LETTER Communicated by Hakan Erdogan

Linear Methods for Efficient and Fast Separation of Two Sources Recorded with a Single Microphone

Saurabh Bhargava
saurabh@ini.ethz.ch
Florian Blättler
florian@ini.ethz.ch
Sepp Kollmorgen
skollmor@ini.ethz.ch
Shih-Chii Liu
shih@ini.ethz.ch
Richard H. R. Hahnloser
rich@ini.ethz.ch
Institute of Neuroinformatics, University of Zurich and ETH Zurich,
Zurich, 8057, Switzerland

This letter addresses the problem of separating two speakers from a single microphone recording. Three linear methods are tested for source separation, all of which operate directly on sound spectrograms: (1) eigenmode analysis of covariance difference to identify spectro-temporal features associated with large variance for one source and small variance for the other source; (2) maximum likelihood demixing in which the mixture is modeled as the sum of two gaussian signals and maximum likelihood is used to identify the most likely sources; and (3) suppression-regression, in which autoregressive models are trained to reproduce one source and suppress the other. These linear approaches are tested on the problem of separating a known male from a known female speaker. The performance of these algorithms is assessed in terms of the residual error of estimated source spectrograms, waveform signal-to-noise ratio, and perceptual evaluation of speech quality scores. This work shows that the algorithms compare favorably to nonlinear approaches such as nonnegative sparse coding in terms of simplicity, performance, and suitability for real-time implementations, and they provide benchmark solutions for monaural source separation tasks.

1 Introduction

The problem of recovering underlying source signals from their mixtures is called source separation. It has been an area of intense research for a long time and has received much attention recently because of the potential applications for hearing aid systems, signal preprocessing in speech

Neural Computation **27, 2231–2259** (2015)
doi:10.1162/NECO_a_00776

recognition systems, and network tomography and medical signal processing in general (Hyvärinen, Karhunen, & Oja, 2001). When the problem is (over-) determined, that is, when the number of sources is no larger than the number of available mixtures or sensor recordings, then generic assumptions such as statistical independence of sources can be used for successful demixing (Hyvärinen et al., 2001). However, most of the time, the problem is underdetermined: the number of sources is larger than the number of available mixtures; hence, more specific assumptions must be made for demixing. One of the most difficult cases is source separation from a single microphone, which is the task of segregating a target source from a masker using a single-channel recording. This task has been proven to be extremely challenging (Wang & Brown, 2006) because one can only use the intrinsic acoustic properties of the target and the masker with no additional cues.

Essentially three general approaches have been proposed for monaural separation: speech enhancement, computational auditory scene analysis (CASA), and model-based methods such as nonnegative matrix factorization (NMF). The linear methods have mostly been used for speech enhancement in noisy environments. These methods include spectral subtraction, Wiener filtering, and subspace-based approaches (Loizou, 2013). The subspace methods for speech enhancement are based on the principle that sources are usually confined to a subspace of Euclidean space. Consequently, methods have been developed that compute these source-specific subspaces. Some of the widely used methods are singular value decomposition (SVD), eigenvalue decomposition (EVD), and subspace tracking algorithms (Loizou, 2013). These methods have been primarily used for speech enhancement of noisy speech (Jensen, Hansen, Hansen, & Sorensen, 1995; Hansen & Jensen, 2005, 2007; Weiss, 2009). Another approach to deal with source separation is to compute an ideal binary mask (IBM) for the target source, considered one of the main goals in CASA systems for speech segregation (Wang & Brown, 2006). An IBM is defined as a binary matrix in which a matrix element is 1 if the power of the target source is known to be higher than that of the masker and 0 otherwise in the corresponding time frequency bin of the mixture signal. Binary masking and related approaches have been used extensively for demixing (Aoki et al., 2001; Brungart, Chang, Simpson, & Wang, 2006; Han & Wang, 2012; Jourjine, Rickard, & Yilmaz, 2000; Nguyen, Belouchrani, Abed-Meraim, & Boashash, 2001; Rickard, Balan, & Rosca, 2001). Deep neural networks (DNNs) have also been used to learn binary masks (Wang & Wang, 2013; Zhao, Wang, & Wang, 2014) and soft masks (Narayanan & Wang, 2013, 2014; Huang, Kim, Hasegawa-Johnson, & Smaragdis, 2014). A major disadvantage of using IBM is the loss of information in regions where the source and masker overlap. Model-based approaches such as NMF that factorize a nonnegative matrix into (usually) two nonnegative matrices have been used extensively for monaural source separation and over a wide range of sources, including human speech, noise, and music (Schmidt, Larsen, & Hsiao, 2007).

This letter proposes various linear methods for separating (artificial) mixtures of two sources. The approaches are based on the power spectrogram, a time-frequency representation of sound that has the advantage of being phase insensitive and so allows speakers to be distinguished based on differences in the spectro-temporal sound patterns they produce. This work introduces new methods based on eigenanalysis, probabilistic demixing, and linear regression. These methods are evaluated by estimating (reconstructing) target spectrograms of single speakers from mixture speech. The reconstructed spectrograms are compared to those yielded by nonlinear approaches, that is, nonnegative sparse coding (NNSC) (Schmidt et al., 2007) and a supervised version thereof (which we refer to as non-blind NNSC, described in section 3.5).

2 Methods

All computations are performed using Matlab R2011a on a 64-bit machine with 8 GB of RAM and an Intel Core i7 processor with 4 cores and clock frequency 2.93 GHz.

Following Berouti, Schwartz, and Makhoul (1979) the signal $s(t)$ in the time-frequency domain is represented as an element-wise exponentiated short-time Fourier transform (STFT):

$$S = |STFT\{s(t)\}|^{\gamma}. \tag{2.1}$$

The STFT is computed using a Fourier window size of n samples and 75% overlap between successive Hanning windows. The algorithms are applied on m adjacent columns of the spectrogram, concatenated into a single column vector. Each of the spectrogram columns contains $n/2$ distinct frequency bands. Therefore, the dimensionality of the data processed by the algorithms is given by ($N = mn/2$). The exponent γ is referred as the sparseness factor; it defines the sparseness of the sound representation (the sparseness of S increases with γ). The performances of the algorithms are evaluated for various n, m, and γ. Note that for $\gamma = 2$ in equation 2.1, S is referred to as the power spectral density (PSD).

2.1 The Problem and the Approach. Speaker separation is evaluated on audio speech data using the GRID corpus and Mocha-TIMIT—Texas Instruments (TI) and Massachusetts Institute of Technology (MIT)—database. All sentences are sampled at 16 KHz. The training data consist of 20 sets of audio files, each set containing roughly 5 minutes of clean speech from a distinct speaker (10 male and 10 female speakers). In the training phase, source-specific filters are learned from the training set. The testing data consist of roughly 1 minute of clean speech from each speaker. On each run of testing, one male and one female speaker are randomly chosen and are denoted by

x and y. The same set of speakers is used for training and testing. The test audio signals (mixture speech) $s_z(t)$ are artificially created by summing the two underlying speech signals $s_x(t)$ and $s_y(t) : s_z(t) = s_x(t) + s_y(t)$. Note that since the STFT is a linear function, the mixture $s_z(t) = s_x(t) + s_y(t)$ can be expressed in the time-frequency domain as $Ze^{\varphi_{tz}} = Xe^{\varphi_{tx}} + Ye^{\varphi_{ty}}$, where Z, X, and Y are the absolute values of the FFT for the mixture, speaker x, and speaker y, respectively, and φ_t is the corresponding phase. Since the inequality $|X - Y| \leq Z \leq X + Y$ holds true for any X, Y, and Z, therefore for a set of uncorrelated signals X and Y, $E\left(Z^2\right) \approx X^2 + Y^2$ where $E\left(.\right)$ is the expected value.

In this letter, the approximation $Z^\gamma = X^\gamma + Y^\gamma$ is used, which can be thought of as a generalization of the binary mask approach because the approximation is excellent ($X^\gamma + Y^\gamma$ is close to Z^γ) for any γ when $X \gg Y$ or $X \ll Y$. In general, the assumption is good for γ close to 2 (Berouti et al., 1979). The assumption becomes worse as $\gamma \to 0$, and thus the performances of the various methods are expected to be lower in this range.

The spectrograms of speech for the two speakers in the training set are denoted $\tilde{X}_t$ and $\tilde{Y}_t$. From the training set, the filters (linear methods) or libraries (NNSC) are derived. These filters and libraries are used to estimate each speaker's spectrogram $\hat{X}_t$ and $\hat{Y}_t$ from mixture spectrograms $\boldsymbol{Z}_t$.

From $\hat{X}_t$, an estimate $\hat{s}_x(t)$of the underlying single-speaker sound waveform is also derived by inverting the spectrogram $\hat{X}_t$ as follows:

$$\hat{s}_x(t) = Re(iFFT(\exp(i\varphi_t)\hat{X}_t)), \tag{2.2}$$

where $iFFT$ represents the inverse Fourier transform, $Re\left(.\right)$ represents the real value, and the phase signal φ_t is obtained from applying the STFT to the mixture signal $s_z(t)$. To obtain the final waveform from the set of overlapping estimates $\hat{s}_x$, each $\hat{s}_x$ is multiplied by a Hanning window, and their weighted mean (weighted by the amount of overlap) is computed.

Because speech signals typically show strong temporal correlations that extend beyond typical STFT windows, a reasonable extension is to consider more than individual spectrogram columns. To that end, the window vector $\tilde{\boldsymbol{X}}_t$ is defined to encompass multiple spectrogram columns,

$$\tilde{\boldsymbol{X}}_t = \tilde{X}_{t-L}, \tilde{X}_{t-L+1}, \ldots, \tilde{X}_t, \tilde{X}_{t+1}, \ldots, \tilde{X}_{t+R}$$

for $L \geq 0$ and $R \geq 0$ (depicted in Figure 1) and likewise for $\tilde{\boldsymbol{Y}}_t$ and $\boldsymbol{Z}_t$.

