Real-time detection of overlapping sound events with non-negative matrix factorization

Arnaud Dessein, Arshia Cont, Guillaume Lemaitre

Real-Time Detection of Overlapping Sound Events with Non-Negative Matrix Factorization

Arnaud Dessein, Arshia Cont, and Guillaume Lemaitre

STMS Lab (IRCAM, CNRS, UPMC, INRIA),
1 place Stravinsky, 75004 Paris, France
{dessein,cont,lemaitre}@ircam.fr

Abstract. In this paper, we investigate the problem of real-time detection of overlapping sound events by employing non-negative matrix factorization techniques. We consider a setup where audio streams arrive in real-time to the system and are decomposed onto a dictionary of event templates learned off-line prior to the decomposition. An important drawback of existing approaches in this context is the lack of controls on the decomposition. We propose and compare two provably convergent algorithms that address this issue, by controlling respectively the sparsity of the decomposition and the trade-off of the decomposition between the different frequency components. Sparsity regularization is considered in the framework of convex quadratic programming, while frequency compromise is introduced by employing the beta-divergence as a cost function. The two algorithms are evaluated on the multi-source detection tasks of polyphonic music transcription, drum transcription and environmental sound recognition. The obtained results show how the proposed approaches can improve detection in such applications, while maintaining low computational costs that are suitable for real-time.

Keywords: Real-time multi-source detection, overlapping sound events, non-negative matrix factorization, convex quadratic programming, sparsity regularization, beta-divergence, frequency compromise.

1 Introduction

This paper presents non-negative matrix factorization techniques for real-time detection of overlapping sound events.[1] In general terms, *non-negative matrix factorization* (NMF) is a technique for data analysis, where the observed data are supposed to be non-negative [1–3]. The main philosophy of NMF is to build up these observations in a constructive additive manner. Such assumptions are particularly interesting when negative values cannot be interpreted (e.g., pixel intensity for images, word occurrence for texts, magnitude spectrum for sounds).

[1] Additional material including sound files described in the paper are available on a companion website: imtr.ircam.fr/imtr/Real-Time_Multi-Source_Detection.

1.1 Motivations

The main goal of this paper is to devise a robust real-time system that processes rapidly the incoming audio stream and detects the presence of multiple sound events potentially corrupted by noise. Since several sound events may overlap when considering realistic situations, we cannot use single-source detection techniques such as a simple spectral template to audio stream correlation. Instead, we rely on NMF techniques that intrinsically allow to cope with the simultaneity of the detected sound events.

The sound events considered in this paper can be produced by various kinds of sound sources such as a polyphonic instrument (e.g., piano), instruments of a drum kit (e.g., snare), or environmental sounds (e.g., car horn). These sound sources are represented with a dictionary of event templates onto which the audio stream is decomposed incrementally as it unfolds in time. This general scheme is called *non-negative decomposition*, or *supervised non-negative matrix factorization*, and has been employed for audio signal processing in real-time as well as non real-time setups [4–9].

During the decomposition of a signal, the price to pay for the simplicity of a standard NMF scheme is the misuse of event templates to explain this signal. For the specific task of polyphonic music transcription, this amounts to common note insertions and substitutions such as octave or harmonic errors. The issue is almost as serious in a general pattern recognition setting where different classes of events are allowed to overlap in time with the presence of noise. In such realistic cases, providing controls on the decomposition can improve the detection of the different events. Yet in the literature, a few attention has been payed to providing such controls. To the best of our knowledge, we are only aware of [8, 9] where a control on the sparsity of the decomposition is provided.

In our context, controlling sparsity can help to reduce the space of plausible results and increase the economy of class usage during decomposition, thus improving generalization and robustness of the system. In most applications however, the user does not know in advance the sparsity of the solutions and cannot estimate it easily. Moreover, sparsity may also change along the signal. For example, in the problem of polyphonic music transcription, sparsity is highly correlated to the number of notes played simultaneously at each instant. The same interpretation holds in problems such as drum transcription and environmental sound detection, where the number of activated sources at a time is both unknown and variable.

In the system of [8, 9], sparsity is nonetheless considered as fixed over the whole signal, what is not a plausible assumption. We are thus interested in more adaptable techniques where the controls are flexible enough so as to fit the decomposition to the current dynamic of the signal. We are finally also interested in providing other controls than sparsity. In particular, we want to introduce a control on the frequency trade-off of the decomposition between the different frequency components. Such a control may help to better balance the consideration of certain important frequencies in the decomposition, such as partials for musical notes, and thus improve detection.

1.2 Contributions

To address the issues discussed above, we propose two computationally efficient algorithms that include flexible controls on the decomposition with convergence guarantees. These algorithms exhibit low computational costs, making them suitable for real-time constraints even on a common desktop computer. Our contributions in this context can be summarized as follows:

1. We formulate a non-negative decomposition problem with an explicit and flexible sparsity control by employing a technique from convex optimization, namely *convex quadratic programming* [10]. In the proposed formulation of the problem, sparsity is regularized through a penalty term added to the standard Euclidean cost function as in [11]. The derived algorithm implements a multiplicative update for non-negative quadratic programming introduced in [12]. This update provably converges to the global solution of the problem. Moreover, the corresponding scheme is in contrast to the system of [8,9] where sparsity is considered as fixed over the whole signal and where optimality of the solutions is not guaranteed.
2. We investigate the use of an information-theoretic divergence, called the *beta-divergence* [13,14], as a parametric cost function to control the trade-off of the decomposition between the different frequency components. This is in contrast to previous systems for non-negative decomposition which have either considered the Euclidean distance or the Kullback-Leibler divergence with no control on the frequency compromise of the decomposition. NMF with the beta-divergence has recently proved its relevancy for off-line applications in audio [15–19]. We adapt these approaches to a real-time setup and, based on recent theoretical results [20–22], we propose a multiplicative update to compute the corresponding non-negative decomposition with convergence guarantees. Preliminary work on this contribution was presented in [23] where the discussion lacks convergence guarantees.
3. The two proposed algorithms are applied to several paradigms of multi-source detection in real-time. We evaluate quantitatively our approach for polyphonic music transcription and obtain results that are comparable to off-line systems at the state-of-the-art. We also showcase applications such as drum transcription and environmental sound detection in complex auditory scenes, where explicitly controlling the decomposition becomes crucial.

1.3 Organization

The remainder of this paper is organized as follows. Section 2 introduces the related background on NMF. Section 3 depicts the general architecture of the proposed real-time system for detection of overlapping sound events. Section 4 focuses on our contributions to the formulation of a non-negative decomposition problem with a sparsity control, and provides a multiplicative update that provably converges to the global solution of this problem. Section 5 discusses our contributions in employing the beta-divergence as a cost function to control the

frequency compromise of the decomposition, and provides a multiplicative update to solve the corresponding problem with convergence guarantees. Section 6 reports a comparative quantitative evaluation of the two proposed algorithms for polyphonic music transcription, and also discusses the problems of drum transcription and environmental sound detection in complex auditory scenes to demonstrate the generality of the system as well as the importance of providing controls on the decomposition. Section 7 draws conclusions and provides perspectives for improvement of the proposed system.

1.4 Notations

In the sequel, uppercase bold letters denote matrices, $\mathbf{I}$ is the identity matrix, and $\mathbf{E}$ is the matrix full of ones. Lowercase bold letters denote vectors, and $\mathbf{e}$ is the vector full of ones. A bold zero $\mathbf{0}$ represents either a null matrix or vector. Lowercase plain letters such as n, m, r denote scalars. $\mathbb{R}_+$ and $\mathbb{R}_{++}$ denote respectively the sets of non-negative and of positive scalars. The element-wise multiplication and division between two matrices $\mathbf{A}$ and $\mathbf{B}$ are denoted respectively by a circled times $\mathbf{A} \otimes \mathbf{B}$ and a fraction bar $\mathbf{A}/\mathbf{B}$. The element-wise power p of $\mathbf{A}$ is denoted by $\mathbf{A}^{\cdot p}$, and the element-wise square-root of $\mathbf{A}$ can alternatively be denoted by $\sqrt{\mathbf{A}}$. Element-wise inequalities between $\mathbf{A}$ and $\mathbf{B}$ are simply written as $\mathbf{A} \leq \mathbf{B}$. The transpose of $\mathbf{A}$ is denoted by $\mathbf{A}^\top$. The non-negative matrices $\mathbf{A}^+$ and $\mathbf{A}^-$ denote respectively the positive and negative parts of $\mathbf{A}$ defined as follows:

$$a_{ij}^+ = \begin{cases} a_{ij} & \text{if } a_{ij} > 0 \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad a_{ij}^- = \begin{cases} -a_{ij} & \text{if } a_{ij} < 0 \\ 0 & \text{otherwise} \end{cases} . \tag{1}$$

2 Related Background

In this section, we first introduce the standard NMF problems and algorithms. We then provide an overview of NMF techniques in applications to audio event detection.

2.1 Non-Negative Matrix Factorization

The standard NMF model is a low-rank approximation technique for unsupervised multivariate data analysis. Given an $n \times m$ non-negative matrix $\mathbf{V}$ and a positive integer $r < \min(n, m)$, NMF tries to factorize $\mathbf{V}$ into an $n \times r$ non-negative matrix $\mathbf{W}$ and an $r \times m$ non-negative matrix $\mathbf{H}$ such that:

$$\mathbf{V} \approx \mathbf{W}\mathbf{H} . \tag{2}$$

The multivariate data to decompose are stacked into $\mathbf{V}$, whose columns represent the different observations, and whose rows represent the different variables. Each column $\mathbf{v}_j$ of $\mathbf{V}$ can then be expressed as $\mathbf{v}_j \approx \mathbf{W}\mathbf{h}_j = \sum_i h_{ij}\mathbf{w}_i$, where $\mathbf{w}_i$ and $\mathbf{h}_j$ are respectively the i-th column of $\mathbf{W}$ and the j-th column of $\mathbf{H}$. The columns

of $\mathbf{W}$ then form a *basis* and each column of $\mathbf{H}$ is the *decomposition* or *encoding* of the corresponding column of $\mathbf{V}$ into this basis.

