Monaural Sound Source Separation by Nonnegative Matrix Factorization With Temporal Continuity and Sparseness Criteria

Tuomas Virtanen

***Abstract*—An unsupervised learning algorithm for the separation of sound sources in one-channel music signals is presented. The algorithm is based on factorizing the magnitude spectrogram of an input signal into a sum of components, each of which has a fixed magnitude spectrum and a time-varying gain. Each sound source, in turn, is modeled as a sum of one or more components. The parameters of the components are estimated by minimizing the reconstruction error between the input spectrogram and the model, while restricting the component spectrograms to be nonnegative and favoring components whose gains are slowly varying and sparse. Temporal continuity is favored by using a cost term which is the sum of squared differences between the gains in adjacent frames, and sparseness is favored by penalizing nonzero gains. The proposed iterative estimation algorithm is initialized with random values, and the gains and the spectra are then alternatively updated using multiplicative update rules until the values converge. Simulation experiments were carried out using generated mixtures of pitched musical instrument samples and drum sounds. The performance of the proposed method was compared with independent subspace analysis and basic nonnegative matrix factorization, which are based on the same linear model. According to these simulations, the proposed method enables a better separation quality than the previous algorithms. Especially, the temporal continuity criterion improved the detection of pitched musical sounds. The sparseness criterion did not produce significant improvements.**

***Index Terms*—Acoustic signal analysis, audio source separation, blind source separation, music, nonnegative matrix factorization, sparse coding, unsupervised learning.**

I. Introduction

In real-world audio signals, several sound sources are usually mixed. The process in which individual sources are estimated from the mixture signal is called sound source separation. Separation of mixed sounds has several applications in the analysis, editing, and manipulation of audio data. These include for example structured audio coding and automatic transcription of music. There are effective algorithms for the processing of isolated sounds; therefore, the capability of separating sources from polyphonic mixtures is very appealing. In this paper, the focus is on the sound separation in music signals.

The definition of a sound source depends somewhat on the application. Usually, the term is used to refer to an individual physical source or to an entity that humans perceive as an entity. Humans are extremely skillful in "hearing out" individual sources from complex mixtures even in noisy conditions. Computational modeling of this ability is very difficult: all the existing separation systems have a limited separation quality, and usually the complexity of the target signals has to be restricted. The most successful algorithms have been those which try to extract only the most prominent source [1], [2], or which utilize prior information of the source signals [3], [4].

Recently, unsupervised machine learning algorithms have been successfully used in one-channel source separation. These are typically based on a simple linear model, and instead of using prior knowledge of the source signal characteristics, the separation is done by finding a decomposition where the sources are statistically independent or nonredundant. Algorithms have been proposed that are based on independent component analysis (ICA) [5]–[7], nonnegative matrix factorization (NMF) [8], and sparse coding [9]–[11].

This paper proposes an unsupervised sound source separation algorithm which combines NMF with temporal continuity and sparseness objectives. The proposed algorithm is shown to provide a better separation quality than the existing algorithms. The outline of this paper is as follows. Section II gives an overview of the existing unsupervised learning separation algorithms. The proposed method is explained in Section III, and simulation experiments are presented in Section IV.

II. Unsupervised Learning Algorithms in Sound Source Separation

Most of the above-mentioned algorithms for unsupervised sound source separation are based on a signal model where the magnitude or power spectrum vector $\mathbf{x}_t$ in frame t is modeled as a linear combination of basis functions $\mathbf{b}_j$. This can be written as

$$\mathbf{x}_t = \sum_{j=1}^{J} g_{j,t}\mathbf{b}_j \tag{1}$$

where J is the number of basis functions, and $g_{j,t}$ is the gain of the jth basis function in frame t.

The term *component* is used to refer to one basis function $\mathbf{b}_j$ and its time-varying gain $g_{j,t}, t = 1, \ldots, T, T$ being the number of frames. Each source is modeled as a sum of one or more components. The separation is done by first factorizing the spectrogram of the input signal into components and then grouping these to sound sources.

Manuscript received June 3, 2005; revised July 31, 2006. This work was supported by the Academy of Finland under Project 213462, Finnish Centre of Excellence program (2006–2011). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Michael Davies.

The author is with the Tampere University of Technology, Tampere FI-33101, Finland (e-mail: tuomas.virtanen@tut.fi).

Digital Object Identifier 10.1109/TASL.2006.885253

In the case of music signals, each component usually represents a musically meaningful entity or parts of it, so that different entities are represented with different components. The entities can be for example the sounds produced by a percussive instrument or all equal-pitched notes of a pitched musical instrument. This representation is enabled by the spectral structure of musical sounds, which is usually rather static over time compared to speech signals, for example.

The time-domain signals of concurrent sounds and their complex spectra sum linearly. However, the phase spectra of natural sounds are very unpredictable, and estimation of the framewise phases of the sources would make the model too complex. Also, the human auditory perception is quite insensitive to phase. Therefore complex spectra are usually not used in this framework but the linear addition of complex spectra is to be approximated as a linear addition of magnitude or power spectra. The expectation value for the power spectrum of the superposition of arbitrary complex sources with unknown random phase spectra equals the sum of the power spectra of the sources, provided that the phase spectra of the sources are independent. However, in many systems the linear summation of magnitude spectra has turned out to produce better results. In this paper, the observation vector and the basis functions are magnitude spectra.

The model (1) can be written using a matrix notation as

$$\mathbf{X} = \mathbf{BG} \tag{2}$$

where $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_T], \mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_J]$ and $[\mathbf{G}]_{j,t} = g_{j,t}$. In this paper we discuss methods which estimate the sources blindly, i.e., only the observation matrix $\mathbf{X}$ is known, and matrices $\mathbf{B}$ and $\mathbf{G}$ are estimated without source-specific prior knowledge. In the following sections, we briefly review some commonly used separation principles. The model (1) can be extended to allow time-varying components; some proposals have been made by Blumensath and Davies [12], Smaragdis [13], and Virtanen [14]. More complex models can potentially enable better separation quality, but in this paper we compare only methods which are based on the linear model (1).