Note that for the subsequent methods, one spectrogram window $\hat{\boldsymbol{X}}_t$ is predicted at each time-step t and averaged over overlapping windows to obtain $\hat{X}_t$, which is ultimately inverted into cleaned speech. The exception will be suppression-regression, where single columns $\hat{X}_t$ and $\hat{Y}_t$ are directly estimated from windows $\tilde{\boldsymbol{X}}_t$ and $\boldsymbol{Z}_t$. Note also that source separation

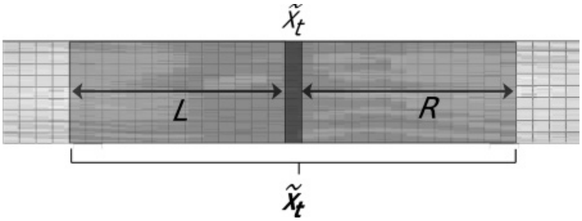


Figure 1: A single spectrogram column $\tilde{X}_t$ and the window $\tilde{\boldsymbol{X}}_t$ comprising L columns on the left and R columns on the right.

methods based on $\hat{\boldsymbol{X}}_t$ are feasible for real-time implementation only when R is small, ideally $R = 0$.

2.2 Mean Subtraction (Applicable to the Linear Methods). All methods make the simplifying assumption that $Z_t = X_t + Y_t$, that is, the mixture spectrogram is the sum of the two individual speaker spectrograms. From this assumption, the following mean subtraction method is derived. The means $\bar{X} = \langle \tilde{X}_t \rangle_t$ and $\bar{Y} = \langle \tilde{Y}_t \rangle_t$ are defined over individual training spectrograms, where $\langle . \rangle_t$ represents the mean over time. Equally the means $\bar{\boldsymbol{X}} = \langle \tilde{\boldsymbol{X}}_t \rangle_t$ and $\bar{\boldsymbol{Y}} = \langle \tilde{\boldsymbol{Y}}_t \rangle_t$ are defined over windows of training spectrograms. As the training sets for both speakers are chosen to be equally large, the overall mean $\bar{M}$ of the training spectrogram columns is given by $\bar{\boldsymbol{M}} = \left(\bar{X} + \bar{Y}\right)/2$ and likewise $\bar{\boldsymbol{M}} = (\bar{X} + \bar{Y})/2$ for the mean spectrogram window. The linear methods to be defined extract demixing filters W_x and W_y from the mean-subtracted training spectrograms $\tilde{X}_t - \bar{X}$ and $\tilde{Y}_t - \bar{Y}$. The filter W_x maps components of X_t onto themselves and components of $Y_t$onto zero, and vice versa for filter W_y.

For demixing, the steps are to subtract from the mixture spectrogram Z_t twice the overall mean $\bar{M}$, $Z_t - 2\bar{M}$, apply the demixing filters and add the individual training spectrogram means $\bar{X}$ and $\bar{Y}$ to the result to obtain the estimates $\hat{X}_t$ and $\hat{Y}_t$, respectively. It is simple to show that by doing so (and assuming $Z_t = X_t + Y_t$), the following $\hat{X}_t + \hat{Y}_t = Z_t$ is true: adding the estimates $\hat{X}_t = W_x\left(Z_t - 2\bar{M}\right) + \bar{X}$ and $\hat{Y}_t = W_y\left(Z_t - 2\bar{M}\right) + \bar{Y}$ yields $\hat{X}_t + \hat{Y}_t = W_x\left(X_t + Y_t - \bar{X} - \bar{Y}\right) + \bar{X} + W_y\left(X_t + Y_t - \bar{X} - \bar{Y}\right) + \bar{Y} = X_t + Y_t$ under the assumption $W_x + W_y = I$.

The following mean-subtraction methods are also evaluated, but their performance is worse (larger residuals and lower signal-to-noise ratios; data not shown) compared to the above chosen method for mean subtraction:

1. During training, the speaker-specific filters on the individual-mean subtracted training spectrograms $\tilde{X}_t - \bar{X}$ and $\tilde{Y}_t - \bar{Y}$ are computed. During testing, the mixture-mean subtracted mixture spectrogram

$\tilde{Z}_t - \bar{Z}$ is projected on these filters, and subsequently half of the mixture mean $\bar{Z}/2$ is added back to the projections to obtain the estimates $\hat{X}_t$ and $\hat{Y}_t$.

2. The speaker-specific filters are computed on a set of overall-mean $\bar{M}$ subtracted training spectrograms: $\tilde{X}_t - \bar{M}$ and $\tilde{Y}_t - \bar{M}$. During testing, $\hat{X}_t$ and $\hat{Y}_t$ are estimated by projecting the overall-mean $\bar{M}$ subtracted mixture spectrogram, $Z_t - \bar{M}$, on these filters; subsequently the individual training spectrogram mean $\bar{X}$ (resp. $\bar{Y}$) is added to the projections to obtain the estimates $\hat{X}_t$ and $\hat{Y}_t$.
3. The speaker-specific filters are computed on a set of overall-mean $\bar{M}$ subtracted training spectrograms, $\tilde{X}_t - \bar{M}$ and $\tilde{Y}_t - \bar{M}$, as above. However during the testing phase, the $2\bar{M}$ subtracted mixture spectrogram is projected on these filters; subsequently, the individual training spectrogram mean $\bar{X}$ (resp. $\bar{Y}$) is added to the projections to obtain the estimates $\hat{X}_t$ and $\hat{Y}_t$.

Note that methods 1 and 3 also satisfy the assumption $Z_t = X_t + Y_t$ but method 2 does not.

3 Algorithms

The three linear methods for source separation are introduced followed by two variants of a nonlinear method.

3.1 Eigenmode Analysis of Covariance Difference (EACD). Eigenmode analysis is applied in order to compute filters that span directions associated with large variance for one speaker and small variance for the other speaker, an approach inspired by Machens, Romo, and Brody (2010). The speech mixture spectrograms are then projected onto these filters to selectively suppress one of the two speakers present in the mixture.

Concretely, the covariance matrices for the two speakers are denoted by

$$C_X = \langle \tilde{X}_t \tilde{X}_t^T \rangle_t - \bar{X}\bar{X}^T \text{ and } C_Y = \langle \tilde{Y}_t \tilde{Y}_t^T \rangle_t - \bar{Y}\bar{Y}^T. \tag{3.1}$$

The difference of covariance matrix is further defined as

$$D = C_X - C_Y.$$

Let V be the matrix of eigenvectors that diagonalizes D :

$$E = V^{-1}DV, \tag{3.2}$$

where E is the diagonal matrix consisting of eigenvalues of D. Since C_X and C_Y are symmetric matrices, the matrix D is also symmetric. As a result, V is an orthonormal matrix, and therefore $V^{-1} = V^T$. By substituting the definitions of C_X and C_Y into equation 3.2, the following is obtained:

$$E = V^T \langle (\tilde{X}_t - \bar{X})(\tilde{X}_t - \bar{X})^T \rangle V - V^T \langle (\tilde{Y}_t - \bar{Y})(\tilde{Y}_t - \bar{Y})^T \rangle V.$$

Because the average is a linear operator,

$$\begin{aligned} E &= \langle V^T (\tilde{X}_t - \bar{X})(\tilde{X}_t - \bar{X})^T V \rangle - \langle V^T (\tilde{Y}_t - \bar{Y})(\tilde{Y}_t - \bar{Y})^T V \rangle \\ &= \langle A_t A_t^T \rangle - \langle B_t B_t^T \rangle, \end{aligned} \tag{3.3}$$

with $A_t = V^T(\tilde{X}_t - \bar{X})$ and $B_t = V^T(\tilde{Y}_t - \bar{Y})$. Thus, the diagonal elements of E give the difference in variance between the two mean-subtracted data sets along the eigenvectors (columns of V). These eigenvectors span two subspaces. In the first subspace, speaker x produces a higher variance than speaker y and vice versa. The sets of filters that span the first and the second subspaces are labeled F_x and F_y.

Once these filters are computed, the sources present in a mixture can then be estimated simply by projecting the mean subtracted mixture spectrogram $Z_t - 2\bar{M}$ onto the filters and subsequently adding back the individual means of the two speakers to obtain the estimated spectrograms for the two speakers:

$$\hat{X}_t = F_x F_x^T (Z_t - 2\bar{M}) + \bar{X} \tag{3.4}$$

$$\hat{Y}_t = F_y F_y^T (Z_t - 2\bar{M}) + \bar{Y}. \tag{3.5}$$

The vectors $\hat{X}_t$ and $\hat{Y}_t$ as defined in equations 3.4 and 3.5 satisfy the assumption $Z_t = \hat{X}_t + \hat{Y}_t$ (because $F_x F_x^T + F_y F_y^T = I$ is the identity matrix, here $F_x F_x^T = W_x$ and $F_y F_y^T = W_y$ as mentioned in section 2). The algorithm works identically for windows of multiple spectral columns, $m > 1$, as described in section 2, by simply replacing the variables by their boldface counterparts and averaging $\hat{\boldsymbol{X}}_t$ and $\hat{\boldsymbol{Y}}_t$ over their overlapping regions.

3.2 Probabilistic Approach (Maximum Likelihood Demixing). This approach performs source separation under the assumption that the two (mean-subtracted) sources are independent and gaussian distributed.

