paper. The novelty in this paper is to train nonnegative discriminative dictionaries simultaneously for the source signals. Discriminative dictionary for a source signal in this paper means that, a dictionary that is good in representing this source signal and at the same time is bad in representing the other source signals [6]. Enforcing the dictionary for each source signal to poorly represent the other source signals increases the separation capability of the NMF decomposition of the observed mixed signal. The NMF solution for training a dictionary for a source signal is usually not unique, and there are multiple solutions that can be used as a dictionary for the same source. In this paper, we are seeking a dictionary for each source during the training that minimizes the reconstruction error and preventing its bases from representing the other sources. To prevent the dictionaries from representing the sources of each other, we propose to minimize the cross-coherence between the source dictionaries. To achieve good representative and discriminative dictionaries with nonnegativity constraints, we formulate these objectives using a regularized NMF cost function with simplified cross-coherence penalties. The new update rules for simultaneously training the dictionaries that solve the regularized NMF cost function are introduced in this paper.

This paper is organized as follows: Section 2 shows a brief introduction about NMF. Section 3 describes SCSS using NMF. In Sections 4 and 5, we introduce the discriminative training for the nonnegative dictionaries which is our main contribution in this paper. In the remaining sections we present our experimental results.

2. Non-negative matrix factorization

Non-negative matrix factorization decomposes any nonnegative matrix $\boldsymbol{V} \in \Re_+^{M\times N}$ into a nonnegative basis matrix $\boldsymbol{B} \in \Re_+^{M\times K}$ and a nonnegative gains matrix $\boldsymbol{G} \in \Re_+^{K\times N}$ as follows:

$$\boldsymbol{V} \approx \boldsymbol{BG}, \tag{1}$$

where $K < M, N$. The solution of the matrices $\boldsymbol{B}$ and $\boldsymbol{G}$ can be found by solving the following generalized Kullback-Leibler divergence cost function [1]:

$$\min_{\boldsymbol{B},\boldsymbol{G}} D_{KL}\left(\boldsymbol{V} \,||\, \boldsymbol{BG}\right), \tag{2}$$

where

$$D_{KL}\left(\boldsymbol{V} \,||\, \boldsymbol{BG}\right) = \sum_{k,l}\left(\boldsymbol{V}_{k,l}\log\frac{\boldsymbol{V}_{k,l}}{(\boldsymbol{BG})_{k,l}} - \boldsymbol{V}_{k,l} + (\boldsymbol{BG})_{k,l}\right),$$

This work was supported by Turk-Telekom under grant number 3014-06.

subject to elements of $\boldsymbol{B}, \boldsymbol{G} \geq 0$. The solution for equation (2) can be computed by alternating updates of $\boldsymbol{B}$ and $\boldsymbol{G}$ as follows:

$$\boldsymbol{B} \leftarrow \boldsymbol{B} \otimes \frac{\boldsymbol{V}}{\boldsymbol{B}\boldsymbol{G}}\boldsymbol{G}^T, \qquad \boldsymbol{G} \leftarrow \boldsymbol{G} \otimes \frac{\boldsymbol{B}^T}{\boldsymbol{B}^T\boldsymbol{1}}\frac{\boldsymbol{V}}{\boldsymbol{B}\boldsymbol{G}}, \quad (3)$$

where $\boldsymbol{1}$ is a matrix of ones with the same size of $\boldsymbol{V}$, the operation $\otimes$ is element-wise multiplication, and divisions also are element-wise operations. The matrices $\boldsymbol{B}$ and $\boldsymbol{G}$ are initialized by positive random numbers and the multiplicative update rules in equation (3) guarantee the nonnegativity of the decomposition matrices.