As the model in (2) may provide an approximate factorization $\mathbf{\Lambda} = \mathbf{WH}$ of $\mathbf{V}$, the aim is to find a factorization that optimizes a given goodness-of-fit measure called *cost function*. For a given cost function $\mathcal{C}(\mathbf{V}, \mathbf{\Lambda})$ the corresponding NMF problem can thus be rewritten as a constrained optimization problem:

$$\underset{\mathbf{W}\in\mathbb{R}_+^{n\times r},\ \mathbf{H}\in\mathbb{R}_+^{r\times m}}{\arg\min} \mathcal{C}(\mathbf{V}, \mathbf{WH}) \ . \tag{3}$$

In the standard formulation, the Frobenius norm is used to define the following Euclidean cost function:

$$\mathcal{C}(\mathbf{V}, \mathbf{\Lambda}) = \frac{1}{2}\|\mathbf{V} - \mathbf{\Lambda}\|_{\mathrm{F}}^2 = \frac{1}{2}\sum_{i,\,j}\left(v_{ij} - \lambda_{ij}\right)^2 \ . \tag{4}$$

For this particular cost function, factors $\mathbf{W}$ and $\mathbf{H}$ can be computed with the popular *multiplicative updates* introduced in [2, 3]. These updates are derived from a gradient descent scheme with judiciously chosen adaptive steps as follows:

$$\mathbf{H} \leftarrow \mathbf{H} \otimes \frac{\mathbf{W}^\top\mathbf{V}}{\mathbf{W}^\top\mathbf{WH}} \quad \text{and} \quad \mathbf{W} \leftarrow \mathbf{W} \otimes \frac{\mathbf{VH}^\top}{\mathbf{WHH}^\top} \ . \tag{5}$$

The respective updates are applied in turn until convergence, and ensure both non-negativity of the factors $\mathbf{W}$ and $\mathbf{H}$ as well as monotonic decrease of the cost, but not necessarily convergence of the factors nor local optimality.

A flourishing literature also exists about other algorithms and extensions to the standard NMF problem [24,25]. These extensions can be thought of in terms of modified models (e.g., using tensors), of modified constraints (e.g., imposing the sparsity of the factors), and of modified cost functions (e.g., using divergences or adding penalty terms).

For example, the standard Euclidean cost function is often replaced with the Kullback-Leibler divergence:

$$\mathcal{C}(\mathbf{V}, \mathbf{\Lambda}) = \mathcal{D}_{\mathrm{KL}}(\mathbf{V}|\mathbf{\Lambda}) = \sum_{i,\,j} v_{ij}\log\frac{v_{ij}}{\lambda_{ij}} + \lambda_{ij} - v_{ij} \ , \tag{6}$$

for which specific multiplicative updates have also been derived [2, 3]:

$$\mathbf{H} \leftarrow \mathbf{H} \otimes \frac{\mathbf{W}^\top\left(\mathbf{V} \otimes (\mathbf{WH})^{\cdot -1}\right)}{\mathbf{W}^\top\mathbf{E}} \quad \text{and} \quad \mathbf{W} \leftarrow \mathbf{W} \otimes \frac{\left(\mathbf{V} \otimes (\mathbf{WH})^{\cdot -1}\right)\mathbf{H}^\top}{\mathbf{EH}^\top} \ . \tag{7}$$

These updates again ensure non-negativity of the factors $\mathbf{W}$ and $\mathbf{H}$ and monotonic decrease of the cost, but not necessarily convergence nor local optimality of the factors.

2.2 Applications to the Detection of Overlapping Sound Events

NMF algorithms have been applied to various problems in computer vision, signal processing, biomedical data analysis and text classification among others [25]. In the context of sound processing, the matrix $\mathbf{V}$ is in general a time-frequency representation of the sound to analyze. The rows and columns represent respectively different frequency bins and successive time-frames. The factorization $\mathbf{v}_j \approx \sum_i h_{ij}\mathbf{w}_i$ can then be interpreted as follows: each basis vector $\mathbf{w}_i$ contains a spectral template, and the decomposition coefficients h_{ij} represent the activations of the i-th template at the j-th time-frame.

Concerning the detection of overlapping sound events, NMF has been widely used in off-line systems for polyphonic music transcription, where the sound events correspond roughly to notes (e.g., see [26,27]). Several problem-dependent extensions have been developed to provide controls on NMF in this context, such as a source-filter model [28], an harmonic constraint [29], a selective sparsity regularization [30], or a subspace model of basis instruments [31]. Most of these systems consider either the standard Euclidean cost or the Kullback-Leibler divergence. Recent works yet have investigated the use of other cost functions such as the Itakura-Saito divergence [32–35] or the more general parametric beta-divergence [17].

Some authors have also used non-negative decomposition for sound event detection. A real-time system to identify the presence and determine the pitch of one or more voices is proposed in [4] and is adapted to sight-reading evaluation of solo instrument in [5]. Concerning automatic transcription, off-line systems are used in [6] for drum transcription and in [7] for polyphonic music transcription. A real-time system for polyphonic music transcription is also proposed in [8] and is further developed in [9] for real-time coupled multiple-pitch and multiple-instrument recognition. All these systems consider either the Euclidean or the Kullback-Leibler cost function, and only the latter provides a control on the decomposition by enforcing the solutions to have a fixed desired sparsity.

Other approaches in the framework of probabilistic models with latent variables also share common perspectives with NMF techniques [36]. In this framework, the non-negative data are considered as a discrete distribution and are factorized into a mixture model where each latent component represents a source. It can then be shown that maximum likelihood estimation of the mixture parameters amounts to NMF with the Kullback-Leibler divergence, and that the classical expectation-maximization algorithm is equivalent to the multiplicative updates scheme. Considering the problem in a probabilistic framework is however convenient for enhancing the standard model and adding regularization terms through priors and maximum a posteriori estimation instead of maximum likelihood estimation. In particular, the framework has been employed in polyphonic music transcription to include shift-invariance and sparsity [37]. Recent works have extended the later model to include a temporal smoothing and a unimodal prior for the impulse distributions [38], a hierarchical subspace model for instrument families [39], a scale-invariance [40], a time-varying harmonic structure [41], and multiple spectral templates [42].

3 General Architecture of the System

In this section, we present the system proposed for real-time detection of overlapping sound events. The general architecture of the system is shown schematically in Fig. 1. The right side of the figure represents the audio signal arriving in real-time, and its decomposition onto sound events whose descriptions are provided a priori to the system as a dictionary of sound event templates as in [4–9]. These event templates are learned off-line prior to the decomposition as shown on the left side of the figure. We describe the two general modules hereafter.

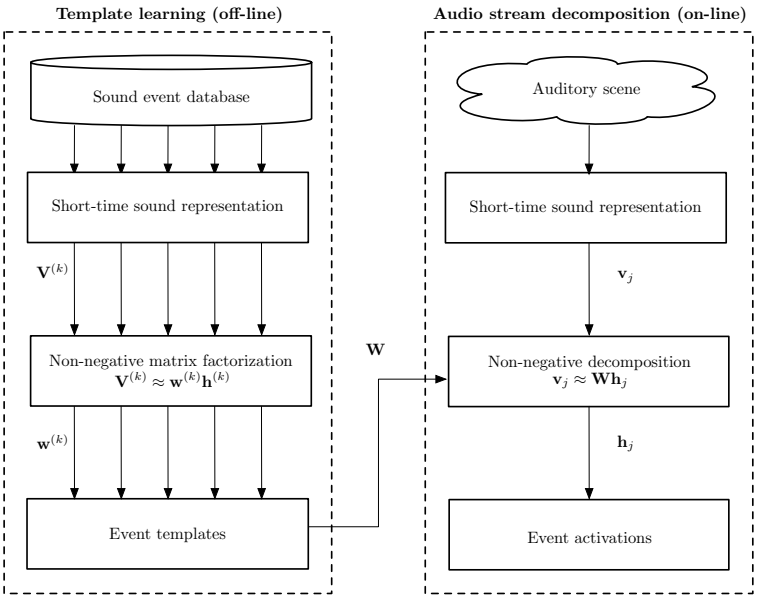


Fig. 1. Schematic view of the general architecture of the proposed system.

3.1 Template Learning

The learning module aims at building a dictionary $\mathbf{W}$ with characteristic and discriminative templates of the sound events to detect. In the literature, the event templates are generally learned off-line with NMF, the differences between the approaches being in the way NMF is formulated.

In the present work, we use a simple rank-one NMF with the standard Euclidean cost function as a learning scheme. We suppose that the user possesses a sound database of isolated exemplars of the events to detect, from which the system learns desired characteristic templates. The whole sound exemplar k is first processed in a short-time sound representation supposed to be non-negative and

approximatively additive (e.g., short-time magnitude spectrum). The representations are stacked in a matrix $\mathbf{V}^{(k)}$ where each column $\mathbf{v}_j^{(k)}$ is the representation of the j-th time-frame. We then solve standard NMF with $\mathbf{V}^{(k)}$ and a rank of factorization $r = 1$, by employing the standard multiplicative updates in (5) with a max-normalization of the activations along time. This learning scheme simply gives a sound event template in the column vector $\mathbf{w}^{(k)}$ for each exemplar (the information in the row vector $\mathbf{h}^{(k)}$ is then discarded).

3.2 Audio Stream Decomposition

Having learned the templates, we construct a dictionary $\mathbf{W}$ where all templates $\mathbf{w}^{(k)}$ are stacked in columns. The problem of real-time decomposition of an audio stream then amounts to projecting the incoming signal $\mathbf{v}_j$ onto $\mathbf{W}$, where $\mathbf{v}_j$ share the same representational front-end as the templates. The problem is thus equivalent to a non-negative decomposition $\mathbf{v}_j \approx \mathbf{W}\mathbf{h}_j$ where $\mathbf{W}$ is kept fixed and only $\mathbf{h}_j$ is learned. The learned vectors $\mathbf{h}_j$ would then provide the activations of the different sound events potentially present in the auditory scene.