Let C_X and C_Y again be the covariance matrices of the two sources that are assumed to be gaussian distributed:

$$P(X_t = x) = p(x) = K_x \exp(-0.5(x - \bar{X})^T C_X^{-1} (x - \bar{X})) \tag{3.6}$$

and

$$P(Y_t = y) = p(y) = K_y \exp(-0.5(y - \bar{Y})^T C_Y^{-1} (y - \bar{Y})), \tag{3.7}$$

with K_x and K_y normalization constants.

With the assumption of independence of the two sources, the joint probability $P(X_t = x, Y_t = y)$ can be written as

$$\begin{aligned} P(X_t = x, Y_t = y) &= p(x)p(y) \\ &= K_x K_y \exp(-0.5((x - \bar{X})^T C_X^{-1}(x - \bar{X}) + (y - \bar{Y})^T C_Y^{-1}(y - \bar{Y}))). \end{aligned}$$

The underlying sources $\hat{X}_t$ and $\hat{Y}_t$ are estimated as the most likely constituents of the mixture. The estimate $\hat{X}_t$ is derived by setting the derivative of the joint probability (given the mixture Z_t) to zero:

$$\begin{aligned} 0 &= \frac{\partial}{\partial x}(\log(P(x, y|Z_t))|_{x=\hat{X}_t} \\ &= \frac{\partial}{\partial x}[(x - \bar{X})^T C_X^{-1}(x - \bar{X}) + (Z_t - \bar{Y} - x)^T C_Y^{-1}(Z_t - \bar{Y} - x)]|_{x=\hat{X}_t}, \end{aligned}$$

where in the last equation, the assumption $Z_t = x + y$ is used. Isolating $\hat{X}_t$ yields

$$\hat{X}_t = C_X(C_X + C_Y)^{-1}(Z_t - 2\bar{M}) + \bar{X}, \tag{3.8}$$

and

$$\hat{Y}_t = C_Y(C_X + C_Y)^{-1}(Z_t - 2\bar{M}) + \bar{Y}. \tag{3.9}$$

(The detailed matrix operations are described in Petersen & Pedersen, 2008.) Because the matrices $C_X(C_X + C_Y)^{-1}$ and $C_Y(C_X + C_Y)^{-1}$ have to be computed only once, this method of demixing is extremely efficient. Note that for evaluating the performance of the algorithm for multiple spectral columns ($m > 1$), the procedure is exactly as in EACD for multiple columns. (See section 3.1.)

3.3 Suppression-Regression. A simple approach to source separation is obtained through linear regression. The idea is to compute a speaker-specific filter L_x that maps the training spectrogram $\tilde{Y}_t$ onto zero (suppression) and $\tilde{X}_t$ onto itself (regression):

$$\tilde{X}_t - \bar{X} = L_x(\tilde{X}_t - \bar{X}) + \varepsilon_t^x, \tag{3.10}$$

$$0 = L_x(\tilde{Y}_t - \bar{Y}) + \varepsilon_t^y \tag{3.11}$$

where ε_t^x and ε_t^y are sources of uncorrelated gaussian noise. The linear map L_x that minimizes the sum of squared errors, $\sum_{t=1}^{T}(\varepsilon_t^{x^2} + \varepsilon_t^{y^2})$, is computed, where T denotes the common length of the training spectrograms.

For the second source, the linear map L_y is obtained by minimizing $\sum_{t=1}^{T}(\delta_t^{x^2} + \delta_t^{y^2})$ in the analogous system:

$$\tilde{Y}_t - \bar{Y} = L_y(\tilde{Y}_t - \bar{Y}) + \delta_t^x, \tag{3.12}$$

$$0 = L_y(\tilde{X}_t - \bar{X}) + \delta_t^y. \tag{3.13}$$

The solutions of the above equations are

$$L_x = C_X(C_X + C_Y)^{-1}, \tag{3.14}$$

$$L_y = C_Y(C_X + C_Y)^{-1}, \tag{3.15}$$

where C_X and C_Y are the covariance matrices of the respective speakers. Based on the maps L_x and L_y, the two sources from the mixture Z_t are estimated as

$$\hat{X}_t = L_x(Z_t - 2\bar{M}) + \bar{X}, \tag{3.16}$$

$$\hat{Y}_t = L_y(Z_t - 2\bar{M}) + \bar{Y}. \tag{3.17}$$

These source estimations are identical to equations 3.8 and 3.9 of MLD. Thus, in the one-column case, suppression-regression is identical to maximum likelihood demixing.

As mentioned earlier, correlations over longer durations are also considered and are modeled by demixing filters that span over several spectrogram columns. Therefore, analogous to equations 3.10 to 3.13, $\boldsymbol{L}_x$ and $\boldsymbol{L}_y$ are chosen such that they minimize the sum of squared error in the equation systems

$$\tilde{X}_t - \bar{X} = \boldsymbol{L}_x(\tilde{\boldsymbol{X}}_t - \bar{\boldsymbol{X}}) + \varepsilon_t^x, \tag{3.18}$$

$$0 = \boldsymbol{L}_x(\tilde{\boldsymbol{Y}}_t - \bar{\boldsymbol{Y}}) + \varepsilon_t^y, \tag{3.19}$$

and

$$\tilde{Y}_t - \bar{Y} = \boldsymbol{L}_y(\tilde{\boldsymbol{Y}}_t - \bar{\boldsymbol{Y}}) + \delta_t^y, \tag{3.20}$$

$$0 = \boldsymbol{L}_y(\tilde{\boldsymbol{X}}_t - \bar{\boldsymbol{X}}) + \delta_t^x, \tag{3.21}$$

where $\bar{\boldsymbol{X}}$ and $\bar{\boldsymbol{Y}}$ are the individual speaker means for the larger predictor window. From these equations, the following are derived:

$$\boldsymbol{L}_x = C_{\boldsymbol{X}X}(\boldsymbol{C}_X + \boldsymbol{C}_Y)^{-1}, \tag{3.22}$$

$$\boldsymbol{L}_y = C_{\boldsymbol{Y}Y}(\boldsymbol{C}_X + \boldsymbol{C}_Y)^{-1}, \tag{3.23}$$

where $\boldsymbol{C}_X$ and C_Y are the covariance matrices of the windows $\tilde{\boldsymbol{X}}_t$ and $\tilde{\boldsymbol{Y}}_t$ respectively, and C_{XX} and C_{YY} are the covariance matrices between single columns $\tilde{X}_t$ and their corresponding windows $\tilde{\boldsymbol{X}}_t$ and between the columns $\tilde{Y}_t$ and their corresponding windows $\tilde{\boldsymbol{Y}}_t$.

Having determined the linear maps $\boldsymbol{L}_x$ and $\boldsymbol{L}_y$ on the training data, the mixture is separated according to

$$\hat{X}_t = \boldsymbol{L}_x(\boldsymbol{Z}_t - 2\bar{\boldsymbol{M}}) + \bar{X}, \tag{3.24}$$

$$\hat{Y}_t = \boldsymbol{L}_y(\boldsymbol{Z}_t - 2\bar{\boldsymbol{M}}) + \bar{Y}. \tag{3.25}$$

Because suppression-regression is identical to MLD for single columns, the results for suppression-regression are not shown in Figures 3 and 5. In the multiple spectral column case ($m > 1$), unlike in MLD where demixing filters are learned from the past, in suppression-regression, the demixing filters are learned based on both the past and the future ($R > 0$), as shown in Figure 4.

Note that a large condition number of the matrix (to be decomposed) can become a significant problem leading to ill-conditioned matrices. The problem of ill-conditioned matrices is observed only for suppression-regression in case of multiple spectral columns ($m > 1$). One of the methods of dealing with such ill-conditioned matrices is regularization. In this work, Tikhonov regularization (Tikhonov & Arsenin, 1977) is used. For an ill-posed inverse problem (because of either the nonexistence or nonuniqueness of $\boldsymbol{r}$) such as $A\boldsymbol{r} \approx B$, the estimate $\hat{\boldsymbol{r}}$ of r is usually computed using an ordinary least squares approach that minimizes the residual $\|A\boldsymbol{r} - B\|^2$ where $\|.\|$ represents the Euclidean norm. This may occur because of A being illconditioned or singular. To obtain a solution with desirable properties, a regularization term is included in this minimization $\|A\boldsymbol{r} - B\|^2 + \|\Gamma\boldsymbol{r}\|^2$ where $\Gamma = kI$ is called the Tikhonov matrix, k is a constant, and I is the identity matrix. An explicit solution is $\hat{\boldsymbol{r}} = (A^TA + \lambda I)^{-1}A^TB$ for some regularization constant $\lambda = k^2$. In this work, the regularization constant λ is chosen such that it maximizes the performance of separation (higher SNR, PESQ score, and lower residual). It is varied from 0 to 10^7 (with step sizes initially increasing in multiples of 10 and fixed step size of 10^5 close to peak performance, which typically occurred between $\lambda = 10^6$ and $\lambda = 10^7$). Peak performance value was determined by cross-validation on an independent set of data. Figures 4 and 6b show the results for suppression-regression after performing regularization. The performance improvement is largest for a large number of columns, which is expected because the problem (of inversion) is more ill posed the larger the number of columns. On average, the SNR values improve by 0.7 dB, the PESQ values by 0.3, and residual values decrease by 0.05 after regularization.

3.4 Source Separation Using Nonnegative Sparse Coding (NNSC). These linear approaches are compared to nonlinear approaches based on NMF, which encompasses a set of methods in which a matrix S is factorized into two nonnegative matrices D and H such that $S = DH$ (all components of D and H are nonnegative). Nonnegative matrix factorization leads to a parts-based representation because the factorization allows only additive, not subtractive, combinations (Eggert & Korner, 2004).