3. Single channel source separation

In single channel source separation problems, we try to find estimates of source signals that are mixed on a single channel $y(t)$. For simplicity, in this paper we assume the number of sources is two. This problem is usually solved in the short time Fourier transform (STFT) domain. Let $Y(t,f)$ be the STFT of $y(t)$, where t represents the frame index and f is the frequency-index. Due to the linearity of the STFT, we have

$$Y(t,f) = S^{(1)}(t,f) + S^{(2)}(t,f), \quad (4)$$

where $S^{(1)}(t,f)$ and $S^{(2)}(t,f)$ are the unknown STFT of the first and second sources in the mixed signal. In this framework [7, 8, 9], the phase angles of the STFT were usually ignored. Hence, we can approximate the magnitude spectrum of the measured signal as the sum of source signals' magnitude spectra as follows:

$$|Y(t,f)| \approx \left|S^{(1)}(t,f)\right| + \left|S^{(2)}(t,f)\right|. \quad (5)$$

We can write the magnitude spectrogram in matrix form as follows:

$$\boldsymbol{Y} \approx \boldsymbol{S}^{(1)} + \boldsymbol{S}^{(2)}. \quad (6)$$

where $\boldsymbol{S}^{(1)}$ and $\boldsymbol{S}^{(2)}$ are the unknown magnitude spectrograms of the source signals and need to be estimated using the observed mixed signal and the training data. The magnitude spectrogram for the observed signal $y(t)$ is obtained by taking the magnitude of the DFT of the windowed signal.

As shown in [2, 3, 10, 11], the main idea to solve for $\boldsymbol{S}^{(1)}$ and $\boldsymbol{S}^{(2)}$ is to use NMF to train a basis matrix for each source. NMF is used to decompose the magnitude spectrogram of each source training data as follows:

$$\boldsymbol{S}^{(1)}_{train} \approx \boldsymbol{B}^{(1)}\boldsymbol{G}^{(1)}_{train}, \qquad \boldsymbol{S}^{(2)}_{train} \approx \boldsymbol{B}^{(2)}\boldsymbol{G}^{(2)}_{train}, \quad (7)$$

where $\boldsymbol{S}^{(1)}_{train} \in \Re_+^{M\times N_1}$ and $\boldsymbol{S}^{(2)}_{train} \in \Re_+^{M\times N_2}$ are the magnitude spectrograms of the training data for the first and second sources respectively, $\boldsymbol{B}^{(1)} \in \Re_+^{M\times K_1}$ and $\boldsymbol{B}^{(2)} \in \Re_+^{M\times K_2}$ are considered as trained dictionaries that are used in mixed signal decomposition as shown later. The update rules in equation (3) are used to decompose $\boldsymbol{S}^{(1)}_{train}$ and $\boldsymbol{S}^{(2)}_{train}$ in equation (7). After each NMF iteration the columns of the basis matrices are normalized using the ℓ^2 norm and the gain matrices are calculated accordingly.

NMF is then used to decompose the mixed signal magnitude spectrogram $\boldsymbol{Y}$ with the trained basis matrices $\boldsymbol{B}^{(1)}$ and $\boldsymbol{B}^{(2)}$ as follows:

$$\boldsymbol{Y} \approx \left[\boldsymbol{B}^{(1)}, \boldsymbol{B}^{(2)}\right]\boldsymbol{G} \quad \text{or} \quad \boldsymbol{Y} \approx \left[\boldsymbol{B}^{(1)} \quad \boldsymbol{B}^{(2)}\right]\begin{bmatrix}\boldsymbol{G}^{(1)}\\ \boldsymbol{G}^{(2)}\end{bmatrix}. \quad (8)$$

The only unknown here is the gains matrix $\boldsymbol{G}$ since the bases matrix $\boldsymbol{B}_{sep} = \left[\boldsymbol{B}^{(1)}, \boldsymbol{B}^{(2)}\right]$ is fixed. The update rule of the gains matrix in equation (3) is used to find $\boldsymbol{G}$. After finding the value of $\boldsymbol{G}$, the initial estimate for each source magnitude spectrogram can be found as

$$\widetilde{\boldsymbol{S}}^{(1)} = \boldsymbol{B}^{(1)}\boldsymbol{G}^{(1)}, \quad \widetilde{\boldsymbol{S}}^{(2)} = \boldsymbol{B}^{(2)}\boldsymbol{G}^{(2)}. \quad (9)$$