1) Independent Subspace Analysis (ICA): ICA has been successfully used to solve blind source separation problems in several application areas. ICA separates an observation vector $\mathbf{x}$ by finding an unmixing matrix $\mathbf{W}$, so that the estimated variables, i.e., the elements of vector $\hat{g} = \mathbf{Wx}$ are statistically independent from each other. The convolutive extension of ICA ([15], pp. 361–370) suits well for multichannel sound source separation, where the elements of the observation vector are the signals recorded with different microphones. In ICA the number of sources has to be less than or equal to the number of input variables, i.e., the length of the observation vector. Therefore, ICA cannot be directly used for the separation of monaural time-domain signals.

Independent subspace analysis (ISA) tries to remove the above limitation. The magnitude or power spectrogram can be viewed as a phase-invariant projection of the input signal, and the amplitude of each frequency line as a phase-invariant feature calculated in each frame. The factorization of the spectrogram can be seen as separation of phase-independent features into *invariant feature subspaces* [16]. Letting the magnitude or power spectrum in frame t to be the observation, the separation can be done using basic ICA, as explained above. With this procedure the estimated gains of different components are statistically independent from each other.

ISA has been used in one-channel sound source separation, for example, by Casey and Westner [5], Orife [17], FitzGerald, Coyle, and Lawlor [18], Uhle, Dittmar, and Sporer [19], and Brown and Smaragdis [7]. Also, a sound recognition system based on ISA has been adopted in the MPEG-7 standardization framework [20].

2) Nonnegative Matrix Factorization: In addition to statistical independence, some other estimation principles have also been found useful in finding the decomposition $\mathbf{X} = \mathbf{BG}$. Each component is modeled using a fixed magnitude or power spectrum, which are nonnegative by definition. It is therefore natural to restrict the basis functions to be entry-wise nonnegative. Moreover, the components can be restricted to be purely additive, meaning that the gains are restricted to be nonnegative.

The NMF algorithms proposed by Lee and Seung [21] do the decomposition by minimizing the reconstruction error between the observation matrix $\mathbf{X}$ and the model $\mathbf{BG}$ while constraining the matrices to be entry-wise nonnegative. The algorithms have been used in several unsupervised learning tasks and also in the analysis of music signals, where the nonnegativity constraints alone have turned out to be sufficient for the separation of sound sources [8]. Time-frequency representations of natural sound sources are often *sparse*, meaning that most of the frames and frequencies are inactive. Sparse spectrograms often have a unique decomposition into nonnegative components, each of which represents parts of a single sound source.

Lee and Seung used two measures for the reconstruction error, and proposed the corresponding minimization algorithms for calculating the factorization. The measures are the square of the Euclidean distance, $\sum_{k,t}([\mathbf{X}]_{k,t} - [\mathbf{BG}]_{k,t})^2$, and divergence D, which is defined as

$$D(\mathbf{X} \,||\, \mathbf{BG}) = \sum_{k,t} [\mathbf{X}]_{k,t} \log \frac{[\mathbf{X}]_{k,t}}{[\mathbf{BG}]_{k,t}} - [\mathbf{X}]_{k,t} + [\mathbf{BG}]_{k,t}. \tag{3}$$

The above measures implicitly assume a certain noise distribution: Euclidean distance is the maximum likelihood estimator (MLE) in the presence of additive Gaussian noise, and the divergence is the MLE when the observations are generated by a Poisson process with mean value $[\mathbf{BG}]_{k,t}$. This naturally affects the performance of the methods. For example, the divergence is more sensitive to low-energy observations, making it a better approximation of human auditory perception.

3) Sparse Coding: An unsupervised learning technique called sparse coding has been successfully used for example to model the early stages of vision [22]. The term sparse refers to a signal model, where the data is represented in terms of a small number of active elements chosen out of a larger set. In the signal model (1), this means that the probability of the gain $g_{j,t}$ to be zero is high. This can be expressed by using a prior probability density function (pdf) for the gains, and the estimation can be done using maximum *a posteriori* (MAP)

estimation. This leads to the minimization of a cost function which is usually a sum of a reconstruction error term and a term which penalizes nonzero gains $g_{j,t}$. Sparse coding has been used for audio signal separation by Abdallah and Plumbley [9], [11], Benaroya, McDonagh, Bimbot, and Gribonval [23], and Blumensath and Davies [12], to mention a few examples.

The nonnegative sparse coding algorithm proposed by Hoyer [24] combines NMF and sparse coding. He estimated the matrices by combining the multiplicative update rule proposed by Lee and Seung [21] with projected gradient descent [discussed, e.g., in ([25], pp. 203–224)]. The algorithm was used with a temporal continuity criterion for sound separation by Virtanen [10].

No large-scale evaluation has been carried out to investigate whether the use of a sparse prior increases the separation quality in the case of audio signals. The spectra which typically constitute a sound source are usually active in a small fraction of frames, and therefore their gains have a sparse pdf. However, the MAP estimator in [22] assumes that each frame is independent from each other, which is not a realistic assumption for natural sound sources.

4) Components With Relation to Sources: In addition to the factorization algorithm, there is a need to determine a suitable number of components and to group them to sources. Since the definition of a sound source depends on the application, there is no universal way of selecting the number of components. In the separation of drum patterns, FitzGerald found out that certain drum instruments can be well modeled using a single component [26, pp. 93–100]. In the case of more complex signals, typically a large number of components are used which are then clustered to sound sources.

Automatic clustering of the components has turned out to be a difficult task. Some unsupervised clustering methods have been proposed [5], [6], but in our simulations their performance was not sufficient. Supervised clustering based on pattern recognition techniques produces better results [27], [28], but these require that the sources are known and their models trained beforehand. In this paper, we do not consider the clustering problem but circumvent this step by using the original signals as a reference, as explained in Section IV-C.

III. Proposed Method

An input signal is represented using the magnitude spectrogram, which is calculated as follows. First, the time-domain signal is divided into frames and windowed. A fixed 40-ms frame size is used with 50% overlap between frames. The discrete Fourier transform (DFT) is applied on each frame, the length of the DFT being equal to the frame size. Only positive frequencies are retained and phases are discarded by taking the absolutive values of the DFT spectra, resulting in a magnitude spectrogram matrix $[\mathbf{X}]_{k,t}$ where $k = 1, \ldots, K$ is the discrete frequency index and $t = 1, \ldots, T$ is the frame index.