Training spectrograms $\tilde{X}_t = DH_t$ are factorized such that the matrix D is a dictionary matrix whose columns contain a set of source-specific basis vectors and H_t is the code matrix that contains nonnegative weights that determine the linear combination of basis used to approximate $\tilde{X}_t$. Inspired by Schmidt et al. (2007) in which NMF was applied to source separation, a variant of sparse NMF termed nonnegative sparse coding (NNSC) is applied in which only a few filters are used to represent the data, because most of the filter coefficients are constrained to take close to zero values. In NNSC, sparseness of H_t is enforced by minimizing the following cost function,

$$F(D, H_t) = \frac{1}{2}\|\tilde{X}_t - DH_t\|_2^2 + \lambda\|H_t\|_1, \tag{3.26}$$

where $\|.\|_2$ represents the 2-norm and $\|.\|_1$ represents the 1-norm (Manhattan norm), and λ is a sparsity parameter.

Sparsity of basis coefficients H_t is enforced by the second term in equation 3.26. This term guarantees that only a small subset of dictionary elements is used at any time, thus forcing the dictionary elements to be source specific. Note that NNSC is a nonlinear approach because solving the global optimum of equation 3.26 is not tractable (NP-hard).

There exist diverse algorithms for computing this factorization (Eggert & Korner, 2004; Hoyer, 2002; Lee & Seung, 1999, 2006; Lin, 2007; Schmidt et al., 2007) including the multiplicative update rule from Lee and Seung (1999, 2006). This update rule has been very popular due to the simplicity of its implementation. An advantage of multiplicative update rules over standard gradient descent update is that convergence is guaranteed because no step-size parameter is needed (Lee & Seung, 1999). Also the relaxation toward a local minimum is fast and computationally inexpensive (Lee & Seung, 1999). This work uses the multiplicative update rules of Schmidt et al. (2007), which are inspired by Eggert and Korner (2004) and Lee and Seung (1999, 2006). These update rules provide an extra advantage in that they allow the target dictionaries to be learned from the mixture and thus constitute a semisupervised method of learning.

The masker dictionary is first computed using the clean speech of the masker as follows (speaker x being the masker and speaker y the target in this particular case):

1. Start with a randomly initialized dictionary matrix, D_x and code matrix H_x.
2. Alternate the following updates until convergence:

$$H_x \leftarrow H_x \bullet \frac{\bar{D}_x^T \tilde{X}}{\bar{D}_x^T \bar{D}_x H_x + \lambda}, \tag{3.27}$$

$$D_x \leftarrow \bar{D}_x \bullet \frac{\tilde{X} H_x^T + \bar{D}_x \bullet (1(\bar{D}_x H_x H_x^T \bullet \bar{D}_x))}{\bar{D}_x H_x H_x^T + \bar{D}_x \bullet (1(\tilde{X} H_x^T \bullet \bar{D}_x))}, \tag{3.28}$$

   where $\bullet$ represents pointwise multiplication and the horizontal line pointwise division, λ defines the sparsity parameter of single-speaker dictionaries, H_x is the full code matrix, and $\bar{D}_x$ is the column-wise normalized dictionary matrix for speaker x.
3. Under the assumption of additive mixing of sources, the following is obtained:

$$Z_t \approx X_t + Y_t = [D_y D_x] \begin{bmatrix} H_t^y \\ H_t^x \end{bmatrix} = DH_t, \tag{3.29}$$

   where D_x and D_y are the dictionary matrices and H_t^x and H_t^y are single columns of the code matrices for speakers x and y, respectively. D and H_t are the concatenated dictionaries and code matrices.

The target dictionaries are then learned directly from the mixture Z via the following iterative update rules until convergence:

$$H_y \leftarrow H_y \bullet \frac{\bar{D}_y^T Z}{\bar{D}_y^T \bar{D} H + l_y}, \tag{3.30}$$

$$H_x \leftarrow H_x \bullet \frac{\bar{D}_x^T Z}{\bar{D}_x^T \bar{D} H + l_x}, \tag{3.31}$$

$$D_y \leftarrow \bar{D}_y \bullet \frac{Z H_y^T + \bar{D}_y \bullet (1(\bar{D} H H_y^T \bullet \bar{D}_y))}{\bar{D} H H_y^T + \bar{D}_y \bullet (1(Z H_y^T \bullet \bar{D}_y))}, \tag{3.32}$$

where l_x and l_y are the sparsity parameters for the code matrices of speakers x and y, respectively. That is, when the dictionaries are trained on clean speech, the sparsity parameter λ is applied, and when the dictionary is applied on the mixture, the sparsity parameters l_x and l_y are applied.

Once D_y and H_y are computed, they are used to estimate the target spectrogram $\hat{Y}_t$ as follows:

$$\hat{Y}_t = D_y H_t^y, \tag{3.33}$$

where H_t^y are the single columns of the code matrix of speaker y.

The above approach of semiblind NNSC is also modified by first learning the target dictionary D_y from the clean data set $\tilde{Y}_t$ and then learning the masker dictionary from the mixture using update rules analogous to equations 3.30 to 3.32. Note that learning the dictionary of the masker from its clean speech is very useful when the target model is unavailable. If the masker model is unavailable, one can learn the masker dictionary from the mixture while learning the target from its clean speech. We show results for both of these semiblind approaches.

3.5 Non-Blind NNSC. One might expect a better performance when the dictionaries for both speakers are prelearned from clean training sentences as was done in the linear methods (in the NNSC approach described in section 3.4, the dictionary of one of the speakers from clean speech and the dictionary for the other speaker were computed using the mixture). In this section, NNSC is applied in learning both dictionaries D_x and D_y from clean training sentences; the code matrices H_t^x and H_t^y are then inferred from update rules, equations 3.30 and 3.31.

The individual sources in this non-blind NNSC method are then estimated as

$$\hat{X}_t = D_x H_t^x \tag{3.34}$$

and

$$\hat{Y}_t = D_y H_t^y. \tag{3.35}$$

Parameter values for non-blind and semi-blind NNSC were chosen by maximizing SNR as follows. The optimal parameter values in Schmidt et al. (2007) were taken as starting points ($\gamma = 0.6$, $\lambda = 0.2$, $l_x = 0.05$, $l_y = 0$, $N_x = 64$, $N_y = 64$) and varied one after the other over an empirically chosen range while keeping the others at their optimal values. The parameters l_x and l_y were varied over the range [0, 0.1] in linear steps of 0.01, λ over the range [0,1] in linear steps of 0.1, and N_x and N_y over the range [1,512] in powers of 2. The other parameters were set to $n = 1024 \mathrel{\widehat{=}} 64$ ms, $m = 1$. The following optimal values for the NNSC parameters were obtained:

- Sparsity parameters, $l_x = 0$ and $l_y = 0.05$ in equations 3.30 and 3.31
- Number of components of the target and masker dictionaries, $N_x = N_y = 64$
- Sparsity parameter, $\lambda = 0.2$ used in equation 3.27 for learning single-speaker dictionaries

4 Performance Criteria

The performances of the algorithms are evaluated using the following measures:

1. Normalized residual $\Re$ between estimated and target spectrograms,

$$\Re = \sum_t \frac{\|X_t - \hat{X}_t\|_2^2}{\|X_t\|_2^2}, \tag{4.1}$$

where $\|.\|_2^2$ represents the square of the 2 norm
2. SNR computed over the target audio signal $s_x(t)$ and the predicted audio signal $\hat{s}_x(t)$,

$$SNR_s(dB) = 10\, log_{10} \frac{\sum_t (\hat{s}_x(t))^2}{\sum_t (s_x(t) - \hat{s}_x(t))^2} \tag{4.2}$$

3. Perceptual evaluation of speech quality (PESQ) scores (Hu & Loizou, 2008) computed over the target audio signal $s_x(t)$ and the predicted audio signal $\hat{s}_x(t)$

5 Simulation Results

After training the filters and dictionaries, artificial speech mixtures from two speakers (male and female), in Figure 2a, were created, and the corresponding mixture spectrogram in Figure 2b was computed. From this mixture spectrogram, the speech spectrogram of the target speaker (male) was estimated using the algorithms already presented.

The example in Figure 2 shows all of speech from only the target, only the masker, and a superposition of both. Masker suppression is strongest in the case of NNSC (masker learned). However, the target itself is also highly suppressed in this case. As a result, the target speech suffers from low intelligibility. By contrast, masker suppression is weaker in the linear approaches and NNSC (target learned and both learned), but the target speech is better preserved.

Shown in Figure 2c is an excerpt of the waveforms comparing the target, the mixture, and the estimated target using MLD/suppression-regression (the best method). The estimated waveform matches well with the target waveform.

The performance of the various algorithms is compared for different values of the STFT window size n, the STFT exponent γ, and the number m of spectrogram columns (see Figures 3 to 5). The base parameter set is $n = 1024 \,\widehat{=}\, 64$ ms, $m = 1$, $\gamma = 1$ and the values for one of the three parameters are varied while keeping the other two parameters fixed. In all figures, the plots of residual spectrograms and SNRs that were averaged over all speakers are shown. The performances of the algorithms are further compared using the widely used and International Telecommunication Union–Telecommunication (ITU-T) recommended perceptual PESQ metric. The PESQ scores support the results shown in Figures 3 to 5 and align well to them (see Figure 6).