As such, the system reports only a frame-level activity of the different events. Depending on the final application, some post-processing is thus needed to extract more information about the presence of each sound source at the frame level or at a longer-term level (e.g., activation thresholding, onset detection, temporal modeling or smoothing). This application-dependent processing is not thoroughly discussed in this paper; we rather focus on providing flexible controls on the decomposition.

In the literature, the non-negative decomposition is performed either with the Euclidean or the Kullback-Leibler cost functions in (4) and (6). Also, there is in general no control on the decomposition, except from the system in [8, 9] where the sparsity of the solutions is regularized but considered as fixed over the whole signal. In the next two sections, we discuss the two independent approaches we investigated to provide flexible controls on the decomposition. In Sect. 4, we first focus on controlling the sparsity of the decomposition in a flexible way by employing the Euclidean cost function and the framework of convex quadratic programming. In Sect. 5, we then address the use of the information-geometric beta-divergence as a parametric cost function to control the frequency compromise during decomposition. To simplify the notations, we restrict without lack of generality to the case where there is only one vector $\mathbf{v}$ to decompose as $\mathbf{v} \approx \mathbf{W}\mathbf{h}$.

4 Non-Negative Decomposition and Sparsity Regularization with Convex Quadratic Programming

In this section, we first review the notion of sparsity and its use in combination with NMF. We then formulate a non-negative decomposition with explicit and flexible sparsity regularization within the framework of convex quadratic programming and provide a provably convergent multiplicative update to perform the real-time decomposition.

4.1 Definition and Measures of Sparsity

The simplest definition of *sparsity*, or *sparseness*, is that a vector is *sparse* when most of its elements are null. The sparsity measure that corresponds to this definition is based on the ℓ_0-norm and just counts the number of non-null coefficients of this vector. However, it is only applicable in noiseless situations and alternative definitions and measures have been proposed in the literature to cope with realistic scenarios [43]. The idea is that a vector is sparse when it is not dense, i.e., much of its energy is packed into a few components.

In practice, the ℓ_p-norms for $0 < p \leq 1$ are often used directly to measure sparsity. In the context of NMF, another sparsity measure has also been introduced in [44]:

$$\mathrm{sp}(\mathbf{x}) = \frac{\sqrt{n} - \|\mathbf{x}\|_1 / \|\mathbf{x}\|_2}{\sqrt{n} - 1} \ , \tag{8}$$

where n is the length of the vector $\mathbf{x}$. This measure increases as $\mathbf{x}$ becomes sparser and is scale-independent. It is comprised between 0 for any vector with all components equal up to the signs, and 1 for any vector with a single non-null component, interpolating smoothly between the two bounds.

4.2 Non-Negative Matrix Factorization and Sparsity

In the standard NMF formulation, the sparsity of solutions is implicit. Explicit control of sparsity becomes however crucial in certain situations, and several NMF extensions have been proposed to this end. For example, a sparsity penalty is employed in [45] and the problem is solved using ad hoc multiplicative updates. In [46], a penalty term also regularizes sparsity, and the problem is solved with a modified alternating least squares algorithm. From a theoretical viewpoint however, these schemes do not guarantee convergence nor monotonic decrease of the cost in general, what is undesirable to design a robust real-time system.

More rigorous frameworks have been considered in [47] where the proposed algorithm uses provably convergent multiplicative updates for the ℓ_1-penalized factor, and projected gradient to ensure non-negativity and ℓ_1-normalization of the other factor. In [44], the sparsity measure in (8) is introduced and projected gradient is used to enforce a fixed desired sparsity s on solutions, yet the choice of a fixed sparsity s remains an important issue. These schemes ensure both cost decrease and convergence but not necessarily local optimality of solutions.

To achieve flexible sparsity, a second-order cone programming framework is proposed in [48] to introduce min and max-sparsity bound constraints and give the user more flexibility than fixing a priori the sparsity s of solutions. In [11], the framework of convex quadratic programming is used to penalize less sparse solutions with a ℓ_1-norm. Such approaches can not only help to prove cost decrease and convergence but also local optimality of solutions.

In the context of audio analysis, some authors have already considered sparsity controls in off-line setups for speech analysis [15], polyphonic music transcription [26], and source separation [49]. In these works, sparse coding is introduced by means of penalties and solved using multiplicative updates with

no guaranteed convergence.[2] Concerning real-time setups, the system in [8,9] is the only one to consider sparsity controls. However, sparsity in this system is controlled as in [44] by projection onto a non-adaptable fixed sparsity s chosen a priori by the user. Moreover, the authors slightly modify the scheme proposed in [44], which breaks down its geometric interpretation and falsifies the projection scheme.

From a theoretical viewpoint, the approaches proposed in [11] and [48] are the most interesting since they not only provide a flexible control on sparsity, but also guarantee monotonic decrease of the cost, convergence guarantees, and local optimality of solutions. From a practical viewpoint, the convex quadratic scheme of [11] is computationally more attractive than the scheme of [48]. Indeed, the latter scheme requires solving an expensive sequence of second-order cone programs, which can become problematic when the number of templates increases. Moreover, a sparsity penalty as in [11] reveals more convenient than sparsity bounds as in [48] for robustness issues in problems involving background noise. We thus focus on formulating a non-negative decomposition problem with a sparsity penalty term in a convex quadratic programming framework similar to that of [11].

4.3 Problem Formulation and Multiplicative Update

Let us first recall the notion of convex quadratic program. A *convex quadratic program* (CQP) is a constrained convex optimization problem that can be expressed in the following form:

$$\min_{\mathbf{x}} \ \frac{1}{2}\mathbf{x}^\top \mathbf{P}\mathbf{x} + \mathbf{q}^\top \mathbf{x} \quad \text{s.t.} \quad \mathbf{A}\mathbf{x} \leq \mathbf{b} \ , \tag{9}$$

where $\mathbf{P}$ is supposed to be a positive-semidefinite matrix [10].

This general form of optimization problem is interesting in our context since a non-negative decomposition problem with sparsity regularization can be formulated in a CQP form as follows:

$$\min_{\mathbf{h}} \ \frac{1}{2}\mathbf{h}^\top(\mathbf{W}^\top\mathbf{W} + \lambda_2\mathbf{I})\mathbf{h} + (\lambda_1\mathbf{e} - \mathbf{W}^\top\mathbf{v})^\top\mathbf{h} \quad \text{s.t.} \quad -\mathbf{I}\,\mathbf{h} \leq \mathbf{0} \ , \tag{10}$$

where $\lambda_1, \lambda_2 \geq 0$ are regularization parameters. Indeed, this CQP is equivalent to the following regularized non-negative least squares problem:

$$\operatorname*{arg\,min}_{\mathbf{h}\in\mathbb{R}_+^r} \ \frac{1}{2}\|\mathbf{v} - \mathbf{W}\mathbf{h}\|_2^2 + \lambda_1\|\mathbf{h}\|_1 + \frac{\lambda_2}{2}\|\mathbf{h}\|_2^2 \ . \tag{11}$$

The ℓ_1-norm penalizes less sparse vectors, and the ℓ_2-norm is a particular case of *Tikhonov regularization* which is often used in CQP because it makes the matrix

[2] Recent work presented in [21] may however prove a posteriori the cost monotonicity for certain heuristic multiplicative updates with sparsity penalty.

$\mathbf{P} = \mathbf{W}^\top\mathbf{W} + \lambda_2\mathbf{I}$ positive-definite at least for any $\lambda_2 > 0$ and thus makes the problem strictly convex [10].

To the best of our knowledge, although similar formulations have been considered to introduce sparsity penalization in constrained least-squares and NMF problems (e.g., see [11]), there is no such formulation for non-negative decomposition of audio signals. We are only aware of the system proposed in [8,9] which addresses sparsity regularization by different means as discussed previously.

To solve the problem formulated in (11), we propose to update $\mathbf{h}$ iteratively by using a multiplicative update developed in [12] for the specific case of non-negative quadratic programs, i.e., for CQPs where $\mathbf{A} = -\mathbf{I}$ and $\mathbf{b} = \mathbf{0}$ in (9). For a general non-negative quadratic program, the multiplicative update of [12] takes the following form:

$$\mathbf{x} \leftarrow \mathbf{x} \otimes \frac{-\mathbf{q} + \sqrt{\mathbf{q}.^2 + 4(\mathbf{P}^+\mathbf{x})(\mathbf{P}^-\mathbf{x})}}{2\mathbf{P}^+\mathbf{x}} \ , \tag{12}$$

and is proved to make the cost decrease and to converge to the global solution as soon as $\mathbf{P}$ is positive-definite, $\mathbf{x}$ is initialized with positive values, and the problem is non-degenerate. The problem becomes degenerate when there exists a positive vector $\mathbf{x}$ and a row i such that the update sets x_i to zero. Such a case can happen only when $q_i \geq 0$ and the i-th row of $\mathbf{P}$ is non-negative. In this situation however, the problem reduces to a smaller problem since the global solution has its i-th coefficient equal to zero. As a result, if the problem is degenerate, it suffices to solve the corresponding non-degenerate reduced problem, and then insert back the zero coefficients in the solution as discussed in [12]. We now apply this framework to our specific problem.

Let us first discuss the case when a degeneracy occurs. Since $\mathbf{P} = \mathbf{W}^\top\mathbf{W}+\lambda_2\mathbf{I}$ is non-negative, all rows of $\mathbf{P}$ are non-negative and the problem is degenerate as soon as any coefficient q_i of $\mathbf{q} = \lambda_1\mathbf{e} - \mathbf{W}^\top\mathbf{v}$ is non-negative. The vector $\mathbf{W}^\top\mathbf{v}$ being non-negative, this may occur only when λ_1 is sufficiently large, meaning that non-sparse vectors $\mathbf{x} = \mathbf{h}$ are highly penalized. In this situation, the degeneracy implies that the global solution has its i-th coefficient equal to zero, which is consistent with the high penalty on non-sparse vectors.