The magnitude spectrogram is modeled as a product of the basis matrix $\mathbf{B}$ and the gain matrix $\mathbf{G}$, so that $\mathbf{X} \approx \mathbf{BG}$, while restricting $\mathbf{B}$ and $\mathbf{G}$ to be entry-wise nonnegative. This models the linear summation of the magnitude spectrograms of the components. As discussed in the previous section, the summation of power spectra is theoretically better justified; however, in our simulation experiments (Section IV), the best results were obtained by assuming linear summation of the magnitude spectra, and therefore the proposed method is formulated using magnitude spectrograms.

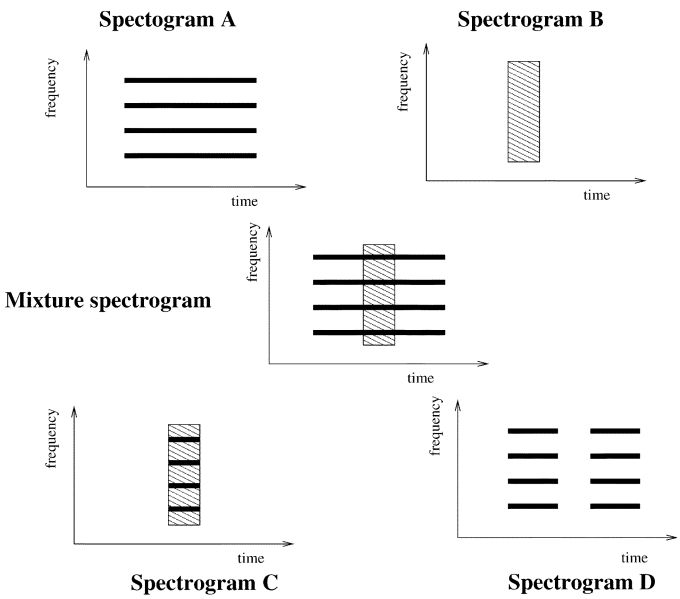

Fig. 1. Simple example which illustrates how the temporal continuity of sources can improve the separation. See text for details.

Estimation of $\mathbf{B}$ and $\mathbf{G}$ is done by minimizing a cost function $c(\mathbf{B}, \mathbf{G})$, which is a weighted sum of three terms: a reconstruction error term $c_{\mathrm{r}}(\mathbf{B}, \mathbf{G})$, a temporal continuity term $c_{\mathrm{t}}(\mathbf{G})$, and a sparseness term $c_{\mathrm{s}}(\mathbf{G})$

$$c(\mathbf{B}, \mathbf{G}) = c_{\mathrm{r}}(\mathbf{B}, \mathbf{G}) + \alpha c_{\mathrm{t}}(\mathbf{G}) + \beta c_{\mathrm{s}}(\mathbf{G}) \tag{4}$$

where α and β are the weights of the latter two terms, respectively.

A. Reconstruction Error Term

The human auditory system has a wide dynamic range: the difference between the threshold of hearing and the threshold of pain is approximately 100 dB [29]. Unsupervised learning algorithms tend to be more sensitive to high-energy observations, and some methods fail to separate low-energy sources even though these are perceptually and musically meaningful.

The divergence cost (3) of an individual observation $[\mathbf{X}]_{k,t}$ is linear as a function of the scale of the input, since $D(\gamma p \,\|\, \gamma q) = \gamma D(p \,\|\, q)$ for any positive scalar γ, whereas for the Euclidean cost the dependence is quadratic. Therefore, the divergence is more sensitive to small-energy observations. Among the tested reconstruction error measures (including also those proposed in [11] and [14]), the divergence produced the best results, and therefore we measure the reconstruction error using the divergence (3).

B. Temporal Continuity Criterion

The separation methods discussed in Section II consider each frame as an individual observation. However, real-world sounds usually have a temporal structure, and their acoustic characteristics vary slowly as a function of time. Fig. 1 shows a simple example where the temporal continuity criterion would increase

the robustness of the separation. The two sources A and B represent a typical sustained harmonic sound and a typical short-duration percussive sound, respectively. The observed mixture spectrogram (illustrated in the middle panel) is separated into two components both of which have a fixed spectrum and time-varying gain. When the separation is done by minimizing the reconstruction error between the observed spectrogram and the separated spectrograms, it is possible to obtain the original spectrograms A and B. However, it is also possible to represent the mixture spectrogram as a sum of spectrograms C and D, resulting in error. By favoring temporal continuity, the separation can be directed towards the spectrograms A and B.

Temporal continuity was addressed in a system proposed Vincent and Rodet who modeled the activity of a source by a hidden Markov model [30]. In this paper, we apply a simple temporal continuity criterion which does not require training beforehand.

Temporal continuity of the components is measured by assigning a cost to large changes between the gains $g_{j,t}$ and $g_{j,t-1}$ in adjacent frames. We propose to use the sum of the squared differences between the gains. To prevent the numerical scale of the gains from affecting the cost, the gains are normalized by their standard deviation estimates σ_j, so that the cost function c_{t} for the temporal continuity can be written as

$$c_{\mathrm{t}}(\mathbf{G}) = \sum_{j=1}^{J} \frac{1}{\sigma_j^2} \sum_{t=2}^{T} (g_{t,j} - g_{t-1,j})^2. \tag{5}$$

If the normalization was not used, the function $c_{\mathrm{t}}(\mathbf{G})$ could be minimized without affecting the reconstruction error by scaling the matrices by $\mathbf{B} \leftarrow \mathbf{B}\theta$ and $\mathbf{G} \leftarrow \mathbf{G}/\theta$, where θ is a large positive scalar. The standard deviation of each component $j = 1, \ldots, J$ is estimated as $\sigma_j = \sqrt{(1/T)\sum_{t=1}^{T} g_{t,j}^2}$.

In [10], a cost function was used which was the sum of the absolute values of the difference of gains of adjacent frames. The motivation was that for a gain rising from a level to another, the absolute value cost is equal for all monotonically increasing transitions. However, it was found that the absolute value of the differences did not increase the performance of the separation algorithm as much as the squared differences. The reason for this might be the used iterative optimization algorithms, in which the gradient of the absolute values depends only on the sign of the difference, not on the magnitude.

C. Sparseness Objective

The sparseness of the gains $g_{j,t}$ has been utilized in several blind source separation algorithms, and in some cases the sparseness criterion improves the quality. For example, when the spectrum of one source (e.g., kick drum) covers partly the spectrum of another (e.g., snare drum), the latter source could be modeled as a sum of the first sound and a residual. The use of sparse gains can favor a representation where only a single spectrum is used to model the latter source.