The initial estimated magnitude spectrograms $\widetilde{\boldsymbol{S}}^{(1)}$ and $\widetilde{\boldsymbol{S}}^{(2)}$ are used to build spectral masks [3, 10] as follows:

$$\boldsymbol{H}^{(1)} = \frac{\widetilde{\boldsymbol{S}}^{(1)}}{\widetilde{\boldsymbol{S}}^{(1)} + \widetilde{\boldsymbol{S}}^{(2)}}, \quad \boldsymbol{H}^{(2)} = \frac{\widetilde{\boldsymbol{S}}^{(2)}}{\widetilde{\boldsymbol{S}}^{(1)} + \widetilde{\boldsymbol{S}}^{(2)}}, \quad (10)$$

where the divisions are done element-wise. The final estimate of each source STFT can be obtained as follows:

$$\hat{S}^{(z)}(t,f) = \boldsymbol{H}^{(z)}(t,f)\, Y(t,f), \quad \forall z \in \{1,2\}, \quad (11)$$

where $Y(t,f)$ is the STFT of the observed mixed signal in equation (5), $\boldsymbol{H}^{(1)}(t,f)$ and $\boldsymbol{H}^{(2)}(t,f)$ are the entries at row f and column t of the spectral masks $\boldsymbol{H}^{(1)}$ and $\boldsymbol{H}^{(2)}$ respectively. The spectral mask entries scale the observed mixed signal STFT entries according to the contribution of each source in the mixed signal [12, 13, 14]. The estimated source signals $\hat{s}^{(1)}(t)$ and $\hat{s}^{(2)}(t)$ can be found by inverse STFT of $\hat{S}^{(1)}(t,f)$ and $\hat{S}^{(2)}(t,f)$ respectively.

One of the main problems in this framework is that the basis matrices $\boldsymbol{B}^{(2)}$ and $\boldsymbol{B}^{(1)}$ sometimes can represent the first and second source signals respectively. When one basis matrix for one source is able to represent the second source, the estimate for each source in equation (9) will contain residual signals from the other source which degrades the separation performance. To fix this problem, the basis matrix for one source should be good in representing this source signal and at the same time to be bad in representing the other source. To achieve this goal, we will solve the two formulas in (7) simultaneously to consider the reconstruction error in equation (2) and the discriminativity penalties between the basis matrices $\boldsymbol{B}^{(1)}$ and $\boldsymbol{B}^{(2)}$.

4. Learning the dictionaries

The matrix $\boldsymbol{B}$ in equation (1) can be seen as a dictionary with nonnegativity constraints that represents each column $\boldsymbol{v}$ in $\boldsymbol{V}$ as a weighted linear combination of its constituent vectors as follows:

$$\boldsymbol{v}_n = \sum_{k=1}^{K} \boldsymbol{g}_{kn}\boldsymbol{b}_k, \quad \boldsymbol{b}_k \in \boldsymbol{B}, \quad (12)$$

where $\boldsymbol{v}_n$ is the column n in matrix $\boldsymbol{V}$, $\boldsymbol{b}_k$ is the column k in matrix $\boldsymbol{B}$ and $\boldsymbol{g}_{kn}$ is its weight in the gains matrix $\boldsymbol{G}$. One of the main quality measurements of a dictionary is its coherence [15]. The coherence is a measurement of the redundancy of the dictionary and small coherence indicated that the dictionary is not far from an orthogonal basis. Minimizing the coherence of a dictionary is defined as follows:

$$\min_{\boldsymbol{B}} \mu(\boldsymbol{B}), \quad \text{where } \mu(\boldsymbol{B}) = \max_{\boldsymbol{b}_i,\boldsymbol{b}_j\in\boldsymbol{B}} < \boldsymbol{b}_i, \boldsymbol{b}_j >, \quad (13)$$

and $< .,. >$ is the dot product. Given two dictionaries for two different source signals, we try to minimize the coherence between the first dictionary $\boldsymbol{B}^{(1)}$ with respect to the second dictionary $\boldsymbol{B}^{(2)}$ which is called cross-coherence [16]. Preventing

the two dictionaries $\boldsymbol{B}^{(1)}$ and $\boldsymbol{B}^{(2)}$ from representing the data for each other can be done by minimizing the following cross-coherence between the two dictionaries as follows:

$$\chi\left(\boldsymbol{B}^{(1)},\boldsymbol{B}^{(2)}\right)=\max_{\boldsymbol{b}_i\in\boldsymbol{B}^{(1)},\boldsymbol{b}_j\in\boldsymbol{B}^{(2)}}<\boldsymbol{b}_i^{(1)},\boldsymbol{b}_j^{(2)}>. \quad (14)$$