We now assume without lack of generality that the considered problem is non-degenerate, so that $-\mathbf{q} = \mathbf{W}^\top\mathbf{v} - \lambda_1\mathbf{e}$ is positive. The right term of the update can then be developed as follows:

$$\frac{-\mathbf{q} + \sqrt{\mathbf{q}.^2 + 4(\mathbf{P}^+\mathbf{x})(\mathbf{P}^-\mathbf{x})}}{2\mathbf{P}^+\mathbf{x}} = \frac{-\mathbf{q} + \sqrt{\mathbf{q}.^2 + 4(\mathbf{P}\mathbf{x})(\mathbf{0}\mathbf{x})}}{2\mathbf{P}\mathbf{x}} = \frac{-\mathbf{q}}{\mathbf{P}\mathbf{x}} \ . \tag{13}$$

This leads to the following specific multiplicative update:

$$\mathbf{h} \leftarrow \mathbf{h} \otimes \frac{\mathbf{W}^\top\mathbf{v} - \lambda_1\mathbf{e}}{(\mathbf{W}^\top\mathbf{W} + \lambda_2\mathbf{I})\mathbf{h}} \ , \tag{14}$$

which ensures positivity of $\mathbf{h}$, monotonic decrease of the cost and convergence to the global solution, as soon as $\mathbf{W}^\top\mathbf{W} + \lambda_2\mathbf{I}$ is positive-definite and $\mathbf{h}$ is

initialized with positive values. Remark that these conditions are not restrictive since $\mathbf{W}^\top\mathbf{W} + \lambda_2\mathbf{I}$ is positive-definite at least for any $\lambda_2 > 0$.

Concerning parameters, we thus just use λ_2 to ensure positive-definiteness. If $\mathbf{W}^\top\mathbf{W}$ is positive-definite, which is equivalent to $\mathbf{W}$ begin invertible, we simply set λ_2 equal to zero. Otherwise, $\mathbf{W}^\top\mathbf{W}$ is only positive-semidefinite and we set λ_2 equal to a small constant $\varepsilon > 0$. The user therefore needs only to tune the sparsity parameter $\lambda_1 \geq 0$.

In the implementation, we can take advantage of $\mathbf{W}$ being fixed to reduce the computational cost by computing $\mathbf{W}^\top\mathbf{W} + \lambda_2\mathbf{I}$ off-line prior to the decomposition, as well as $\mathbf{W}^\top\mathbf{v} - \lambda_1\mathbf{e}$ on-line but only once per time-frame. The update then becomes computationally cheap since it just amounts to computing one matrix-vector multiplication, one element-wise vector multiplication and one element-wise vector division per iteration. Moreover, the problem reduction to a non-degenerate form, which requires simple inequality checks once per time-frame, reduces the dimensionality and thus the computational cost of the update. Finally, $\mathbf{h}$ can be initialized with the output solution of the previous frame, after carefully replacing the zero coefficients with small positive values, which in general greatly speeds up convergence. This makes the proposed scheme suitable for real-time setups.

5 Non-Negative Decomposition and Frequency Compromise with the Beta-Divergence

In this section, we define the parametric beta-divergence and give some of its properties. We then review its use as a cost function for NMF and explain how it provides a flexible control on the compromise of decomposition between the different frequency components. We finally formulate the non-negative decomposition problem with the beta-divergence as a cost function and derive a tailored multiplicative update with convergence guarantees to solve it.

5.1 Definition and Properties of the Beta-Divergence

The beta-divergences form a parametric family of information-theoretic contrast functions [13, 14]. For any $\beta \in \mathbb{R}$ and any points $x, y \in \mathbb{R}_{++}$, the *β-divergence* from x to y can be defined as follows:

$$d_\beta(x|y) = \frac{1}{\beta(\beta-1)}\big(x^\beta + (\beta-1)y^\beta - \beta x y^{\beta-1}\big) \ . \tag{15}$$

As special cases when $\beta = 0$ and $\beta = 1$, taking the limits in the above definition leads respectively to the well-known Itakura-Saito and Kullback-Leibler divergences:

$$d_{\beta=0}(x|y) = d_{\mathrm{IS}}(x|y) = \frac{x}{y} - \log\frac{x}{y} - 1 \ , \tag{16}$$

$$d_{\beta=1}(x|y) = d_{\mathrm{KL}}(x|y) = x\log\frac{x}{y} + y - x \ , \tag{17}$$

while for $\beta = 2$, the definition specializes to the half squared Euclidean distance:

$$d_{\beta=2}(x|y) = d_{\mathrm{E}}(x|y) = \frac{1}{2}(x-y)^2 \ . \tag{18}$$

Concerning its properties, the β-divergence is non-negative and vanishes iff $x = y$. However, it is not necessarily a distance in the strict sense since it is not symmetric and does not satisfy the triangle inequality in general. A property of the β-divergence relevant to the present work is that for any scaling factor $\lambda \in \mathbb{R}_{++}$ we have:

$$d_\beta(\lambda x|\lambda y) = \lambda^\beta d_\beta(x|y) \ . \tag{19}$$

We discuss further the interest of this scaling property for the decomposition of audio signals in the following.

5.2 Non-Negative Matrix Factorization and the Beta-Divergence

The beta-divergence was first used with NMF to interpolate between the Euclidean distance and the Kullback-Leibler divergence [50]. Starting with the scalar divergence in (15), a matrix divergence can be constructed as a *separable* divergence, i.e., by summing the element-wise divergences as follows:

$$\mathcal{D}_\beta(\mathbf{V}|\mathbf{\Lambda}) = \sum_{i,\,j} d_\beta(v_{ij} \,|\, \lambda_{ij}) \ . \tag{20}$$

The NMF problem with the β-divergence then amounts to solving the following constrained optimization problem:

$$\underset{\mathbf{W}\in\mathbb{R}_+^{n\times r},\ \mathbf{H}\in\mathbb{R}_+^{r\times m}}{\arg\min} \mathcal{D}_\beta(\mathbf{V}|\mathbf{WH}) \ . \tag{21}$$

As for standard NMF, several algorithms including multiplicative updates have been derived to solve NMF with the beta-divergence and its extensions [25, 50]. The heuristic multiplicative updates take the following form:

$$\mathbf{H} \leftarrow \mathbf{H} \otimes \frac{\mathbf{W}^\top\big(\mathbf{V} \otimes (\mathbf{WH})^{\cdot\beta-2}\big)}{\mathbf{W}^\top(\mathbf{WH})^{\cdot\beta-1}} \quad \text{and} \quad \mathbf{W} \leftarrow \mathbf{W} \otimes \frac{\big(\mathbf{V} \otimes (\mathbf{WH})^{\cdot\beta-2}\big)\mathbf{H}^\top}{(\mathbf{WH})^{\cdot\beta-1}\mathbf{H}^\top} \ . \tag{22}$$

For $\beta = 1$ and $\beta = 2$, the problem and multiplicative updates specialize respectively to that of Euclidean and Kullback-Leibler NMF. However, even if these updates are proved to make the cost decrease monotonically for $0 \leq \beta \leq 2$, it may not be the case systematically for other values of β [21].

Modified updates that guarantee the monotonic decrease of the cost for any $\beta \in \mathbb{R}$ have been proposed recently in [20, 21] where an exponent step size p depending on β is introduced for the right term of the factor updates:

$$p(\beta) = \begin{cases} 1/(2-\beta) & \text{if } \beta < 1 \\ 1 & \text{if } 1 \leq \beta \leq 2 \\ 1/(\beta-1) & \text{if } \beta > 2 \end{cases} \ . \tag{23}$$

The updates modified with the exponent step size $p(\beta)$ guarantee the convergence of the cost, but not necessarily convergence of the factors nor local optimality. However, the convergence of the cost function ensures that the sequence of vectors $\mathbf{h}$ always improves the reconstruction with respect to the β-divergence, thus limiting unstable situations that are undesirable in practice. To the best of our knowledge, however, there is no NMF system with the beta-divergence that exploits this result; the algorithms in general have no convergence guarantees at all, and even no monotonic decrease of the cost function.[3] We also notice that $p(\beta) \leq 1$ for any $\beta \in \mathbb{R}$ with equality iff $1 \leq \beta \leq 2$. As a result, we may take a unit exponent step size $p(\beta)$ for $0 \leq \beta < 1$, corresponding to the heuristic updates, without compromising the cost monotonicity. This is akin to over-relaxation and produces larger steps, thus reducing their number and fastening convergence.

Concerning its applications, NMF with the beta-divergence has proved its relevancy for audio off-line systems in speech analysis [15], source separation [16], music transcription [17], and non-stationarity modeling with a parametric model of the spectral templates [18] or a source-filter model for time-varying activations [19]. The scaling property in (19) may give an insight in understanding the relevancy of the beta-divergence in this context.

As remarked in [32], the Itakura-Saito divergence for $\beta = 0$ is the only β-divergence to be scale-invariant. This means that the corresponding NMF problem gives the same relative weight to all coefficients, and thus penalizes equally a bad fit of factorization for small and large coefficients. For other values of β, however, the scaling property implies that a different emphasis is put on the coefficients depending on their magnitude. When $\beta > 0$, more emphasis is put on the higher magnitude coefficients, and the emphasis augments with β. When $\beta < 0$, the effect is the converse.

Considering audio signals, this amounts to giving different importance to high and low-energy frequency components. In a context of polyphonic music decomposition, we try to reconstruct an incoming signal by addition of note templates. In order to avoid common octave and harmonic errors, a good reconstruction would have to find a compromise between focusing on the fundamental frequency, the first partials and higher partials. This compromise should also be achieved in an adaptable way, independent of the fundamental frequency, similarly to a compression rather than a global weighting of the different components. The parameter β can thus help to control this trade-off. A similar interpretation holds in a general audio decomposition problem where the decomposition should find a compromise between the high and low-energy frequency components.