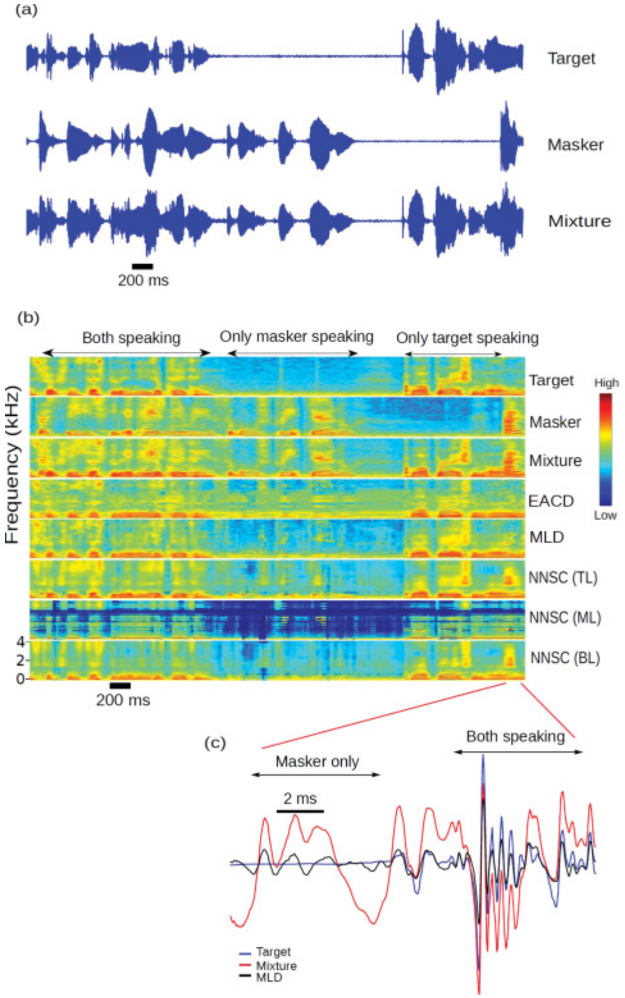


Figure 2: (a) An excerpt of the sound waveforms from the target (male) speaker, the masker (female), and their mixture. (b) From top to bottom: target spectrogram X_t, masker spectrogram Y_t, mixture spectrogram Z_t and estimated target spectrograms $\hat{X}_t$ for the male speaker using EACD, MLD/suppression-regression, NNSC (TL-target learned), NNSC (ML-masker learned), and NNSC (BL-both learned), from top to bottom. Log spectrograms are shown. (c) Waveform excerpts of the target (blue), the mixture (red), and the estimated target by MLD (black). $\gamma = 1$, FFT window size $n = 1024 \widehat{=} 64$ ms, number of spectral columns $m = 1$. The target is well preserved, and the masker is well suppressed in both the voiced and unvoiced regions for the linear methods depicting the strength of these methods. Note that the results for MLD and suppression-regression are exactly the same; hence only the MLD results are shown in the figure.

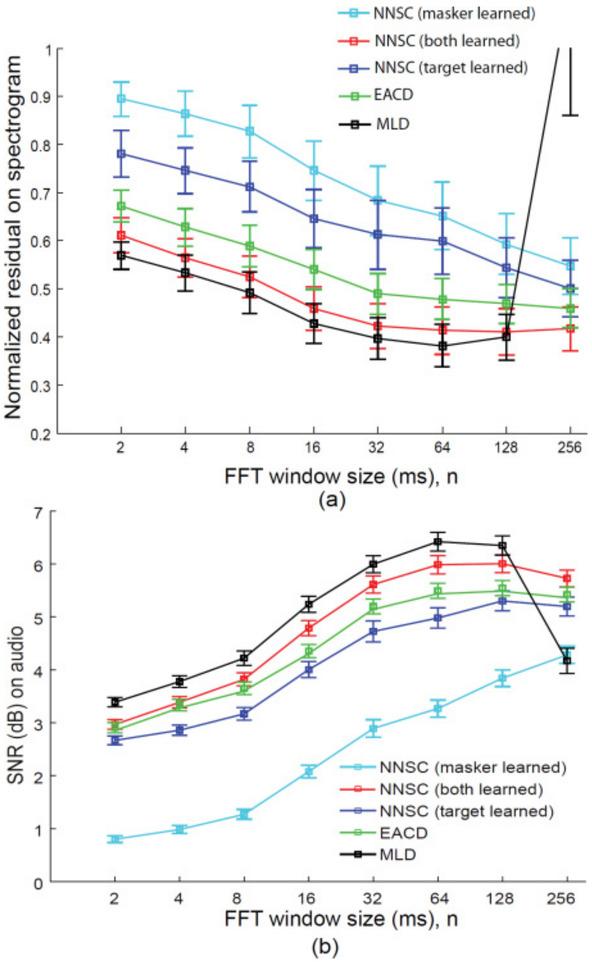
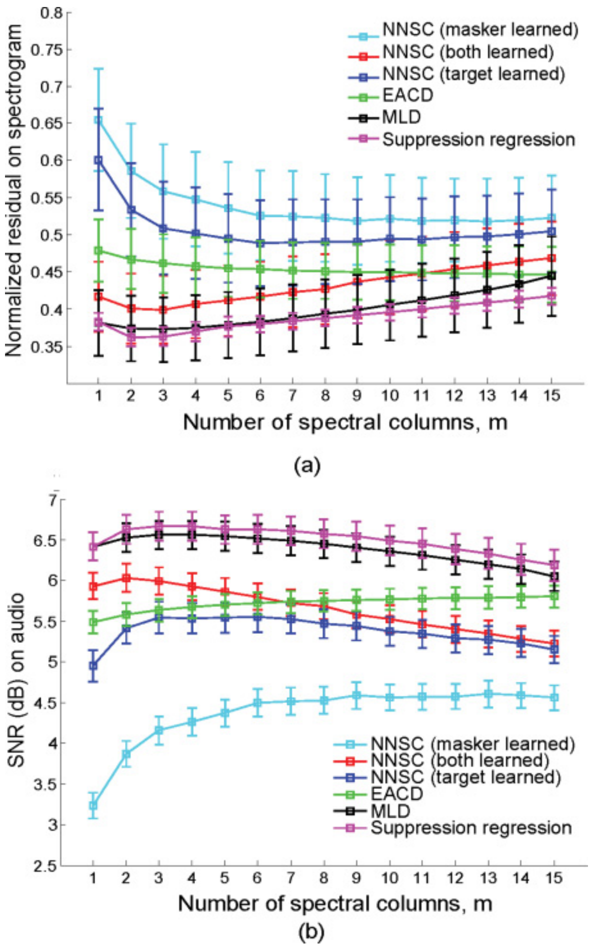


Figure 3: (a) Normalized residual of estimated target spectrogram and (b) SNR of estimated target waveform versus STFT window size, comparing all the algorithms. Plots show averages over all speakers. The optimal performance was at 64 ms for MLD/suppression-regression and NNSC (both learned), 128 ms for EACD and NNSC (target learned), and 256 ms for NNSC (masker learned). $\gamma = 1$, $m = 1$. Note that the results for MLD and suppression-regression are exactly the same; hence, only one of them is shown in this figure.

Figure 3 shows the plots depicting the performances of the algorithms under varying FFT window size n. For all algorithms except NNSC (masker learned), the SNR of the reconstructed signals peaks at either 64 ms or 128 ms. The normalized residual plots confirm this finding for the



(a)

(b)

Figure 4: (a) Normalized residual of estimated target spectrogram and (b) SNR of estimated target waveform versus number m of spectral columns, comparing all the algorithms (plots show averages over all speakers). $\gamma = 1$, $n = 1024 \mathrel{\widehat{=}} 64$ ms.

well-performing methods MLD/suppression-regression and NNSC (both dictionaries learned). For the remaining three methods, the minimal normalized residual is not reached even at the largest window size tested.

Figure 4 shows the performances of the algorithms for various numbers m of spectral columns. Adding more spectral columns captures the context-dependent information in each analysis vector, leading to better performance. However, temporal context beyond a certain limit (depending on the amount of data available) is not useful and leads to performance reduction. Note that research has shown that the temporal context in human speech can typically vary from 20 ms to 200 ms (Rosen, 1992).

The performance comparison of the different algorithms gives a sense of how prone they are to overfitting. The better-performing algorithms MLD, suppression-regression, and NNSC (both learned and target learned) exhibit their peak performances for few spectrogram columns ($n = 2 - 3$), while the performances of the other nonlinear algorithms saturate. EACD shows a monotonically increasing performance, revealing it is more robust to overfitting.

Shown in Figure 5 are the performances of the algorithms for various values of the STFT exponent γ. Overall MLD/suppression-regression performs best, followed by NNSC (both speakers learned). NNSC (masker learned) method performs worst.

The sparseness factor γ has a strong impact on performance. The assumption implicit in the linear methods is that the data are gaussian distributed. The nonlinear NNSC methods, however, assume exponentially distributed independent components present in the mixture (Hoyer, 2002). As $\gamma \to 0$, the distribution of the spectrogram pixels approaches a gaussian. In the midrange $\gamma \approx 1$ this distribution approaches an exponential, and for large γ, this distribution is heavy tailed. The approximation that the sources are additive in the spectrogram domain gets worse the further γ deviates from 2 (Berouti et al., 1979).

Given this trade-off, it may not be surprising that the optimal performance for EACD and MLD is not near small γ values but around $\gamma = 0.8$.

The same argument can be applied to NNSC (both dictionaries learned) for which the performance is optimal around $\gamma = 1$. In this range, MLD/suppression-regression and NNSC (both speakers learned) perform best. This can be seen in the estimated spectrogram for the case of $\gamma = 1$ (see Figure 3). The region where the masker speaker (female in this example) is solely speaking is suppressed best in NNSC (masker learned); however, NNSC also over-suppresses the target speaker, leading to lower SNR values compared to suppression-regression and NNSC (both dictionaries learned). Performances of EACD and MLD are still very good in this range.