We can achieve the minimum of χ when every basis in $\boldsymbol{B}^{(1)}$ is orthogonal to each basis in $\boldsymbol{B}^{(2)}$. Since the two dictionaries are nonnegative matrices, if the set of bases in $\boldsymbol{B}^{(1)}$ are orthogonal on the set of bases in $\boldsymbol{B}^{(2)}$ we expect that some rows in $\boldsymbol{B}^{(1)}$ are zeros and their corresponding rows in $\boldsymbol{B}^{(2)}$ may have nonzero values and vice versa. We need to simplify the cross-coherence in (14) with another formulation that can be easily minimized with the nonnegativity constraint. We propose to replace the maximum in (14) with the summation. We define the simplified cross-coherence penalty as follows:

$$\xi\left(\boldsymbol{B}^{(1)},\boldsymbol{B}^{(2)}\right)=\sum_{\boldsymbol{b}_i\in\boldsymbol{B}^{(1)}}\sum_{\boldsymbol{b}_j\in\boldsymbol{B}^{(2)}}<\boldsymbol{b}_i^{(1)},\boldsymbol{b}_j^{(2)}>. \quad (15)$$

The obvious minimizer of ξ is still the set of bases in $\boldsymbol{B}^{(1)}$ that are orthogonal on the set of bases in $\boldsymbol{B}^{(2)}$.

The formula in (15) can be seen from a least squares point of view, ignoring the nonnegativity constraint, as follows: Given a spectrogram frame (vector) $\boldsymbol{x}$ of the training data of the first source that can be represented well using the first dictionary as $\boldsymbol{x}=\boldsymbol{B}^{(1)}\boldsymbol{\gamma}_1$, if we try to represent $\boldsymbol{x}$ using the second dictionary by minimizing the following least squares problem as follows:

$$\hat{\boldsymbol{\gamma}}_2=\arg\min_{\boldsymbol{\gamma}_2}\left\|\boldsymbol{x}-\boldsymbol{B}^{(2)}\boldsymbol{\gamma}_2\right\|_2^2,$$

the pseudo-inverse (least squares) solution for $\hat{\boldsymbol{\gamma}}_2$ will be

$$\hat{\boldsymbol{\gamma}}_2=\left(\boldsymbol{B}^{(2)^T}\boldsymbol{B}^{(2)}\right)^{-1}\boldsymbol{B}^{(2)^T}\boldsymbol{B}^{(1)}\boldsymbol{\gamma}_1.$$

From the previous formula, if we want $\boldsymbol{x}$ not to be represented by $\boldsymbol{B}^{(2)}$ we need $\boldsymbol{B}^{(2)^T}\boldsymbol{B}^{(1)}=\boldsymbol{0}$. Minimizing the entries of the multiplication $\boldsymbol{B}^{(2)^T}\boldsymbol{B}^{(1)}$ or $\boldsymbol{B}^{(1)^T}\boldsymbol{B}^{(2)}$ minimizes the possibility of each source dictionary from representing the other sources.

The dictionaries $\boldsymbol{B}^{(1)}$ and $\boldsymbol{B}^{(2)}$ that minimize ξ in (15) may not be good representatives for $\boldsymbol{S}_{train}^{(1)}$ and $\boldsymbol{S}_{train}^{(2)}$ in equation (7). We use regularized NMF to find the basis matrices $\boldsymbol{B}^{(1)}$ and $\boldsymbol{B}^{(2)}$ that can solve equation (7) and minimize (15) at the same time.