Last but not least, we notice that in the literature, there is in general no rigorous consideration on the domain of the β-divergence which is usually defined for any $\beta \in \mathbb{R}$ as in (15) but for any $x, y \in \mathbb{R}_+$ instead of $\mathbb{R}_{++}$. This is nonetheless only possible for $\beta > 1$ so that the problem in (21) is not actually rigorously posed for $\beta \leq 1$. Moreover, even when $\beta > 1$, attention must be paid

[3] Results in [21] may again prove a posteriori the cost monotonicity for certain heuristic multiplicative updates employed in the literature.

in the multiplicative updates as soon as zero values are allowed. In the best case, a zero value in a factor remains zero as it is updated, but null coefficients may also introduce divisions by zero. As a result, most of the nice properties such as monotonic decrease of the cost and convergence may break down and algorithms may become unstable as soon as zero values are allowed. Such considerations are important for a real-time application where a stable behavior with no unpredictable errors is mandatory. We thus try in the following to formulate the problem of non-negative decomposition by taking care of such considerations.

5.3 Problem Formulation and Multiplicative Update

The non-negative decomposition with the β-divergence as a cost function is equivalent to the following constrained optimization problem:

$$\underset{\mathbf{h}\in\mathbb{R}_+^r}{\arg\min}\ \mathcal{D}_\beta(\mathbf{v}|\mathbf{W}\mathbf{h})\ . \tag{24}$$

We emphasize again that this problem is not rigorously defined when $\beta \leq 1$. Considering such technical distinctions is not the sake of this paper; our aim is rather to develop a generic scheme with a single simple algorithm that works for any $\beta \in \mathbb{R}$. We will see that assuming a few hypotheses on the problem, we can define properly such a scheme with convergence guarantees.

We first clearly need to assume that $\mathbf{v}$ is positive. For the moment, we put no other restriction than non-negativity on $\mathbf{W}$, and we propose to solve the problem by initializing $\mathbf{h}$ with positive values and by updating $\mathbf{h}$ iteratively with a vector version of its respective update proposed in [20, 21] as follows:

$$\mathbf{h} \leftarrow \mathbf{h} \otimes \left(\frac{\mathbf{W}^\top \left(\mathbf{v} \otimes (\mathbf{W}\mathbf{h})^{\cdot\beta-2} \right)}{\mathbf{W}^\top (\mathbf{W}\mathbf{h})^{\cdot\beta-1}} \right)^{\cdot p(\beta)} . \tag{25}$$

This scheme ensures monotonic decrease and convergence of the cost function as long as $\mathbf{h}$ and $\mathbf{W}^\top(\mathbf{W}\mathbf{h})^{\cdot\beta-1}$ stay positive.

These conditions are clearly equivalent to $\mathbf{W}^\top(\mathbf{W}\mathbf{h})$ staying positive which is in turn equivalent to $\mathbf{W}$ having no null row and no null column. The case of a null row is not interesting in practice since the problem becomes degenerate, implying that we can remove the corresponding rows of $\mathbf{v}$ and $\mathbf{W}$. The case of a null column is also uninteresting since it corresponds to one of the event templates being null, implying that the problem is degenerate so that we can remove this column of $\mathbf{W}$ and the corresponding coefficient of $\mathbf{h}$. We thus suppose without lack of generality that $\mathbf{W}$ has no null row or column, hence the updates guarantee monotonic decrease and convergence of the cost function.

To sum up the required assumptions, we suppose that null rows of $\mathbf{W}$ and corresponding rows of $\mathbf{v}$ have been removed, null columns of $\mathbf{W}$ and corresponding coefficients of $\mathbf{h}$ have been removed, $\mathbf{v}$ is positive, and $\mathbf{h}$ is initialized with positive values. These assumptions allow to unify the proposed approach in a single algorithm with guaranteed convergence of the cost, and a unique parameter $\beta \in \mathbb{R}$ to be tuned by the user. Moreover, the only restrictive assumption

is that of $\mathbf{v}$ being positive, which can be achieved either by pre-whitening or by simply setting the zero coefficients to small values $\varepsilon > 0$.

Under the same assumptions, we can also obtain that for any $1 \leq \beta \leq 2$, the sequence of vectors $\mathbf{h}$ converges to a locally optimal solution. This result is based on boundedness of the sequence, as well as recent theoretical advances on the stability of multiplicative updates [22]. For the sake of conciseness, we yet do not develop this discussion further since we were not able to generalize the result. This is because the upper bound on the exponent step size provided in [22] is still unknown and may be local for other values of β, while it is global and equal to $2 > p(\beta)$ for any $1 \leq \beta \leq 2$. Moreover, boundedness may also break down for $\beta < 0$ because of finite limit of the β-divergence at infinity in the second argument (yet practical values of interest are $\beta \geq 0$).

Concerning implementation, we can take advantage of $\mathbf{W}$ being fixed to employ a multiplicative update tailored to real-time. Indeed, after some matrix manipulations, we can rewrite the update in (25) as follows:

$$\mathbf{h} \leftarrow \mathbf{h} \otimes \left(\frac{\big(\mathbf{W} \otimes (\mathbf{v}\mathbf{e}^\top)\big)^\top (\mathbf{W}\mathbf{h})^{\cdot\beta-2}}{\mathbf{W}^\top \big((\mathbf{W}\mathbf{h}) \otimes (\mathbf{W}\mathbf{h})^{\cdot\beta-2}\big)} \right)^{\cdot p(\beta)} . \tag{26}$$

This helps reduce the computational cost of the update since $\big(\mathbf{W} \otimes (\mathbf{v}\mathbf{e}^\top)\big)^\top$ can be computed only once per time-frame, and $\mathbf{W}\mathbf{h}$ can be computed and exponentiated only once per iteration. In a tailored implementation, the update thus amounts to computing a maximum of three matrix-vector multiplications, two element-wise vector multiplications, one element-wise vector division and two element-wise vector powers per iteration, as well as one additional element-wise matrix multiplication per time-frame. The vector $\mathbf{h}$ can be directly initialized with the output solution of the previous frame to speed up convergence. This makes the scheme suitable for real-time applications even if it is computationally more expensive than the scheme proposed in the previous section.

Finally, we emphasize that the beta-divergence has already been used in NMF problems as mentioned above, yet we are not aware of such a formulation for the context of non-negative decomposition on audio streams. Instead the systems based on non-negative decomposition have rather considered the special cases of the Euclidean and Kullback-Leibler cost functions as discussed previously. Moreover, our formulation allows to consider properly limit cases and to develop a single scheme tailored to real-time with convergence guarantees for any value of $\beta \in \mathbb{R}$.

6 Evaluation and Results

In this section, we evaluate the system with the two proposed algorithms on several tasks of multi-source detection in complex auditory scenes. The analysis of complex auditory scenes has received a lot of attention, mainly in the context of *computational auditory scene analysis* [51] which deals with various real-world problems such as source separation, polyphonic music transcription,

recognition of speech in noisy environments, environmental sound recognition in realistic scenarios. As a quantitative evaluation, we first focus on polyphonic music transcription and perform a comparative evaluation using a standard evaluation framework. We then discuss the tasks of drum transcription and environmental sound detection. Since there is no widely accepted evaluation framework with standard evaluation metrics and publicly available databases of accurate ground-truth references for these two tasks, we demonstrate results on different experiments with realistic sound samples. The obtained results confirm the applicability of the proposed system and algorithms to the general problem of multi-source detection in real-time, and the benefits in using flexible controls on the decomposition. In the sequel, we employ the following names to designate the different non-negative decomposition (ND) algorithms tested with the system: (`END`) Euclidean ND in standard formulation, (`SCND`) sparsity-constrained ND directly adapted from [48] with a min-sparsity bound $s_{\min}$, (`SPND`) sparsity-penalized ND developed in Sect. 4 with a sparsity penalty λ_1, (`BND`) beta ND developed in Sect. 5 with an energy-dependent frequency trade-off β.

6.1 Polyphonic Music Transcription

The task of music transcription consists in converting a raw music signal into a symbolic representation such as a score. Considering polyphonic signals, this task is closely related to multiple-pitch estimation, a problem that has been largely investigated for music as well as speech, and for which a wide variety of methods have been proposed (e.g., see [52]).

To evaluate the two proposed algorithms, we considered the problem of polyphonic music transcription since it provides a rigorous framework with widely accepted evaluation metrics and state-of-the-art algorithms as references. We focused on the task of frame-based multiple-pitch estimation according to the standards of the Music Information Retrieval Evaluation eXchange (MIREX) [53].

For the evaluation dataset, we considered the MIDI-Aligned Piano Sounds (MAPS) database [54]. MAPS contains, among other things, isolated samples of piano notes and real recordings of piano pieces with ground-truth references. We selected 25 real pieces recorded with the Yamaha Disklavier Mark III and truncated each of them to 30 s.

In the dictionary, one template was learned for each of the 88 notes of the piano from an audio fragment created by concatenating the three respective samples in MAPS at dynamics piano, mezzo-forte and forte. As a representation front-end, we employed a simple short-time magnitude spectrum, with a frame size of 50 ms leading to 630 samples at a sampling rate of 12600 Hz, and computed with a zero-padded Fourier transform of 1024 bins. The frames were windowed with a Hamming function, and the hopsize was set to 25 ms for template learning and refined to 10 ms for decomposition.

The system was evaluated with the following algorithms and parameters tuned manually to optimize results over the database: `END`, `SPND` with $\lambda_1 = 100$, `BND` with $\beta = 0.5$. The decompositions were respectively about 10 times, 10 times and 5 times faster than real-time under MATLAB on a 2.40 GHz laptop with

4.00 Go of RAM. We also notice that the evaluation for the algorithm `SCND` is not included since it did not improve results compared to `END` or `SPND`, and it was computationally too expensive to run in real-time.