5.1 Sparse NMF with Kullback-Leibler (KL) and Itakura-Saito Divergence Criteria. Apart from the Euclidian distance-based criterion for matrix factorization, we also tested other divergence criteria such as the Kullback-Leibler (KL) divergence and the Itakura-Saito (IS) divergence. These criteria are usually considered better suited for audio spectra than the Euclidean distance (Févotte & Idier, 2011). Using the KL divergence, the SNRs averaged over all speakers improved by 0.77 dB for NNSC (target learned), 1.18 dB for NNSC (masker learned), and 0.34 dB for non-blind NNSC. When the KL divergence was used, the residuals decreased on average by 0.09 for NNSC (target learned), 0.10 for NNSC (masker learned), and 0.01 for non-blind NNSC. The PESQ scores also improved on average

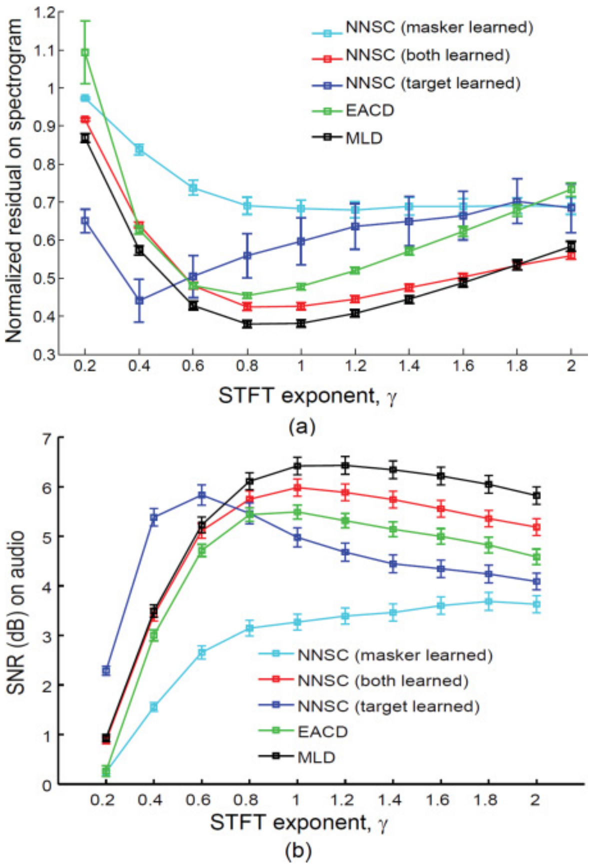
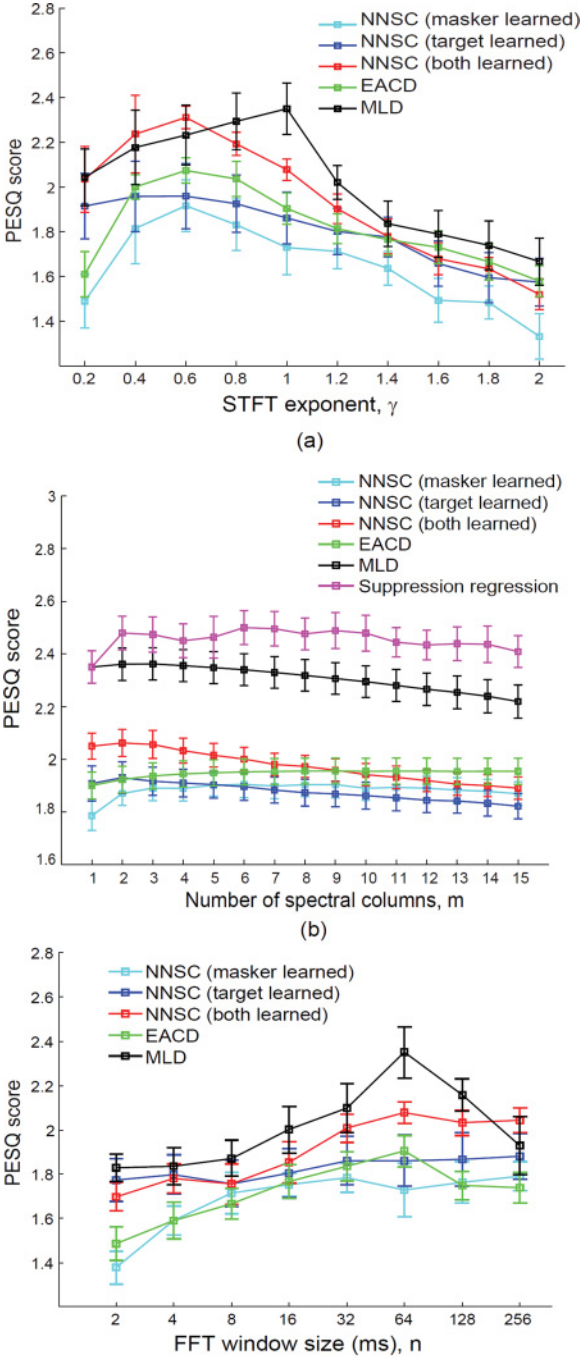


Figure 5: (a) Normalized residual of estimated target spectrogram and (b) SNR of estimated target waveform versus STFT exponent γ, comparing all the algorithms (plots show averages over all speakers). Results in panels a and b are well aligned with each other. Overall MLD/suppression-regression outperforms all methods. EACD outperforms the nonlinear NNSCs only when the masker is learned. As expected, the performance of NNSC (both learned) is better than NNSC (target learned) followed by the masker-learned NNSC variant. STFT window size $n = 1024 \mathrel{\widehat{=}} 64$ ms; number of spectral columns $m = 1$.

by 0.06 for NNSC (target learned), 0.05 for NNSC (masker learned), and 0.06 for non-blind NNSC. The results are summarized in Table 1.

By contrast, the results using sparse IS divergence were slightly worse than those using the Euclidean distance. Compared to SNRs using the Euclidean distance, the SNRs decreased on average by 0.12 dB, 0.27 dB, and 0.18 dB, and residual values increased by 0.037, 0.04, and 0.04 for NNSC


NNSC (masker learned)
NNSC (target learned)
NNSC (both learned)
EACD
MLD
PESQ score
2.8
2.6
2.4
2.2
2
1.8
1.6
1.4
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
STFT exponent, γ
NNSC (masker learned)
NNSC (target learned)
NNSC (both learned)
EACD
MLD
Suppression regression
PESQ score
3
2.8
2.6
2.4
2.2
2
1.8
1.6
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
Number of spectral columns, m
NNSC (masker learned)
NNSC (target learned)
NNSC (both learned)
EACD
MLD
PESQ score
2.8
2.6
2.4
2.2
2
1.8
1.6
1.4
2
4
8
16
32
64
128
256
FFT window size (ms), n

(a)

(b)

(c)

(target learned), NNSC (masker learned), and nonblind NNSC, respectively. The PESQ scores for the IS divergence were reduced as well in comparison to the 2-norm distance-based divergence measure. The PESQ scores averaged over all speakers were reduced by 0.12, 0.02, and 0.05 for NNSC (target learned), NNSC (masker learned), and nonblind NNSC, respectively. The results comparing all the divergence criteria are summarized in Table 1.

5.2 Additional Layer of Wiener Filtering for Speech Enhancement. Wiener filtering is a widely used method in signal processing, particularly for signal denoising and source separation. Wiener filtering is typically applied to audio signals in the spectrogram domain (using the STFT). We use the adaptive Wiener filtering (AWF) approach that enforces a reconstruction constraint of mixture spectrograms being the sum of individual estimated spectrograms. Under this constraint, the new spectrogram estimates $\hat{X}_{AWF}$ and $\hat{Y}_{AWF}$ are given by $\hat{X}_{AWF} = \frac{(\hat{X})^2}{(\hat{X})^2+(\hat{Y})^2} \bullet Z$ and $\hat{Y}_{AWF} = \frac{(\hat{Y})^2}{(\hat{X})^2+(\hat{Y})^2} \bullet Z$, where the horizontal line represents pointwise division and $\bullet$ represents pointwise multiplication of two matrices. This reconstruction constraint improves the separation performance of all methods studied. The results are summarized in Table 2.

We also tried the consistent Wiener filtering (CWF) approach proposed by Roux and Vincent (2013) that enforces consistency between neighboring STFT coefficients, as follows. Under gaussian assumptions, the negative log likelihood of the conditional distribution of the source X given the mixture Z is given by $-\log P(X|Z) = \psi(X) = \sum_{t,f}(X_{tf} - \mu_{tf})^H \sigma_{nf}(X_{tf} - \mu_{tf})$, where $(.)_{tf}$ represents a time-frequency bin, μ_{tf} is the mean, and σ_{tf}^{-1} is the covariance of the conditional distribution $P(X|Z)$. To enforce consistency in X, a necessary and sufficient condition is that the $STFT$ of the $iSTFT$ is equal to itself or, in other words, that it belongs to the null space Ker $\mathcal{F}$ of the $\mathbb{R}$-linear operator $\mathcal{F}$ from X to itself defined by $\mathcal{F} = I - STFT \circ iSTFT$. The hard consistency constraint $\mathcal{F}(X) = 0$ may be inadequate when the estimated source variances are unreliable. Therefore, the L^2 norm of $\mathcal{F}(X)$ is used as a soft penalty term with weight α. The consistent estimate of X is obtained by minimizing the following objective function, $\psi_\alpha(X) = \psi(X) + \alpha \sum_{tf} \|\mathcal{F}(X_{tf})\|^2$, using a conjugate gradient descent method.

Applied as a post-processing step to demixed spectrograms, CWF makes the reconstructed spectrograms more amenable to inversion and therefore enhances the quality of the demixed audio signals. CWF requires as inputs

Figure 6: PESQ scores for all algorithms computed as average over all speakers and plotted as a function of (a) STFT exponent γ, (b) number of spectral columns m, and (c) STFT window size n. The PESQ scores are in alignment with the other measures of performance (SNR on the audio signal and normalized spectrogram residuals).