5. Discriminative dictionary learning

To find better solutions for (7) and to find the basis matrices $\boldsymbol{B}^{(1)}$ and $\boldsymbol{B}^{(2)}$ that minimize (15) at the same time, we form the regularized NMF to solve (7) simultaneously as follows:

$$C=D_{KL}\left(\boldsymbol{S}_{train}^{(1)}\,||\,\boldsymbol{B}^{(1)}\boldsymbol{G}_{train}^{(1)}\right)+\alpha D_{KL}\left(\boldsymbol{S}_{train}^{(2)}\,||\,\boldsymbol{B}^{(2)}\boldsymbol{G}_{train}^{(2)}\right)+\lambda\sum_{i,j}\left(\boldsymbol{B}^{(1)^T}\boldsymbol{B}^{(2)}\right)_{ij}, \quad (16)$$

where α is a regularization parameter that can be used to balance the energy scale differences between the two sources training data, λ is a regularization parameter that controls the trade-off between the NMF reconstruction error terms and the simplified cross-coherence penalty term. The last term in equation (16) enforces the discriminativity between the two dictionaries. The value of α can be determined for example from the ratio between the sum of all entries in matrix $\boldsymbol{S}_{train}^{(1)}$ to the sum of $\boldsymbol{S}_{train}^{(2)}$ entries.

To find the update rule solutions for the basis matrices we follow the same procedures as in [7, 17, 18]. We express the gradient with respect to $\boldsymbol{B}^{(1)}$ of the cost function in equation (16) as a difference of two positive terms $\nabla^+_{\boldsymbol{B}^{(1)}}C$ and $\nabla^-_{\boldsymbol{B}^{(1)}}C$ as follows:

$$\nabla_{\boldsymbol{B}^{(1)}}C=\nabla^+_{\boldsymbol{B}^{(1)}}C-\nabla^-_{\boldsymbol{B}^{(1)}}C. \quad (17)$$

The cost function is shown to be nonincreasing under the following update rule [7, 17]

$$\boldsymbol{B}^{(1)}\leftarrow\boldsymbol{B}^{(1)}\otimes\frac{\nabla^-_{\boldsymbol{B}^{(1)}}C}{\nabla^+_{\boldsymbol{B}^{(1)}}C}. \quad (18)$$

The gradient with respect to $\boldsymbol{B}^{(1)}$ of the cost function in equation (16) can be calculated as follows:

$$\nabla_{\boldsymbol{B}^{(1)}}C=\left(\boldsymbol{1}-\frac{\boldsymbol{S}_{train}^{(1)}}{\boldsymbol{B}^{(1)}\boldsymbol{G}_{train}^{(1)}}\right)\boldsymbol{G}_{train}^{(1)^T}+\lambda\boldsymbol{B}^{(2)}\boldsymbol{1}^{(2)}, \quad (19)$$

where $\boldsymbol{1}$ is a matrix of ones with the same size of $\boldsymbol{S}_{train}^{(1)}$, and $\boldsymbol{1}^{(2)}\in\Re_+^{K_2\times K_1}$ is a matrix of ones. The gradient can be divided as in equation (17) as

$$\nabla^-_{\boldsymbol{B}^{(1)}}C=\frac{\boldsymbol{S}_{train}^{(1)}}{\boldsymbol{B}^{(1)}\boldsymbol{G}_{train}^{(1)}}\boldsymbol{G}_{train}^{(1)^T},\quad \nabla^+_{\boldsymbol{B}^{(1)}}C=\boldsymbol{1}\boldsymbol{G}_{train}^{(1)^T}+\lambda\boldsymbol{B}^{(2)}\boldsymbol{1}^{(2)}. \quad (20)$$

The final update rule for matrix $\boldsymbol{B}^{(1)}$ can be written from equations (18, 20) as follows:

$$\boldsymbol{B}^{(1)}\leftarrow\boldsymbol{B}^{(1)}\otimes\frac{\frac{\boldsymbol{S}_{train}^{(1)}}{\boldsymbol{B}^{(1)}\boldsymbol{G}_{train}^{(1)}}\boldsymbol{G}_{train}^{(1)^T}}{\boldsymbol{1}\boldsymbol{G}_{train}^{(1)^T}+\lambda\boldsymbol{B}^{(2)}\boldsymbol{1}^{(2)}}. \quad (21)$$

The only difference between the update rule of the basis matrix in equation (21) and equation (3) is the additional term in the denominator due to the cross-coherence penalty term.