The activation coefficients output by the algorithms were all post-processed with the same transcription threshold set manually to 0.02. We did not use any further post-processing so as to really compare the quality of the observations output by the different algorithms at the frame level. For complementary information, we discuss the use of further post-processing in [23] where minimum-duration pruning is employed for smoothing the observations at the note level.

To compare results, we also performed the evaluation for two off-line systems at the state-of-the-art: one based on beta NMF with an harmonic model and spectral smoothness [17], and another one based on a sinusoidal analysis with a candidate selection exploiting spectral features [55].

We report the evaluation results per algorithm in Table 1. Standard evaluation metrics from the MIREX are used as defined in [53]: precision $\mathcal{P}$, recall $\mathcal{R}$, F-measure $\mathcal{F}$, accuracy $\mathcal{A}$, total error $\mathcal{E}_{\text{tot}}$, substitution error $\mathcal{E}_{\text{subs}}$, missed error $\mathcal{E}_{\text{miss}}$, false alarm error $\mathcal{E}_{\text{fals}}$. All scores are given in percents.

Table 1. Results of the transcription evaluation per algorithm.

Algorithm	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{F}$	$\mathcal{A}$	$\mathcal{E}_{\text{subs}}$	$\mathcal{E}_{\text{miss}}$	$\mathcal{E}_{\text{fals}}$	$\mathcal{E}_{\text{tot}}$
`END`	51.4	63.3	56.7	39.6	16.9	19.8	42.9	79.6
`SPND`	52.8	61.6	56.8	39.7	16.5	21.9	38.5	77.0
`BND`	68.1	65.9	**67.0**	**50.3**	8.5	25.6	22.4	**56.5**
[17]	61.0	66.7	63.7	46.8	10.4	22.9	32.3	65.6
[55]	60.0	70.8	**65.0**	**48.1**	16.3	12.8	30.8	**60.0**

Overall, the results show that the proposed real-time system and algorithms perform comparably to the state-of-the-art off-line algorithms of [17] and [55]. The algorithm `BND` even outperforms the other approaches for all metrics. Sparsity control in `SPND` improves the economy in the usage of note templates for reconstructing the music signal, resulting in general to a smaller recall but a greater precision compared to `END`. In other terms, more notes are missed but this is compensated by the reduction of note insertions and substitutions. As a result, there is no noticeable global improvement with sparsity control on the general transcription in terms of F-measure, accuracy and total error. This is in contrast with the benefits brought by the flexible control on the energy-dependent frequency compromise in the decomposition for the algorithm `BND`.

To assess the generalization capacity of the system, we focused on the algorithm `BND` and performed two other evaluations. In the first, the templates were learned as above but with three pianos: the Yamaha Disklavier Mark III from MAPS, the Steinway D from MAPS, and the Pianoforte from the Real World Computing Music Database [56]. This resulted in the following general metrics: $\mathcal{F} = 63.4\,\%$, $\mathcal{A} = 46.5\,\%$, $\mathcal{E}_{\text{tot}} = 60.7\,\%$. In the second, the test piano was left out

from training and the templates were learned with the two other pianos. This resulted in the following general metrics: $\mathcal{F} = 58.4\,\%$, $\mathcal{A} = 41.2\,\%$, $\mathcal{E}_{\text{tot}} = 69.1\,\%$.

This shows that the best results are obtained when only the test piano is used for training, meaning that considering other pianos does not add useful information to the system. When the test piano is not used for training, generalization is not perfect yet the system with the algorithm `BND` is still competitive with the other off-line systems. We also emphasize that in a real-time setup, the templates can in general be learned from the corresponding piano.

To go further, we also submitted the system to MIREX 2010 where it was evaluated and compared to other algorithms on different tasks of polyphonic music transcription for various instruments and kinds of music.[4] The system we submitted was a preliminary version of the algorithm `BND` with just piano templates in the dictionary as described in [23], and was the only real-time system in competition. It performed however comparably to the other systems, with the following general metrics at the frame level for general music with various instruments: $\mathcal{F} = 57.4\,\%$, $\mathcal{A} = 45.7\,\%$, $\mathcal{E}_{\text{tot}} = 84.7\,\%$. Moreover, the system also finished second on seven systems for the note level tasks of tracking in general music with various instruments and of tracking in piano music.

6.2 Drum Transcription

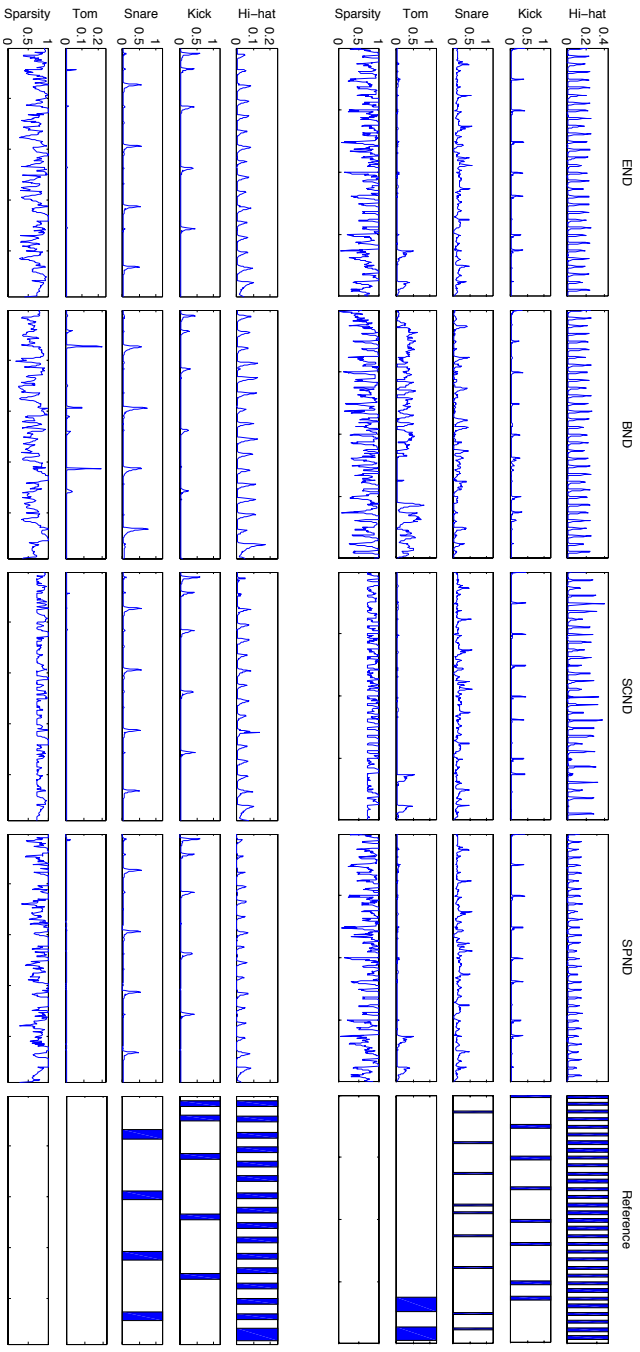
For the problem of drum transcription, we considered two drum loops as sample examples. The first one contains three instruments: kick, snare and hi-hat, and the second one contains four instruments of a different drum kit: kick, snare, hi-hat and tom.

The drum loops were both decomposed onto the same dictionary of four templates representing a kick, a snare, a hi-hat and a tom. The templates were learned from isolated samples of the second drum kit. This was done to assess the generalization capacity of the system and algorithms on the first loop. Moreover, we added an important background of recorded polyphonic music from a wind quintet to the second loop in order to assess robustness issues as well. The two corresponding drum loops are available on the companion website. The representation front-end used for decomposition of the loops was the same as for polyphonic music transcription, except that the sampling rate was set to 22050 Hz to account for high-frequency discriminative information in the hi-hat.

Concerning the non-negative decomposition, we employed the following algorithms: `END`, `SCND` with $s_{\min} = 0.7$, `SPND` with $\lambda_1 = 100$, `BND` with $\beta = 0.5$. The decompositions were respectively about 30 times, 3 times, 30 times and 20 times faster than real-time. The results of the decompositions are shown in Fig. 2.

Figure 2a shows the activations of each template over time and the sparsity of solutions, as defined in (8), for the different algorithms on the first drum loop. A hand-labeled reference also represents the binary occurrence of the respective

[4] The results of the 2010 MIREX evaluation for multiple fundamental frequency estimation and tracking are available on-line: `http://www.music-ir.org/mirex/wiki/2010:Multiple_Fundamental_Frequency_Estimation_%26_Tracking_Results`.



(a) First drum loop. (b) Second drum loop.

Fig. 2. Detection of drum instrument occurrences.

sources over time for comparison. It can be seen that all algorithms have correctly detected the three drum instruments, proving that the system is capable of generalization. However, this example reveals the misuse of the tom template in the decomposition for all algorithms. Indeed the tom is activated even if there was no tom in the original sequence. The algorithm `BND` for $\beta = 0.5$ is the worst with regards to this issue. Decreasing β to 0, corresponding to the Itakura-Saito divergence, the situation even gets worse. Indeed, the more β decreases, the more low-energy components are emphasized in the decomposition. This gets critical when β is null since all components, including noisy parts, are equally weighted (and it is worse for $\beta < 0$). Increasing β up to values between 1 and 2, corresponding respectively to the Kullback-Leibler and Euclidean cost functions, the results improve progressively to reach that of the algorithm `END`. It can also be seen that adding a sparsity penalty with the algorithm `SPND` helps reduce the tom activation compared to `END` and `BND`. Using a min-sparsity bound, the algorithm `SCND` is computationally more expensive than `SPND` but does not improve the results compared to `SPND`.