The sparseness criterion, which can be derived from the MAP estimation of the sources [22], is formulated as

$$c_{\mathrm{s}}(\mathbf{G}) = \sum_{j=1}^{J} \sum_{t=1}^{T} f(g_{j,t}/\sigma_j) \tag{6}$$

where $f(\cdot)$ is a function which penalizes nonzero gains. For example, Olshausen and Field [22] suggested functions $f(x) = \log(x^2+1)$, $f(x) = |x|$, and $f(x) = -\exp(-x^2)$. The first two were tested in our simulations. The differences between these were small, but $f(x) = |x|$ was found to be less sensitive for the weight β, and therefore it is used in the proposed method.

D. Estimation Algorithm

In the estimation algorithm, the matrices $\mathbf{B}$ and $\mathbf{G}$ are first initialized with random positive values and then alternatively updated with multiplicative update rules. The value of the cost function decreases until the algorithm converges. Compared with earlier approaches based on projected steepest descent (for example [24] and [10]), the multiplicative update rules are convenient in a sense that they do not require estimating a suitable step size.

Currently, there is no reliable method for the automatic estimation of the number of components, but this has to be set manually. In practice, a large number of components can be used, which are then clustered to sound sources. If some prior information about the sources is available, it can be used to select the number of components or to initialize the spectra.

In the cost function (4), $\mathbf{B}$ affects only the reconstruction error term $c_{\mathrm{r}}(\mathbf{B}, \mathbf{G})$. For the minimization of the reconstruction error term the update rules proposed by Lee and Seung [21] can be used. The update rule for $\mathbf{B}$ is given by

$$\mathbf{B} \leftarrow \mathbf{B}. \times \frac{\frac{\mathbf{X}}{\mathbf{BG}}\mathbf{G}^{\mathsf{T}}}{\mathbf{1}\mathbf{G}^{\mathsf{T}}} \tag{7}$$

in which $\mathbf{A}.\times\mathbf{B}$ and $(\mathbf{A}/\mathbf{B})$ are the element-wise multiplication and division of matrices $\mathbf{A}$ and $\mathbf{B}$, respectively, and $\mathbf{1}$ is a all-one matrix of the same size as $\mathbf{X}$. The divergence (3) has been shown to be nonincreasing under the update rule (7) [21].

A multiplicative update rule for the gain matrix $\mathbf{G}$ is derived as follows. First, the gradients of the reconstruction error cost $c_{\mathrm{r}}(\mathbf{B}, \mathbf{G})$, temporal continuity cost $c_{\mathrm{t}}(\mathbf{G})$, and sparseness cost $c_{\mathrm{s}}(\mathbf{G})$ with respect to $\mathbf{G}$ are given by

$$\nabla c_{\mathrm{r}}(\mathbf{B}, \mathbf{G}) = \mathbf{B}^{\mathsf{T}}\left(\mathbf{1} - \frac{\mathbf{X}}{\mathbf{BG}}\right) \tag{8}$$

$$\begin{aligned}[\nabla c_{\mathrm{t}}(\mathbf{G})]_{j,t} = {} & 2T\frac{2g_{j,t} - g_{j,t-1} - g_{j,t+1}}{\sum_{i=1}^{T} g_{j,i}^2} \\ & - T\frac{2g_{j,t}\sum_{i=2}^{T}(g_{j,i} - g_{j,i-1})^2}{\left(\sum_{i=1}^{T} g_{j,i}^2\right)^2}\end{aligned} \tag{9}$$

and

$$[\nabla c_{\mathrm{s}}(\mathbf{G})]_{j,t} = \frac{1}{\sqrt{\frac{1}{T}\sum_{i=1}^{T} g_{j,i}^2}} - \sqrt{T}\frac{g_{j,t}\sum_{i=1}^{T} g_{j,i}}{\left(\sum_{i=1}^{T} g_{j,i}^2\right)^{3/2}} \tag{10}$$

respectively.

The gradient of the total cost $c(\mathbf{B}, \mathbf{G})$ is the weighted sum of the gradients of the reconstruction error, temporal error, and sparseness error, given by

$$\nabla c(\mathbf{B}, \mathbf{G}) = \nabla c_{\mathrm{r}}(\mathbf{B}, \mathbf{G}) + \alpha \nabla c_{\mathrm{t}}(\mathbf{G}) + \beta \nabla c_{\mathrm{s}}(\mathbf{G}). \tag{11}$$

The gradient is written as a subtraction $\nabla_c(\mathbf{B},\mathbf{G}) = \nabla c^+(\mathbf{B},\mathbf{G}) - \nabla c^-(\mathbf{B},\mathbf{G})$ of element-wise nonnegative terms $\nabla_c^+(\mathbf{B},\mathbf{G}) = \nabla c_{\mathrm{r}}^+(\mathbf{B},\mathbf{G}) + \alpha\nabla c_{\mathrm{t}}^+(\mathbf{G}) + \beta\nabla c_{\mathrm{s}}^+(\mathbf{G})$ and $\nabla_c^-(\mathbf{B},\mathbf{G}) = \nabla c_{\mathrm{r}}^-(\mathbf{B},\mathbf{G}) + \alpha\nabla c_{\mathrm{t}}^-(\mathbf{G}) + \beta\nabla c_{\mathrm{s}}^-(\mathbf{G})$, where the element-wise positive terms of the gradients of reconstruction error cost, temporal continuity cost, and sparseness cost are given by

$$\nabla c_{\mathrm{r}}^+(\mathbf{B},\mathbf{G}) = \mathbf{B}^{\mathsf{T}}\mathbf{1} \tag{12}$$

$$\nabla c_{\mathrm{r}}^-(\mathbf{B},\mathbf{G}) = \mathbf{B}^{\mathsf{T}}\frac{\mathbf{X}}{\mathbf{B}\mathbf{G}} \tag{13}$$

$$[\nabla c_{\mathrm{t}}^+(\mathbf{G})]_{j,t} = \frac{4Tg_{j,t}}{\sum_{i=1}^{T} g_{j,i}^2} \tag{14}$$

$$[\nabla c_{\mathrm{t}}^-(\mathbf{G})]_{j,t} = 2T\frac{g_{j,t-1}+g_{j,t+1}}{\sum_{i=1}^{T} g_{j,i}^2} + \frac{2Tg_{j,t}\sum_{i=2}^{T}(g_{j,i}-g_{j,i-1})^2}{\left(\sum_{i=1}^{T} g_{j,i}^2\right)^2} \tag{15}$$

$$[\nabla c_{\mathrm{s}}^+(\mathbf{G})]_{j,t} = \frac{1}{\sqrt{\frac{1}{T}\sum_{i=1}^{T} g_{j,i}^2}} \tag{16}$$

and

$$[\nabla c_{\mathrm{s}}^-(\mathbf{G})]_{j,t} = \frac{g_{j,t}\sqrt{T}\sum_{i=1}^{T} g_{j,i}}{\left(\sum_{i=1}^{T} g_{j,i}^2\right)^{3/2}}. \tag{17}$$