Following the same procedures, the update rules for $\boldsymbol{B}^{(2)}$ is

$$\boldsymbol{B}^{(2)}\leftarrow\boldsymbol{B}^{(2)}\otimes\frac{\frac{\boldsymbol{S}_{train}^{(2)}}{\boldsymbol{B}^{(2)}\boldsymbol{G}_{train}^{(2)}}\boldsymbol{G}_{train}^{(2)^T}}{\boldsymbol{1}\boldsymbol{G}_{train}^{(2)^T}+\lambda_2\boldsymbol{B}^{(1)}\boldsymbol{1}^{(1)}}, \quad (22)$$

where $\lambda_2=\lambda/\alpha$, and $\boldsymbol{1}^{(1)}\in\Re_+^{K_1\times K_2}$ is a matrix of ones.

To see the effect of adding the simplified cross-coherence penalties between the two basis dictionaries, we can rewrite the update rules in equations (21) and (22) in more details as follows:

$$\boldsymbol{B}_{ij}^{(1)}\leftarrow\boldsymbol{B}_{ij}^{(1)}\frac{\sum_k\boldsymbol{G}_{train_{jk}}^{(1)}\boldsymbol{S}_{train_{ik}}^{(1)}/\left(\boldsymbol{B}^{(1)}\boldsymbol{G}_{train}^{(1)}\right)_{ik}}{\left(\sum_m\boldsymbol{G}_{train_{jm}}^{(1)}\right)+\lambda\sum_l\boldsymbol{B}_{il}^{(2)}}, \quad (23)$$

$$\boldsymbol{B}_{ij}^{(2)}\leftarrow\boldsymbol{B}_{ij}^{(2)}\frac{\sum_k\boldsymbol{G}_{train_{jk}}^{(2)}\boldsymbol{S}_{train_{ik}}^{(2)}/\left(\boldsymbol{B}^{(2)}\boldsymbol{G}_{train}^{(2)}\right)_{ik}}{\left(\sum_m\boldsymbol{G}_{train_{jm}}^{(2)}\right)+\lambda_2\sum_l\boldsymbol{B}_{il}^{(1)}}. \quad (24)$$

We can see that, each row entry in matrix $\boldsymbol{B}^{(1)}$ is divided over the sum of the entries of its corresponding row in matrix $\boldsymbol{B}^{(2)}$ and vice versa. The extra term in the denominators penalizes

the rows with smaller values in one dictionary matrix more than penalizing their corresponding rows with higher values in the second dictionary. Making some rows in one dictionary to be close to zeros while their corresponding rows in the second dictionary have some values improves the discriminativity of the dictionaries as shown in equation (15).

The multiplicative update rules for the gain matrices $\boldsymbol{G}_{train}^{(1)}$ and $\boldsymbol{G}_{train}^{(2)}$ in (16) are exactly the same as the update rule for the gains matrix in equation (3). All basis and gain matrices are initialized by positive random numbers. After solving (16) and finding solutions for $\boldsymbol{B}^{(1)}$, $\boldsymbol{B}^{(2)}$, $\boldsymbol{G}_{train}^{(1)}$, and $\boldsymbol{G}_{train}^{(2)}$ the basis matrices $\boldsymbol{B}^{(1)}$ and $\boldsymbol{B}^{(2)}$ are used in mixed signal decomposition in (8). After decomposing the mixed signal, equations (9) to (11) are used to estimate the source signals $\hat{s}^{(1)}(t)$ and $\hat{s}^{(2)}(t)$.