Table 1: Mean Values of the SNRs, Normalized Residuals, and PESQ Scores Averaged over All Speakers for the Sparse NMF-Based Methods Comparing Different Divergence Criteria.

	Mean SNR (dB)			Mean Normalized Residual			Mean PESQ Score		
NNSC Method	Euclidean Distance	KL Divergence	IS Divergence	Euclidean Distance	KL Divergence	IS Divergence	Euclidean Distance	KL Divergence	IS Divergence
NNSC (target learned)	4.98	5.75	4.86	0.6	0.51	0.64	1.86	1.92	1.74
NNSC (masker learned)	3.27	4.45	3.0	0.66	0.56	0.70	1.73	1.78	1.71
NNSC (both learned)	5.98	6.32	5.80	0.41	0.40	0.45	2.08	2.14	2.02

Note: The KL divergence criterion yields a better performance than the Euclidean distance criterion. In contrast, the IS divergence criterion performs slightly worse.

Table 2: SNRs, Normalized Residuals, and PESQ Scores, Averaged over All Speakers for the Proposed Demixing Methods without (Wiener Filtering (WF) and with Additional Adaptive Wiener Filtering (AWF) or Consistent Wiener Filtering (CWF).

	Mean SNR(dB) without WF (with AWF / with CWF)	Mean Normalized Residual without WF (with AWF / with CWF)	Mean PESQ Score without WF (with AWF / with CWF)
EACD	5.5, (**6.3/6.09**)	0.47, (**0.45/0.46**)	1.9, (**2.21/2.19**)
MLD/SR	6.42, (**6.96/7.21**)	0.38, (**0.34/0.35**)	2.35, (**2.56/2.56**)
NNSC (target learned)	4.98, (**5.13**/4.74)	0.6, (**0.58**/0.65)	1.86, (**1.9**/1.85)
NNSC (masker learned)	3.27, (**3.4**/3.01)	0.66, (**0.63**/0.74)	1.73, (**1.87**/1.55)
NNSC (both learned)	5.98, (**6.88/6.53**)	0.41, (**0.38/0.377**)	2.08, (**2.19/2.13**)

Notes: The improved performance values (over the methods without WF) are shown in bold. AWF improves the performance of all methods. CWF does not improve the performance of NNSC (target learned) and NNSC (masker learned) but improves it for all other methods. MLD/SR is MLD/suppression-regression. Note that applying CWF on top of AWF did not, on average, improve the performance for any of the demixing methods.

the power spectral densities (PSDs) for both the target and the masker (these correspond to S in equation 2.1 with $\gamma = 2$). Using the PSDs of our estimated audio signals for CWF, we report in Table 2 the separation performance following CWF.

CWF applied to our linear methods using the optimal set of parameters ($\gamma = 1$ and $n = 64$) improved SNRs averaged over all speakers of 0.79 dB for MLD/suppression-regression and of 0.59 dB for EACD. The normalized residual values were reduced on average by 0.03 for MLD/suppression-regression and by 0.01 for EACD. The PESQ scores averaged over all speakers improved by 0.21 for MLD/suppression-regression and by 0.29 for EACD. Note that for the NNSC-based approaches, consistent Wiener filtering improved only the results for NNSC (both learned).

6 Real-Time Implementations

The suitability of the linear methods was tested for real-time applications, and therefore two of them (EACD and MLD) were implemented in real-time. The computer used for this purpose had a 64-bit operating system and an Intel Core i7 processor with 2.70 GHz clock frequency and 8 GB of RAM. To minimize hardware latencies, an audio stream input output (ASIO) sound card driver was used. ASIO bypasses the normal audio path from a user application through layers of intermediary windows' operating systems software on a computer so that an application can communicate directly with the sound card. Each layer that is bypassed contributes to a reduction in latency. The audio signal is acquired at a sampling frequency of 44.1 KHz. The buffer size was kept at 512 samples ($\widehat{=}$23.2 ms) at both recording and playback ends. To achieve a lower latency and yet good performance, the STFT window size was kept at $n = 512 \widehat{=} 23.2$ ms and FFT window overlap at 75%. For simplicity, the number of spectral columns was set to $m = 1$ and the sparseness factor to $\gamma = 1$. The audio latency achieved was ≈46 ms, which was experienced as a well-tolerable latency. This work also tried to implement non-blind NNSC separation as a real-time algorithm. The dictionaries and code matrices for the two speakers were first learned using the training data. These pre-learned code matrices were then used as the starting code matrices (instead of random matrices) for the iterative update, thereby reducing the number of required iterations in equations 3.30 and 3.31. However, the audio latency achieved was ≈500 ms, intolerably large for real-time separation purposes.

7 Conclusion

This letter presented novel linear approaches to audio source separation: (1) eigenmode analysis of covariance difference (EACD) in which spectrotemporal features associated with large variance for one source and small variance for the other source are identified; (2) maximum likelihood demixing (MLD) in which the mixture is modeled as the sum of two gaussian signals and maximum likelihood is applied to identify the most likely sources present in the mixture; and (3) suppression-regression (SR) in which autoregressive models are trained to reproduce one source and suppress the other. The approaches in this work use only a single microphone recordings to perform source separation.

Unlike our proposed methods that perform monaural source separation, there exist various other source separation approaches that require multiple microphone recordings (Hyvärinen et al., 2001; Pham & Cardoso, 2001; Souden, Araki, Kinoshita, Nakatani, & Sawada, 2013). Many of them are based on maximum likelihood considerations like ours (Degerine & Zaidi, 2004; Fevotte & Cardoso, 2005; Pham & Cardoso, 2001). Nevertheless, these approaches differ from the proposed methods not only in terms of the number of inputs but also in other ways. For example, the probabilistic multispeaker model in Souden et al. (2013) is based on a latent variable assuming discrete states. The speech signal is reconstructed assuming that each state is associated with only one speaker. This assumption, as in binary masking-based methods, can lead to loss of information when time-frequency bins contain signals from multiple speakers. Our methods in principle are not limited by this constraint. Pham and Cardoso (2001) propose demixing methods based on maximum likelihood and minimum mutual information principles for gaussian non-stationary sources. By contrast, our proposed methods assume stationary sources but incorporate the temporal dependencies by concatenating consecutive spectral columns into a single vector, thereby increasing the timescale of the represented signal. Another multiple microphone source separation approach that uses the statistical independence and nonstationarity of the sources was proposed by Matsuoka, Ohoya, and Kawamoto (1995). In their algorithm, mixture signals are decorrelated using a neural network trained with stochastic gradient descent. Their approach has the advantage of being independent of the type of distribution of the individual sources. However, by comparison with the complex and iterative approaches proposed by Pham and Cardoso (2001) and by Matsuoka and colleagues (1995), our proposed methods learn the filters in a single step, thereby making our methods more efficient and suitable for real-time applications.

Overall, the linear methods for single-microphone source separation proposed in this letter perform better than more computationally demanding nonlinear approaches such as NNSC (Schmidt et al., 2007), in terms of both SNRs, residual spectrograms and PESQ scores. Unlike nonlinear NNSC, these linear approaches are not only simpler to implement but also faster to execute. Nevertheless, the semiblind NNSC approach has an advantage over the proposed linear approaches of being able to separate a target from an unknown masker or separating an unknown target from known noise. An interesting extension of this work could be to implement such abilities in future linear approaches.

This work is not the first to propose single-channel source separation methods based on the maximum likelihood approach (Jang & Lee, 2003). However, the proposed methods are more suitable for real-time applications because the method in Jang and Lee (2003) uses complex and iterative schemes for the separation task, whereas the methods here in essence require only fast-forward and inverse Fourier transformations and matrix multiplications.

An alternative approach to source separation is the use of binary masks on mixture spectrograms, that is, assigning each point in the time-frequency (TF) bin to the dominant source. The problem with binary masking and related approaches (Aoki et al., 2001; Brungart et al., 2006; Han & Wang, 2012; Jourjine et al., 2000; Nguyen et al., 2001; Rickard et al., 2001) is that artifacts or unnatural sounds may appear in the reconstructed signals. Efforts to combine ICA with binary masks have also been undertaken in Højen-Sørensen, Winther, and Hansen (2002); however, one of the general problems with binary masking is loss of information in the overlapping TF bins where the target utterance has lower energy than the masker. This occurs due to the fact that only one source is supposed to be active per TF bin. In contrast, the proposed linear methods do not exclusively assign a TF bin to one speaker only, but rather compute the subspaces associated with each of the sources and project the mixture onto them. This preserves the information for both speakers in a particular TF bin. In a fairly different approach to monaural source separation, Vishnubhotla and Espy-Wilson perform the segregation task by modeling the TF masking as a combination of complex sinusoids that are harmonics of the pitch frequencies of the speakers by applying a least-squares fitting approach (Vishnubhotla & Espy-Wilson, 2009). Deep neural networks have also been used to compute both binary masks (Wang & Wang, 2013) and soft masks (Huang et al., 2014) for the task of source separation. However, they are highly nonlinear and are computationally expensive. Linear methods are simpler to implement.