The terms (12)–(17) are element-wise nonnegative, since the gains, basis functions, and observations are restricted to nonnegative values.

Finally, the update rule for $\mathbf{G}$ is given by

$$\mathbf{G} \leftarrow \mathbf{G}. \times \frac{\nabla c^-(\mathbf{B},\mathbf{G})}{\nabla c^+(\mathbf{B},\mathbf{G})}. \tag{18}$$

The overall iterative algorithm is as follows:

1) Initialize each element of $\mathbf{B}$ and $\mathbf{G}$ with the absolute value of Gaussian noise.
2) Update $\mathbf{B}$ using the update rule (7).
3) Update $\mathbf{G}$ using the update rule (18).
4) Evaluate the value of the cost function $c(\mathbf{B},\mathbf{G})$.

The steps 2–4 are repeated until the value of the cost function converges. The iteration is stopped when the decrease has been smaller than a predefined threshold for a certain number of iterations. For a 10-s input signal and 20 components the algorithm takes a couple of hundred iterations to converge, equivalent to a couple of minutes of computation on a 1.7-GHz desktop PC when implemented in Matlab.

The multiplicative update (18) does not necessarily decrease the value of the cost function. In the simulation experiments presented in Section IV, we applied the multiplicative update rules on 300 signals, each of which was tested with four different component counts and several combinations of α and β (see Fig. 2). We observed a total of five cases where the value of the cost function increased, which took places when α had a large value. We tested minimizing the cost function also by projected steepest descent, and obtained almost identical results, with the expense of increased computational complexity. This and the small amount of cost increases show that the multiplicative updates are sufficient for minimizing the cost function.

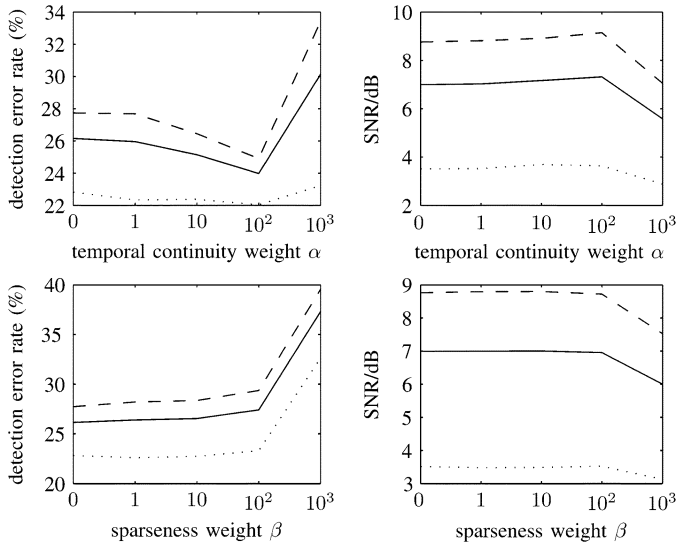

Fig. 2. Effect of different temporal continuity weights α and sparseness weights β on the detection error rate and SNR when other parameter was 0. The solid line is the average of all sources, the dashed line is the average of pitched sounds, and the dotted line is the average of drums.

E. Synthesis

The spectra and gains of the estimated components can be used directly in some acoustic analysis applications. In order to verify the quality by listening, however, the components have to be synthesized to obtain time-domain signals for each source.

In the synthesis, the magnitude spectrum of each component $j = 1,\ldots,J$ within frames $t = 1,\ldots,T$ is calculated as $\mathbf{b}_j g_{j,t}$. To get complex spectra, the phases of the original spectrogram can be used for the separated components, or the phase generation method proposed by Griffin and Lim [31] with the improvements proposed by Slaney, Naar, and Lyon [32] can be used.

In most cases where the separation is successful, the use of the original phases produces good results. It also allows the synthesis of sharp attacks with an accuracy which would otherwise be impossible with the 40-ms window size. However, if the original phases are not suitable for the separated magnitude spectrogram, the resulting time-domain signal may become distorted because of discontinuities at frame boundaries. These are partly attenuated by an overlap-add procedure where each synthesized frame is windowed before combining adjacent frames.

Reliable comparison of different synthesis methods is difficult. The simple approach of comparing the synthesized time-domain signals with original ones cannot be used, since the method in [31] produces time-domain signals which are not necessarily phase-aligned with the original signals.

IV. Simulation Experiments

Reliable evaluation of the quality of a sound source separation algorithm requires a large amount of signals, which makes listening tests slow to conduct and expensive. Therefore, computational procedures are often used which compare reference source signals with the separated signals. To do this, source signals before mixing are required, and in practice, this limits us to the use of generated test signals.

TABLE I
PARAMETERS USED TO GENERATE THE TEST SIGNALS

parameter	interval
number of pitched instrument sources	[0 12]
number of drum sources	[0 6]
length of each pitched musical sound (s)	[0.15 1]
number of notes per drum source	[2 8]
onset time of each repetition (s)	[0 6]
energy of each source (dB)	[0 -20]

A. Acoustic Material

Test signals were generated by mixing samples of pitched musical instruments and drums. The pitched sounds were from a database of individual notes which is a combination of samples from the McGill University Master Samples Collection [33], the University of Iowa website [34], and samples recorded from the Roland XP-30 synthesizer. The instruments introduce several sound production mechanisms, variety of spectra, and also modulations, such as vibrato. The total number of samples available for generating the mixtures was 4128, each having the sampling frequency 44 100 Hz. The drum samples were from the DFH Superior commercial sample database [35], which contains individual drum hits from several drum kits and instruments. Each instrument is multisampled, i.e., the recording is repeated several times for each instrument.