6. Experiments and Discussion

We applied the proposed algorithm to separate a speech signal from a background piano music signal. Our main goal was to get a clean speech signal from a mixture of speech and piano signals. We simulated our algorithm on a collection of speech and piano data at 16kHz sampling rate. For speech data, we used the training and testing male speech data from the TIMIT database. For music data, we downloaded piano music data from the piano society web site [19]. We used 12 pieces with approximate 50 minutes total duration from different composers but from a single artist for training and left out one piece for testing. The magnitude spectrograms for the speech and music data were calculated by using the STFT: A Hamming window with 480 points length and 60% overlap was used and the FFT was taken at 512 points, the first 257 FFT points only were used since the conjugate of the remaining 255 points are involved in the first points. We trained 128 basis vectors for each source dictionary, which makes the size of $\boldsymbol{B}^{(\text{speech})}$ and $\boldsymbol{B}^{(\text{music})}$ matrices to be 257×128. In this experiment, we used the same values for the regularization parameters in equations (21, 22) which means $\alpha = 1, \lambda_2 = \lambda$.

The mixed data was formed by adding random portions of the test music file to 20 speech files from the test data of the TIMIT database at speech to music ratio of 0 dB. The audio power levels of each file were found using the "speech voltmeter" program from the G.191 ITU-T STL software suite [20].

Performance measurements of the separation algorithm were done using the signal to distortion ratio (SDR) and the signal to interference ratio (SIR) [21]. The average SDR and SIR over the 20 test utterances are reported. The source to distortion ratio (SDR) is defined as the ratio of the target energy to all errors in the reconstructed signal. The target signal is defined as the projection of the predicted signal onto the original speech signal. Signal to interference ratio (SIR) is defined as the ratio of the target energy to the interference error due to the music signal only. The higher SDR and SIR we measure the better performance we achieve.

Figure 1 shows the SDR and SIR values in dB for the estimated speech signal with different values for the regularization parameter λ. We can see that, increasing the value of λ until $\lambda = 100$ improves the SDR and SIR values. That means enforcing cross-incoherence between two sources' dictionaries gives better separation results and improves the signal to distortion ratio of the estimated speech signal. When $\lambda > 100$ the SIR is increasing but SDR is decreasing. Increasing λ prevents the bases in the dictionary $\boldsymbol{B}^{(\text{speech})}$ to be able to represent the music signal and also preventing the bases in $\boldsymbol{B}^{(\text{music})}$ to be able to represent the speech signal which improves the SIR. However, increasing the value of $\lambda > 100$ makes each source bases in $\boldsymbol{B}^{(\text{speech})}$ and $\boldsymbol{B}^{(\text{music})}$ to start losing their ability to fully represent their own source signals, which leads to decreasing the values of SDR. According to the shown figure, a good candidate value for λ that improves both SDR and SIR values for the used data sets is 100. Comparing the results of using only NMF without any constraint (λ=0), we can see from the shown figure that discriminative training for the source bases models by using cross-coherence penalties can improve the performance of the separation process.

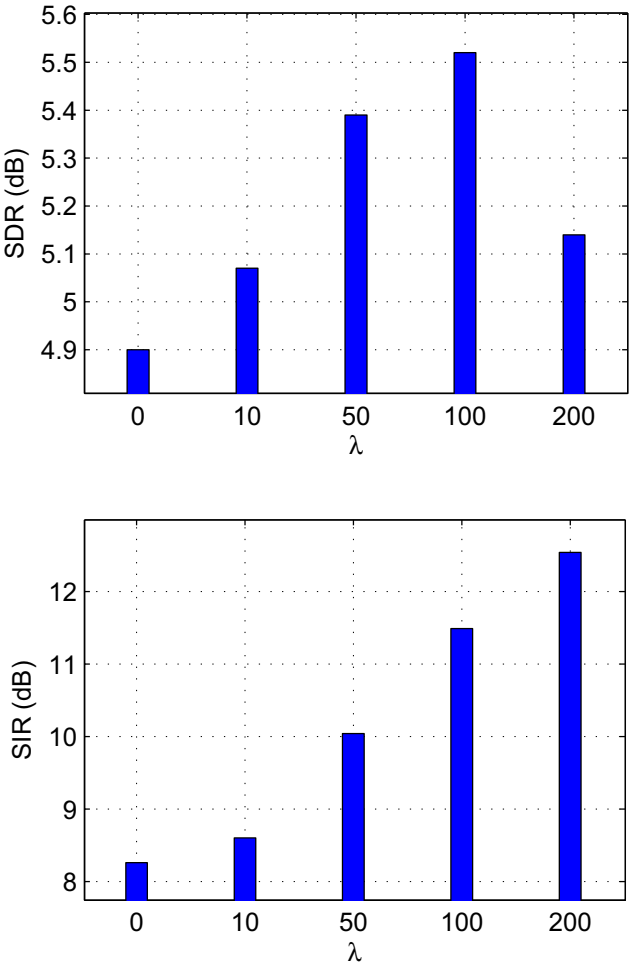

Figure 1: SDR and SIR in dB for the estimated speech signal.