Figure 2b shows the activations and the sparsity of solutions for the different algorithms on the second drum loop. It reveals that the system has correctly detected the four drum instruments despite the important background music. However, this example illustrates the misuse of several templates in the reconstruction of the incoming signal. In particular, it appears that the algorithm `BND` for $\beta = 0.5$ suffers robustness limitations since the tom is highly activated in the whole sequence, whereas it is actually only played twice at the end in the original sequence. Moreover, the kick also exhibits wrong activations compared to the other algorithms. Again, the situation would get worse if we decrease β to 0, but would improve as β increases between 1 and 2. The three other algorithms are much more robust and do not wrongly activate the tom or the kick. Instead, the general level of the snare activation is slightly increased. Adding a penalty term on sparsity with the algorithm `SPND` does not help to reduce this phenomenon compared to `END`. Augmenting λ_1 would reduce the level of the snare activation on the one hand, but also that of the hi-hat on the other hand so that some strokes would be missed. The more computationally expensive algorithm `SCND` does not allow to alleviate this issue neither. Moreover, the tom strokes hinder the detection of the hi-hat with `SCND` because of the rigid sparsity bound constraint compared to the flexible penalty of `SPND`.

6.3 Environmental Sound Detection

For the task of environmental sound detection, we created three complex auditory scenes containing several sound sources among 13 selected common sound events: car horn, beep of a microwave, noise of a refrigerator, electric razor, spray, bell ringing, dog barking, ice cubes falling in an empty glass, closing a door cupboard, clinking glasses, scraping a metal pan, sharpening a knife, removing a cork from a bottle. These sound events are quite various in frequency content and shape, and most of them are non-stationary in different aspects. For example, the razor, spray and refrigerator are long steady sounds with important

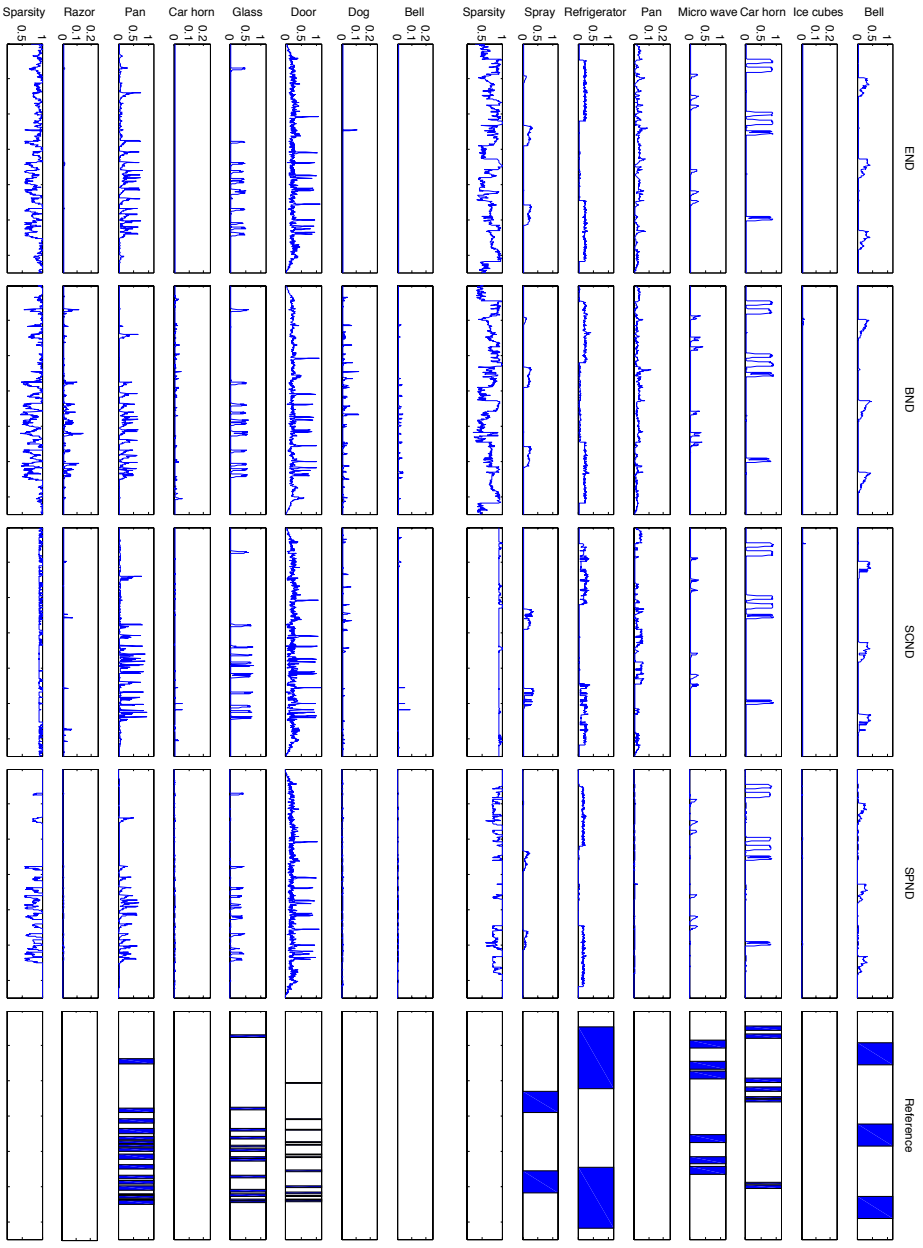
noisy components as well as spectral modulation at a micro-temporal level, and in particular roughness for the razor. The dog, ice cubes, door, glasses, pan, knife and cork are shorter but all exhibit a clear non-stationary temporal pattern in their spectrum. The car horn, microwave and bell are much more stationary and have a spectrum similar to simplified instrument notes with an evident tone and an almost stationary profile with attack, sustain and release.

To make the created scenes realistic and assess the robustness of the system, we also added an important amount of background noise. The first scene was created by mixing three sound sources within the background of a railway station hall featuring many people speaking and footsteps. The second contains five sound sources within the background of a bus stop featuring noise from road construction, from the traffic and from a bus. The third contains five sound sources within the background of a shop featuring many people speaking and noise from human activities. The three corresponding environmental scenes are available on the companion website. The respective dictionaries used for decomposition of the scenes were each composed of seven templates with the present events and other events from the selection. The representation front-end was exactly the same as for polyphonic music transcription.

The mixed auditory scenes were decomposed with the following algorithms: `END`, `SCND` with $s_{\min} = 0.9$, `SPND` with $\lambda_1 = 1000$, `BND` with $\beta = 0.5$. The decompositions were respectively about 40 times, 2 times, 40 times and 20 times faster than real-time. The results of the decompositions are shown in Fig. 3.

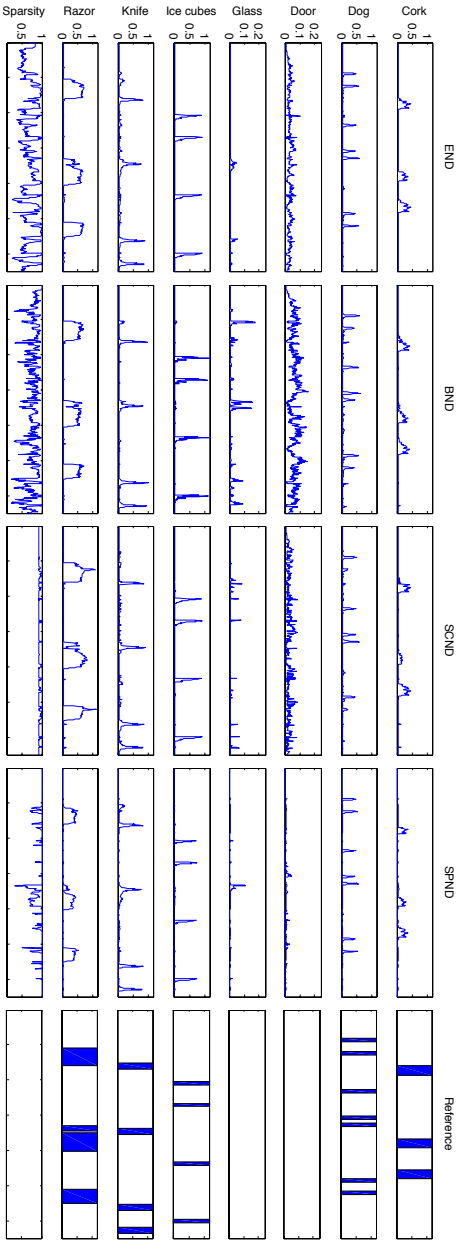
Figure 3a shows the activations of each template and the sparsity of solutions for the different algorithms on the first auditory scene. It can be seen in general that all algorithms have correctly detected the three sound events present in the auditory scene, but that the system tends to use too many templates. The salient voices and footsteps in the background noise activate the bell, dog, car horn and razor templates, and the whole background noise tends to higher the activation levels of the door and the pan templates. These errors are clearly demonstrated with the algorithm `END`. The algorithm `BND` for $\beta = 0.5$ performs even poorer, and using other values of β does not allow to circumvent the problem of wrong activations. The algorithm `SCND` seems to perform better, even if a few errors are still present. Augmenting the min-sparsity bound $s_{\min}$ would slightly attenuate these errors, but would also augment the number of missed events. The algorithm `SPND`, despite being computationally cheaper, seems to fit sparsity better to the signal dynamic and is more robust. It allows a sparser decomposition compared to `END` and `BND`, while being flexible enough when several sources are present compared to `SCND`. It also allows to remove the wrong activations observed for the other algorithms, and to reduce the general activation level of the pan, yet that of the door is still relatively high.

Figure 3b shows the activations and the sparsity of solutions on the second auditory scene. Again, the algorithms have in general correctly detected the five sound events but use too many templates to reconstruct the signal. In particular, the background noise activates the pan even if there is no pan in the original sequence. These errors are clearly demonstrated with the algorithms `END`, `BND` and



(a) First auditory scene. (b) Second auditory scene.

Fig. 3. Detection of environmental sound events.



(c) Third auditory scene.

Fig. 3. Detection of environmental sound events (continued).