Mixture signals were generated by choosing a random number of pitched instrument sources and a random number of drum sources. For each mixture, the number of sources was chosen randomly from within the limits shown in Table I. Once the number of sources had been chosen, each source was picked randomly from the databases. For pitched-instrument sources, a random instrument and a random fundamental frequency from the available samples were allotted, and for drum sources a random drum kit and a random drum instrument were allotted.

Each pitched instrument sample was used only once within a mixture, and they were truncated to random lengths. A random number of repetitions were used for each drum sound, each repetition being an unique sample. The location of each note was randomized by allotting an onset time between 0 and 6 s. The length of all test signals was chosen to be 7 s. This resulted in material where 79% of the frames contained more than one source, i.e., the sources were mainly overlapping.

Since real-world signals incorporate sources with different mixing levels, each source was scaled to obtain a random total energy between 0 and -20 dB. The reference signals before mixing were stored to allow the measurement of separation quality. The total number of test mixtures was 300. Audio examples of the mixtures and separated signals are available at http://www.cs.tut.fi/~tuomasv/demopage.html.

It should be noted that the acoustic material differs from real-world music in a sense that it consists of individual notes instead of note sequences. However, none of the tested methods is able to utilize different pitch values of a source in the separation, and it is therefore unlikely that the results would be significantly different if note sequences were present.

B. Evaluated Algorithms

Some recently published algorithms were used as a baseline in the evaluation. All the algorithms apply a 40-ms window length, Hanning window, and short-time Fourier transform to calculate a spectrogram $\mathbf{X}$ of the mixture signal, as described in the beginning of Section III. Unless otherwise mentioned, the methods operate on the magnitude spectrogram. The following algorithms were tested.

- ISA. Implementation of the algorithm follows the outline proposed by Casey and Westner [5], but instead of aiming at statistically independent gains, we estimated statistically independent spectra, since that produced better results. Independent components were estimated using the FastICA algorithm [36], [37].
- NMF was tested with the algorithms proposed in [21]. These minimize the divergence or the Euclidean distance, and are denoted by NMF-DIV and NMF-EUC, respectively.
- The nonnegative sparse coding algorithm suggested by Abdallah and Plumbley [11] is denoted by NMF-LOG, since the method roughly minimizes the distance between the logarithm of the spectra. It assumes that the sources sum in the power spectral domain, so that the observation vector and basis functions in (1) are power spectra.

The proposed algorithm was evaluated by using different configurations. The weights α and β were not optimized for the test data, but different magnitudes $(1, 10, 100, \ldots,)$ were tried using similar test cases, and the values $\alpha = 100$ and $\beta = 0$ which produced approximately the best results were chosen. The effect of the weights is illustrated in the next section.

C. Evaluation

Each mixture signal was separated into components using all the algorithms. Since there is no reliable method for the estimation of the number of the components, all the methods were tested by factoring the mixture signal into 5, 10, 15, and 20 components, and the results were averaged.

Since the tested methods are blind, we do not know which component belongs to which source, and supervised clustering cannot be used. We tested the unsupervised clustering methods proposed in [5] and [6], trying to create component clusters for each source. However, these deteriorated the results in all the cases. Since manual clustering is too troublesome and unreliable, we had to use automatic clustering, where original signals before mixing are used as references for clusters.

The signal-to-noise ratio (SNR) between each component and source was found to be a good measure for assigning components to sources. To prevent the synthesis procedure from affecting the quality, the measure was calculated between the magnitude spectrograms $\mathbf{Y}_m$ and $\hat{\mathbf{Y}}_j$ of the mth reference and jth separated component, respectively,

$$\mathrm{SNR}(m,j) = \frac{\sum_{k,t} [\mathbf{Y}_m]_{k,t}^2}{\sum_{k,t} \left([\mathbf{Y}_m]_{k,t} - [\hat{\mathbf{Y}}_j]_{k,t}\right)^2}. \tag{19}$$

A component j is assigned to a source m which leads to the highest SNR.

TABLE II
SIMULATION RESULTS

algorithm	detection error rate (%)			SNR (dB)		
	all	pitched	drums	all	pitched	drums
ISA	31	29	33	3.6	4.4	1.9
NMF-EUC	28	28	30	6.6	7.9	3.7
NMF-DIV	26	28	23	7.0	8.8	3.5
NMF-LOG	80	90	57	2.3	2.7	2.2
proposed	**24**	**25**	**22**	**7.3**	**9.1**	**3.6**

A large number of components which are clustered using the original signals as references may produce unrealistically good results. For example, separating the mixture signal into components which had only one nonzero time-frequency point, and assigning the components to the sources using the original signals as references resulted in 0% detection error rate and 18.9-dB SNR, which are unattainable for the existing algorithms: there does not exist a clustering algorithm which could produce as good separated signals without using the original signals.

To overcome these problems, each source was assigned a single component with the largest SNR. This approach utilizes a minimum amount of prior information about the reference signals, but still produces applicable results.

The quality of the separated sources was measured by calculating the SNR between the original magnitude spectrogram $\mathbf{Y}$ and corresponding separated magnitude spectrogram $\hat{\mathbf{Y}}$ according to (19). The SNR has been used in several source separation studies to measure the separation quality; for example, Jang and Lee reported average SNR of about 9.6 for an ISA algorithm, in which the time-domain signals of the sources were trained before mixing [4]. The term *Source to Distortion Ratio* has also been used to refer to this performance measure [38].

The SNR (in decibels) was averaged over all the sources and mixtures to get the total measure of the separation performance. If no components were assigned to a source, the source was defined to be undetected. The detection error rate was defined as the ratio of the total number of undetected sources and the total number of sources. The undetected sources were not used in the calculation of the average SNR.

D. Results

The average SNRs and detection error rates are shown in Table II. The averages are shown for all sources, and separately for pitched and drum sounds. The 95% confidence intervals ([39], pp. 212–219) for the average detection error rate and the SNR were smaller than $\pm 1\%$ and ± 0.1 dB, respectively, for all the algorithms, which means that the differences between the algorithms are statistically significant.

Use of the temporal continuity term improves the detection of pitched sounds, and the proposed method also enables a slightly better SNR of pitched sources than NMF-DIV. If also undetected sources were included in the computing the SNR, the improvement would be even larger.