7. Conclusions

In this paper, we introduced a new discriminative training method for NMF dictionary models. The main idea was to prevent the dictionary of each source from representing the other sources by minimizing the cross-coherence between the source dictionaries. The proposed training method improved the performance of source separation.

8. References

[1] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," *Advances in Neural Information Processing Systems*, vol. 13, pp. 556–562, 2001.

[2] M. N. Schmidt and R. K. Olsson, "Single-channel speech separation using sparse non-negative matrix factorization," in *International Conference on Spoken Language Processing (INTERSPEECH)*, 2006.

[3] E. M. Grais and H. Erdogan, "Spectro-temporal post-smoothing in NMF based single-channel source separation," in *European Signal Processing Conference (EUSIPCO)*, 2012.

[4] ——, "Single channel speech music separation using nonnegative matrix factorization with sliding window and spectral masks," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2011.

[5] N. Boulanger-Lewandowski, Y. Bengio, and P. Vincent, "Discriminative nonnegative matrix factorization for multiple pitch estimation," in *ISMIR*, 2012.

[6] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Discriminative learned dictionaries for local image analysis," in *Computer Vision and Pattern Recognition (CVPR)*, 2008.

[7] T. Virtanen, "Monaural sound source separation by non-negative matrix factorization with temporal continuity and sparseness criteria," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, pp. 1066–1074, Mar. 2007.

[8] E. M. Grais and H. Erdogan, "Adaptation of speaker-specific bases in non-negative matrix factorization for single channel speech-music separation," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2011.

[9] ——, "Single channel speech-music separation using matching pursuit and spectral masks," in *IEEE Conference on Signal Processing and Communications Applications (SIU)*, 2011.

[10] ——, "Single channel speech music separation using nonnegative matrix factorization and spectral masks," in *International Conference on Digital Signal Processing*, 2011.

[11] E. M. Grais, I. S. Topkaya, and H. Erdogan, "Audio-Visual speech recognition with background music using single-channel source separation," in *IEEE Conference on Signal Processing and Communications Applications (SIU)*, 2012.

[12] C. Fevotte, N. Bertin, and J.-L. Durrieu, "Nonnegative matrix factorization with the itakura-saito divergence. With application to music analysis," *Neural Computation*, vol. 21, no. 3, pp. 793–830, 2009.

[13] E. M. Grais and H. Erdogan, "Hidden Markov Models as priors for regularized nonnegative matrix factorization in single-channel source separation," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2012.

[14] ——, "Regularized nonnegative matrix factorization using gaussian mixture priors for supervised single channel source separation," *Computer Speech and Language*, 2013.

[15] P. Jost, P. Vandergheynst, and P. Frossard, "Tree-Based pursuit: Algorithm and properties," *IEEE Trans. Signal Process*, vol. 54, pp. 4685–4697, Dec. 2006.

[16] K. Schnass and P. Vandergheynst, "Dictionary preconditioning for greedy algorithms," *IEEE Transactions on Signal Processing*, vol. 56, no. 5, pp. 1994–2002, 2008.

[17] N. Bertin, R. Badeau, and E. Vincent, "Enforcing harmonicity and smoothness in bayesian nonnegative matrix factorization applied to polyphonic music transcription," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 538–549, 2010.

[18] E. M. Grais and H. Erdogan, "Gaussian mixture gain priors for regularized nonnegative matrix factorization in single-channel source separation," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2012.

[19] URL, "http://pianosociety.com," 2009.

[20] ——, "http://www.itu.int/rec/T-REC-G.191/en," 2009.

[21] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–69, Jul. 2006.