Human speech exhibits structure on multiple temporal scales, in line with natural sounds that tend to slowly vary in time (Bregman, 1990; Rosen, 1992). Congruently, human auditory processing shows a bias toward the perception of continuity in sound streams (Bregman, 1990), motivating the inclusion of temporal continuity into source separation methods. The method proposed in Lim, Shinn-Cunningham, and Gardner (2012) represents any discrete time series as a set of time-frequency contours. This method when applied to source separation allows sources to be extracted based on differences in contour representations of target and masker signals. While this method works well when the timescales of the two underlying signals are highly different, they are likely to fail in separation problems such as the multiple speaker problem when the underlying timescales of the two signals are very similar. Temporal continuity is incorporated naively in this work by simply concatenating consecutive spectral columns into a single vector, thereby increasing the timescale of the signal representation. Performance (in terms of SNR and residual) of all algorithms peaked for multiple-column representations, though less sensitively than expected. Concatenating too many spectral columns led to a reduction in performance, presumably due to overfitting.

Temporal continuity has also been addressed in a system proposed by Vincent and Rodet (2004) who modeled the activity of a source with a hidden Markov model. Such models are known to produce good separation results but are less suitable for real-time implementation. Virtanen (2007) incorporated temporal continuity of features by introducing a dedicated cost function. Because this cost function is computed iteratively, it might be difficult to implement this algorithm in real time.

Acknowledgments

This work was funded by Swiss National Science Foundation grant 200021-126844 “Early Auditory Based Recognition of Speech,” grant 200020-153565 “Fast Separation of Auditory Sounds,” and by European Research Council (ERC-Advanced Grant 268911).

References

Aoki, M., Okamoto, M., Aoki, S., Matsui, H., Sakurai, T., & Kaneda, Y. (2001). Sound source segregation based on estimating incident angle of each frequency component of input signals acquired by multiple microphones. *Acoust. Sci. Technology*, *22*, 149–157.

Berouti, M., Schwartz, R., & Makhoul, J. (1979). Enhancement of speech corrupted by acoustic noise. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (pp. 208–211). Piscataway, NJ: IEEE.

Bregman, A. S. (1990). *Auditory scene analysis*. Cambridge, MA: MIT Press.

Brungart, D. S., Chang, P. S., Simpson, B. D., & Wang, D. (2006). Isolating the energetic component of speech-on-speech masking with ideal time-frequency segregation. *J. Acoust. Soc. Am.*, *120*, 4007–4018.

Degerine, S., & Zaidi, A. (2004). Separation of an instantaneous mixture of gaussian autoregressive sources by the exact maximum likelihood approach. *IEEE Trans. Signal Process.*, *52*, 1492–1512.

Eggert, J., & Korner, E. (2004). Sparse coding and NMF. in *Proceedings of the IEEE International Joint Conference on Neural Networks* (pp. 2529–2533). Piscataway, NJ: IEEE.

Févotte, C., & Cardoso, J.-F. (2005). Maximum likelihood approach for blind audio source separation using time-frequency gaussian source models. In *Proceedings of the IEEE Workshop Applications of Signal Processing to Audio and Acoustics* (pp. 78–81). Piscataway, NJ: IEEE.

Févotte, C., & Idier, J. (2011). Algorithms for nonnegative matrix factorization with the β-divergence. *Neural Computation*, *23*(9), 2421–2456.

Han, K., & Wang, D. (2012). A classification based approach to speech segregation. *J. Acoust. Soc. Am.*, *132*, 3475–3483.

Hansen, P. C., & Jensen, S. H. (2005). Prewhitening for rank-deficient noise in subspace methods for noise reduction. *IEEE Trans. on Signal Process.*, *53*, 3718–3726.

Hansen, P. C., & Jensen, S. H. (2007). Subspace-based noise reduction for speech signals via diagonal and triangular matrix decompositions: Survey and analysis. *EURASIP Journal on Advances in Signal Process*, *1*, 092953.

Højen-Sørensen, P.A.d.F.R., Winther, O., & Hansen, L. K. (2002). Analysis of functional neuroimages using ICA with adaptive binary sources. *Neurocomputing*, *49*, 213–225.

Hoyer, P. O. (2002). Non-negative sparse coding. In *Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing* (pp. 557–565). Piscataway, NJ: IEEE.

Hu, Y., & Loizou, P. C. (2008). Evaluation of objective quality measures for speech enhancement. *IEEE Trans. Audio. Speech Lang. Process*, *16*, 229–238.

Huang, P. S., Kim, M., Hasegawa-Johnson, M., & Smaragdis, P. (2014). Deep learning for monaural speech separation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 1581–1585). Piscataway, NJ: IEEE.

Hyvärinen, A., Karhunen, J., & Oja, E. (2001). What is independent component analysis? In S. Haykin (Ed.), *Independent component analysis* (pp. 145–164). New York: Wiley.

Jang, G.-J., & Lee, T.-W. (2003). A maximum likelihood approach to single-channel source separation. *J. Mach. Learn. Res.*, *4*, 1365–1392.

Jensen, S. H., Hansen, P. C., Hansen, S. D., & Sorensen, J. A. (1995). Reduction of broad-band noise in speech by truncated QSVD. *IEEE Transactions on Speech and Audio Processing*, *3*(6), 439–448.

Jourjine, A., Rickard, S., & Yilmaz, O. (2000). Blind separation of disjoint orthogonal signals: demixing N sources from 2 mixtures. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (pp. 2985–2988). Piscataway, NJ: IEEE.

Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, *401*, 788–791.

Lee, D. D., & Seung, H. S. (2006). Algorithms for non-negative matrix factorization. In T. K. Leen, T. G. Dietterich, & V. Tresp (Eds.), *Advances neural information processing systems*, *13* (pp. 556–562). Cambridge, MA: MIT Press.

Lim, Y., Shinn-Cunningham, B., & Gardner, T. J. (2012). Sparse contour representations of sound. *IEEE Signal Process. Lett.*, *19*, 684–687.

Lin, C.-J. (2007). Projected gradient methods for nonnegative matrix factorization. *Neural Computation*, *19*, 2756–2779.

Loizou, P. C. (2013). *Speech enhancement: Theory and practice*. Boca Raton, FL: CRC Press.

Machens, C. K., Romo, R., & Brody, C. D. (2010). Functional, but not anatomical, separation of "what" and "when" in prefrontal cortex. *J. Neurosci.*, *30*, 350–360.

Matsuoka, K., Ohoya, M., & Kawamoto, M. (1995). A neural net for blind separation of nonstationary signals. *Neural Netw.*, *8*, 411–419.

Narayanan A., & Wang D. L. (2013). Ideal ratio mask estimation using deep neural networks for robust speech recognition. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (pp. 7092–7096). Piscataway, NJ: IEEE.

Narayanan A., & Wang D. L. (2014). Investigation of speech separation as a frontend for noise robust speech recognition. *IEEE Trans. Audio. Speech Lang. Process.*, *22*, 826–835.

Nguyen, L. T., Belouchrani, A., Abed-Meraim, K., & Boashash, B. (2001). Separating more sources than sensors using time-frequency distributions. In *Proceedings of the Sixth International Symposium on Signal Processing and Its Applications* (pp. 583–586). Piscataway, NJ: IEEE.

Petersen, K. B., & Pedersen, M. S. (2008). The matrix cookbook. Technical University of Denmark.

Pham, D.-T., & Cardoso, J. F. (2001). Blind separation of instantaneous mixtures of nonstationary sources. *IEEE Trans. Signal Process.*, *49*, 1837–1848.

Rickard, S., Balan, R., & Rosca, J. (2001). Real-time time-frequency based blind source separation. In *Proceedings of the International Workshop on Independent Component Analysis and Blind Source Separation* (pp. 651–656). New York: Springer.

Rosen, S. (1992). Temporal information in speech: Acoustic, auditory and linguistic aspects. *Phil. Trans. Roy. Soc. Lond. Series B: Biol. Sci.*, *336*, 367–373.

Roux, J., & Vincent, E. (2013). Consistent Wiener filtering for audio source separation. *Signal Processing Letters, IEEE*, *20*(3), pp. 217–220.

Schmidt, M. N., Larsen, J., & Hsiao, F.-T. (2007). Wind noise reduction using non-negative sparse coding. In *Proceedings of the IEEE Workshop on Machine Learning for Signal Processing* (pp. 431–436). Piscataway, NJ: IEEE.

Souden, M., Araki, S., Kinoshita, K., Nakatani, T., & Sawada, H. (2013). A multichannel MMSE-based framework for speech source separation and noise reduction. *IEEE Trans. Audio. Speech Lang. Process.*, *21*, 1913–1928.

Tikhonov, A. N., & Arsenin, V. Y. (1977). *Solutions of ill-posed problems*. Ultimo, NSW: Halsted Press.

Vincent, E., & Rodet, X. (2004). Music transcription with ISA and HMM. *Lecture Notes in Computer Science* (pp. 1197–1204). New York: Springer.

Virtanen, T. V. T. (2007). Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria. *IEEE Trans. Audio. Speech Lang. Process*, *15*, 1066–1074.

Vishnubhotla, S., & Espy-Wilson, C. Y. (2009). An algorithm for speech segregation of co-channel speech. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 109–112). Piscataway, NJ: IEEE:

Wang, D. L., & Brown, G. J. (2006). Fundamentals of computational auditory scene analysis. In D. L. Wang & G. J. Brown (Eds.), *Computational auditory scene analysis: Principles, algorithms, and applications* (pp. 1–44). (Hoboken, NJ: Wiley and IEEE Press.

Wang Y., & Wang D. L. (2013). Towards scaling up classification-based speech separation. *IEEE Trans. Audio. Speech Lang. Process.*, *21*, 1381–1390.

Weiss, R. J. (2009). Underdetermined source separation using speaker subspace models. Doctoral dissertation, Columbia University.

Zhao X., Wang Y., & Wang D. L. (2014). Robust speaker identification in noisy and reverberant conditions. *IEEE Trans. Audio. Speech Lang. Process.*, *22*, 836–845.

Received December 9, 2014; accepted June 22, 2015.