The nonnegative matrix factorization algorithms NMF-EUC and NMF-DIV produce clearly better results than ISA, and the performance of NMF-DIV is better than NMF-EUC. The performance of NMF-LOG is poor according to the detection error rate and SNR. These measures are derived from the energy of the error signal, but NMF-LOG is much more sensitive to log-energy observations. To investigate this further, we used also the likelihood measure proposed in [11] to evaluate the quality. The likelihood is based on a multiplicative noise model, and it results in a distortion measure $\sum_{k,t}([\mathbf{Y}_m]^2_{k,t}/[\hat{\mathbf{Y}}_j]^2_{k,t} - 1 + \log([\hat{\mathbf{Y}}_j]^2_{k,t}/[\mathbf{Y}_m]^2_{k,t}))$ between the original magnitude spectrogram $\mathbf{Y}_m$ and the separated magnitude spectrogram $\hat{\mathbf{Y}}_j$ (a small positive constant was added to both terms to avoid numerical problems in the divisions). The measure is more sensitive to low-energy observations than the SNR. The distortion measures were 0.30 (ISA), 2.38 (NMF-EUC), 1.94 (NMF-DIV), 5.08 (NMF-LOG), and 1.02 (proposed method), a smaller value indicating a better quality. This shows that the chosen performance measure has some effect on the results, since according to the this measure ISA gives the best quality, although the order of the other methods remains the same. Unlike ISA, the NMF-based algorithms do not allow subtraction of components, and therefore often produce values $[\hat{\mathbf{Y}}_j]_{k,t} \approx 0$, which results in a large distortion measure. The quality analysis in this paper is mainly based on the SNR, since it is more widely used.

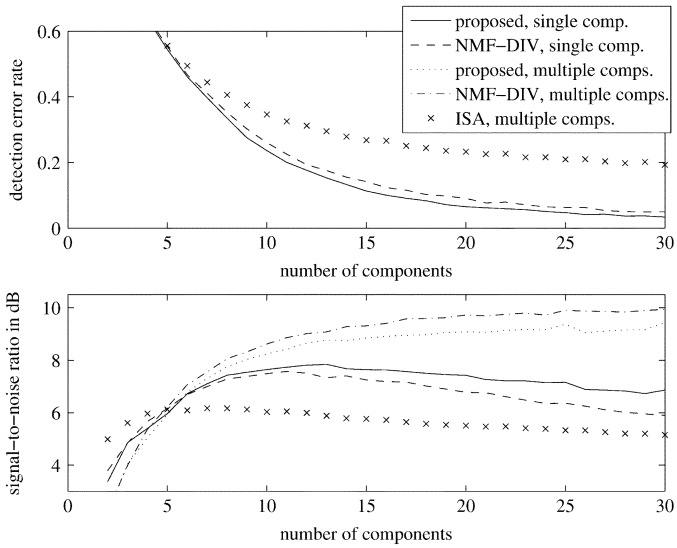

Fig. 3. Illustration of the effect of the component count. "Single comp." refers to measures where a single component was used to model each source, and "multiple comps." refers to measures where all the components were clustered using the original signals are references.

The effect of the weights α and β is illustrated in Fig. 2. The use of the temporal continuity term $(\alpha > 0)$ improves especially the detection of pitched instrument sources. The sparseness term $(\beta > 0)$ was not found to improve the results. When either value is too large, the quality of the separation degrades clearly.

Fig. 3 illustrates the performance of the proposed method, NMF-DIV, and ISA as a function of the number of components, separately for the cases where either a single component or all the components were clustered using the original signals as references. The latter case was included because the original idea of ISA is based on using multiple components per source. The detection error rate of all the algorithms approaches zero as the number of components increases. The proposed method enables a better detection error rate with all component counts.

Increasing the number of components increases the average SNR of the separated sources up to a certain point, after which it decreases in the cases for single-component algorithms, and saturates for multiple-component algorithms. When all the components are used, the asymptotic SNR of NMD-DIV is better than the proposed method. However, the SNR of both algorithms is limited, which suggests that nonnegativity restrictions alone are not sufficient for high-quality separation, but further assumptions such as harmonicity of the sources or a more flexible signal model might have to be used. The performance of the multiple-component ISA is clearly worse than the single-component NMF algorithms.

V. Conclusion

An algorithm for monaural sound source separation was presented. The existing algorithms based on the linear model for a spectrogram are limited in a sense that they consider each frame as an individual observation, even though natural sounds are often slowly-varying in time. The proposed cost function which is the sum of the squared differences between the gains in adjacent frames is a simple and efficient way of including the temporal continuity objective into this separation framework. The simulation experiments show that the temporal continuity criterion improves significantly the detection accuracy of pitched sounds and slightly their SNRs. The sparseness assumptions did not lead to significantly better detection accuracy or SNR.

The simulations show that in the spectrogram factorization framework the nonnegative matrix factorization algorithms produce better separation results than the independent component analysis. Even when all the components were clustered to source using the original signals as references, ISA could not achieve the performance of the NMF or the proposed method. Multiplicate update rules are efficient in the estimation of nonnegative parameters, since they do not require estimating a step size. When other cost terms are used in addition to the reconstruction error, the proposed updates are not guaranteed to always decrease the value of the cost function, but the simulations show that their performance suffices to minimize the cost function and to estimate the components in practice.

References

[1] M. Goto, "A predominant-f0 estimation method for real-world musical audio signals: MAP estimation for incorporating prior knowledge about f0s and tone models," in *Proc. Workshop Consistent Reliable Acoust. Cues for Sound Anal.*, 2001.

[2] G. Hu and D. L. Wang, "Monaural speech segregation based on pitch tracking and amplitude modulation," *IEEE Trans. Neural Netw.*, vol. 15, no. 5, pp. 1135–1150, Sep. 2004.

[3] S. Roweis, "One microphone source separation," in *Proc. Neural Inf. Proc. Syst. (NIPS)*, T. K. Leen, T. G. Dietterich, and V. Tresp, Eds., Denver, CO, 2000, pp. 793–799.

[4] G.-J. Jang and T.-W. Lee, "A maximum likelihood approach to single channel source separation," *J. Mach. Learn. Res.*, vol. 23, pp. 1365–1392, 2003.

[5] M. A. Casey and A. Westner, "Separation of mixed audio sources by independent subspace analysis," in *Proc. Int. Comp. Music Conf.*, Berlin, Germany, 2000, pp. 154–161.

[6] S. Dubnov, "Extracting sound objects by independent subspace analysis," in *Proc. 22nd Int. Audio Eng. Soc. Conf.*, Espoo, Finland, Jun. 2002.

[7] J. C. Brown and P. Smaragdis, "Independent component analysis for automatic note extraction from musical trills," *J. Acoust. Soc. Amer.*, vol. 115, pp. 2295–2306, May 2004.

[8] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *IEEE Workshop on Applications of Signal Process. Audio Acoust.*, New Paltz, NY, 2003, pp. 177–180.

[9] S. A. Abdallah and M. D. Plumbley, "An independent component analysis approach to automatic music transcription," in *Proc. Audio Eng. Soc. 114th Convention*, Amsterdam, The Netherlands, Mar. 2003.

[10] T. Virtanen, "Sound source separation using sparse coding with temporal continuity objective," in *Proc. Int. Comput. Music Conf.*, Singapore, 2003, pp. 231–234.

[11] S. A. Abdallah and M. D. Plumbley, "Polyphonic transcription by nonnegative sparse coding of power spectra," in *Int. Conf. Music Inf. Retrieval*, Barcelona, Spain, Oct. 2004, pp. 318–325.

[12] T. Blumensath and M. Davies, "Sparse and shift-invariant representations of music," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 1, pp. 50–57, Jan. 2006.

[13] P. Smaragdis, "Discovering auditory objects through nonnegativity constraints," in *Proc. ISCA Tutorial and Research Workshop Statistical Perceptual Audio Process.*, Jeju Island, Korea, 2004.

[14] T. Virtanen, "Separation of sound sources by convolutive sparse coding," in *Proc. ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio Process.*, Jeju Island, Korea, 2004.

[15] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. New York: Wiley, 2001.

[16] A. Hyvärinen and P. Hoyer, "Emergence of phase and shift invariant features by decomposition of natural images into independent feature subspaces," *Neural Comput.*, vol. 12, no. 7, pp. 1705–1720, 2000.

[17] I. Orife, "A rhythm analysis and decomposition tool based on independent subspace analysis," Masters thesis, Dartmouth College, Hanover, NH, 2001.

[18] D. FitzGerald, E. Coyle, and B. Lawlor, "Sub-band independent subspace analysis for drum transcription," in *Proc. Int. Conf. Digital Audio Effects*, Hamburg, Germany, 2002, pp. 65–69.

[19] C. Uhle, C. Dittmar, and T. Sporer, "Extraction of drum tracks from polyphonic music using independent subspace analysis," in *Proc. 4th Int. Symp. Independent Compon. Anal. Blind Signal Separation*, Nara, Japan, 2003, pp. 843–848.

[20] M. A. Casey, "MPEG-7 sound-recognition tools," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 6, pp. 737–747, Jun. 2001.

[21] D. D. Lee and H. S. Seung, "Algorithms for nonnegative matrix factorization," in *Neural Inf. Process. Syst.*, Denver, CO, 2001, pp. 556–562.

[22] B. A. Olshausen and D. F. Field, "Sparse coding with an overcomplete basis set: A strategy employed by VI?," *Vis. Res.*, vol. 37, pp. 3311–3325, 1997.

[23] L. Benaroya, F. Bimbot, L. McDonagh, and R. Gribonval, "Non negative sparse representation for Wiener based source separation with a single sensor," in *IEEE Int. Conf. Audio, Speech, Signal Process,*, Hong Kong, China, 2003, pp. 613–616.

[24] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *J. Mach. Learning Res.*, vol. 5, pp. 1457–1469, 2004.

[25] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA: Athena Scientific, 1995.

[26] D. FitzGerald, "Automatic drum transcription and source separation," Ph.D. dissertation, Dublin Inst. Technol., Dublin, Northern Ireland, 2004.

[27] S. Vembu and S. Baumann, "Separation of vocals from polyphonic audio recordings," in *Proc. Int. Conf. Music Inf. Retrieval*, London, U.K., 2005, pp. 337–344.

[28] M. Helén and T. Virtanen, "Separation of drums from polyphonic music using nonnegtive matrix factorization and support vector machine," in *Proc. Eur. Signal Process. Conf.*, Istanbul, Turkey, 2005.

[29] T. D. Rossing, *The Science of Sound*, 2nd ed. Reading, MA: Addison-Wesley, 1990.

[30] E. Vincent and X. Rodet, "Music transcription with ISA and HMM," in *Proc. 5th Int. Symp. Independent Compon. Anal. Blind Signal Separation*, Granada, Spain, 2004, pp. 1197–1204.

[31] D. Griffin and J. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Trans. Acoustics, Speech, Signal Process.*, vol. ASSP-32, no. 2, pp. 236–242, Apr. 1984.

[32] M. Slaney, D. Naar, and R. F. Lyon, "Auditory model inversion for sound separation," in *IEEE Int. Conf. Audio, Speech, Signal Process.*, Adelaide, Australia, 1994, pp. 77–80.

[33] F. Opolko and J. Wapnick, "McGill University master samples," McGill Univ., Montreal, QC, Canada, Tech. Rep., 1987.

[34] The University of Iowa Musical Instrument Samples Database [Online]. Available: http://theremin.music.uiowa.edu.

[35] "DFH Superior," 2003 [Online]. Available: http://www.toontrack.com/superior.shtml, Toontrack Music

[36] A. Hyvärinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Trans. Neural Netw.*, vol. 10, no. 3, pp. 626–634, May 1999.
[37] *FastICA package for MATLAB*, [Online]. Available: http://www.cis.hut.fi/projects/ica/fastica/
[38] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, no. 4, pp. 1462–1469, Jul. 2006.
[39] R. V. Hogg and A. T. Craig, *Introduction to Mathematical Statistics*, 4th ed. New York: Macmillan, 1989.

Tuomas Virtanen received the M.Sc. degree in information technology from the Tampere University of Technology (TUT), Tampere, Finland, in 2001. He is currently pursuing a postgraduate degree at the TUT Institute of Signal Processing, where he has been since 1999.

His research interests include audio signal processing, source separation, and machine learning algorithms.