SCND. Changing the value of β does not help to alleviate this issue, and increasing $s_{\min}$ would undermine the correct detection of several sound events. Moreover, for the three algorithms, there is a clear wrong detection in the spray event template, where a gas noise from the bus is confused with the spray source. These issues are not reported at all for the algorithm SPND which is very robust against the background noise on this example and still detects correctly the occurrences of the five sources. This is a consequence of the sparsity being adapted to the signal dynamics thanks to the penalization in SPND, whereas the less flexible bound constraint of SCND reveals insufficient in this example.

Figure 3c shows the activations and the sparsity of solutions on the third auditory scene. The results corroborate the previous ones. Even if all algorithms have been able to detect the five sound events, the issues of robustness discussed previously are confirmed. Here, the door and glass templates are activated while there are not present in the original sequence. The algorithm BND also wrongly detects several occurrences of the dog source. Furthermore, we notice again a limitation of the sparsity bound constraint $s_{\min}$ in SCND since some dog occurrences are missed. The algorithm SPND seems to cope better with these issues by reducing importantly the door and glass activations compared to END, while still detecting correctly the present sources. This again results from the sparsity value being flexible enough even if it is regularized.

7 Conclusion

In this paper, we discussed the problem of real-time detection of overlapping sound events. To address this problem, we designed a general system based on NMF techniques, and we investigated the introduction of flexible controls in the non-negative decomposition of the input signal. We proposed two computationally efficient and provably convergent algorithms that include controls respectively on the sparsity and on the frequency compromise of the decomposition. We applied the proposed algorithms to several multi-source detection tasks with real-time constraints and discussed the benefits in using such controls to improve detection.

On the one hand, sparsity control has revealed efficient for improving the robustness of the system in the task of environmental sound detection, where one has to deal with background noise and salient undesirable sound events with highly overlapping frequency content. For the task of drum transcription, however, sparsity did not improve significantly the results on the considered examples, even if the system was still able to correctly detect the different instruments in general. Further investigation is needed on this line to understand and address the problem. On the other hand, a control on the frequency compromise of the decomposition has revealed efficient in the task of polyphonic music transcription, where partials or high frequencies with low energy are important for discriminating between the different musical events. This control thus helped the system to perform comparably to the state-of-the-art but in real-time. This is encouraging for further improvement of the proposed approaches.

To begin with, we want to develop a computationally efficient scheme that couples the advantages of the two proposed algorithms. Recent advances in [21] demonstrate the possibility to combine the beta-divergence with sparsity regularization while keeping cost monotonicity. Such a scheme may find benefits in complex situations of environmental sound detection or music information retrieval. For example, in the task of multiple-instrument transcription, a sparsity control in combination with frequency trade-off may help the discrimination and the detection of the correct instruments when several instruments overlap in pitch range. Another example is that of melody extraction, where frequency trade-off may improve discrimination and separation while sparsity may help to find the most predominant musical events that define the melody. Other parametric families of divergences than the beta-divergences could also be studied in these contexts to find relevant interpretation of their parameters as flexible controls on the decomposition.

Also, we would like to overcome the implicit assumption in NMF techniques that the templates are stationary. This has not been a serious issue here even if we considered non-stationary sounds in our experiments. The rigorous consideration of non-stationarity is however likely to become crucial when considering sounds with more complex temporal profiles than those employed in this paper. To tackle this limitation, it is possible to consider front-end representations that capture variability over a short time-span, such as the modulation spectrum used in [9]. We believe however that a more elaborate approach is necessary to address efficiently the non-stationarity of real-world objects, by considering the temporality of templates directly within the NMF model. We could for example consider extended models as those proposed in [18, 19, 57] which allow to deal with time-varying objects. Another potential approach is to combine NMF with a state representation of sounds similar to hidden Markov models as in [58–60]. These two approaches should be investigated further.

Besides modeling the temporality of the events, the template learning phase may also be improved. In our case of rank-one non-negative factorization, we could have used the singular value decomposition theory instead. An advantage in formulating the learning phase in an NMF framework is that of the variety of extended schemes available to learn one or more templates for each sound source. For example, we tried employing the beta-divergence for template learning, yet it did not improve systematically the results in our experience. Further considerations are also needed in this direction.

In addition, other representation front-ends could be employed instead of a simple magnitude spectrum. For the task of polyphonic music transcription, considering non-linear frequency scales (e.g., constant-Q transform) may improve the system. In a more general setup, we would like also to address the use of a wavelet transform, maybe coupled with a modulation spectrum representation, to provide a multi-scale analysis of the spectro-temporal features of the sounds. The extension of NMF to tensors may also enhance the system, allowing for instance to use multi-channel information in the representation. We have also extended the proposed sparse algorithm to deal with complex representations.

This extension has not been discussed in the paper, but can help to consider more informative representations that account for phase information.

We would like finally to improve further the robustness and the generalization capacity of the system. Concerning robustness, a first direction may be to model information from the encoding coefficients during template learning to improve detection during decomposition. We could alternatively investigate the use of non-fixed updated basis vectors to absorb noise and other undesirable sound components. Concerning generalization, we may enhance our model to deal with adaptive event templates. For example, second-order cone programming may be employed to consider non-fixed templates constrained within geometric cones. A similar idea has already been proposed in [48] for supervised classification with NMF. Other possibilities come from the use of a hierarchical instrument basis as in [39] or more generally from convex NMF techniques with convergence guarantees as proposed in [21]. Future work should address the adaptation of these approaches to the proposed algorithms.

Acknowledgments. This work was partially funded by a doctoral fellowship from the UPMC (EDITE). The authors would like to thank Chunghsin Yeh and Roland Badeau for their valuable help, Emmanouil Benetos for his helpful comments on the paper, Valentin Emiya for kindly providing the MAPS database, as well as Patrick Hoyer and Emmanuel Vincent for sharing their source code.

References

1. Paatero, P., Tapper, U.: Positive Matrix Factorization: A Non-Negative Factor Model with Optimal Utilization of Error Estimates of Data Values. Environmetrics. 5(2), 111–126 (1994)
2. Lee, D.D., Seung, H.S.: Learning the Parts of Objects by Non-Negative Matrix Factorization. Nature. 401(6755), 788–791 (1999)
3. Lee, D.D., Seung, H.S.: Algorithms for Non-Negative Matrix Factorization. In: Advances in Neural Information Processing Systems, vol. 13, pp. 556–562. MIT Press, Cambridge, MA, USA (2001)
4. Sha, F., Saul, L.K.: Real-Time Pitch Determination of One or More Voices by Nonnegative Matrix Factorization. In: Advances in Neural Information Processing Systems, vol. 17, pp. 1233–1240. MIT Press, Cambridge, MA, USA (2005)
5. Cheng, C.-C., Hu, D.J., Saul, L.K.: Nonnegative Matrix Factorization for Real Time Musical Analysis and Sight-Reading Evaluation. In: 33rd IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 2017–2020. Las Vegas, NV, USA (2008)
6. Paulus, J., Virtanen, T.: Drum Transcription with Non-Negative Spectrogram Factorisation. In: 13th European Signal Processing Conference. Antalya, Turkey (2005)
7. Niedermayer, B.: Non-Negative Matrix Division for the Automatic Transcription of Polyphonic Music. In: 9th International Conference on Music Information Retrieval, pp. 544–549. Philadelphia, PA, USA (2008)
8. Cont, A.: Realtime Multiple Pitch Observation Using Sparse Non-Negative Constraints. In: 7th International Conference on Music Information Retrieval. Victoria, Canada (2006)

9. Cont, A., Dubnov, S., Wessel, D.: Realtime Multiple-Pitch and Multiple-Instrument Recognition for Music Signals Using Sparse Non-Negative Constraints. In: 10th International Conference on Digital Audio Effects. Bordeaux, France (2007)
10. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press (2004)
11. Zdunek, R., Cichocki, A.: Nonnegative Matrix Factorization with Quadratic Programming. Neurocomputing. 71(10–12), 2309–2320 (2008)
12. Sha, F., Lin, Y., Saul, L.K., Lee, D.D.: Multiplicative Updates for Nonnegative Quadratic Programming. Neural Computation. 19(8), 2004–2031 (2007)
13. Basu, A., Harris, I.R., Hjort, N.L., Jones, M.C.: Robust and Efficient Estimation by Minimising a Density Power Divergence. Biometrika. 85(3), 549–559 (1998)
14. Eguchi, S., Kano, Y.: Robustifying Maximum Likelihood Estimation. Technical Report, Institute of Statistical Mathematics, Tokyo, Japan (2001)
15. O'Grady, P.D., Pearlmutter, B.A.: Discovering Speech Phones Using Convolutive Non-Negative Matrix Factorisation with a Sparseness Constraint. Neurocomputing. 72(1–3), 88–101 (2008)
16. FitzGerald, D., Cranitch, M., Coyle, E.: On the Use of the Beta Divergence for Musical Source Separation. In: 20th IET Irish Signals and Systems Conference. Galway, Ireland (2009)
17. Vincent, E., Bertin, N., Badeau, R.: Adaptive Harmonic Spectral Decomposition for Multiple Pitch Estimation. IEEE Transactions on Audio, Speech and Language Processing. 18(3) (2010)
18. Hennequin, R., Badeau, R., David, B.: Time-Dependent Parametric and Harmonic Templates in Non-Negative Matrix Factorization. In: 13th International Conference On Digital Audio Effects, 246–253. Graz, Austria (2010)
19. Hennequin, R., Badeau, R., David, B.: NMF With Time-Frequency Activations to Model Nonstationary Audio Events. IEEE Transactions on Audio, Speech, and Language Processing. 19(4), 744–753 (2011)
20. Nakano, M., Kameoka, H., Le Roux, J., Kitano, Y., Ono, N., Sagayama, S.: Convergence-Guaranteed Multiplicative Algorithms for Nonnegative Matrix Factorization with β-Divergence. In: IEEE International Workshop on Machine Learning for Signal Processing, pp. 283–288. Kittilä, Finland (2010)
21. Févotte, C., Idier, J.: Algorithms for Nonnegative Matrix Factorization with the β-divergence. Neural Computation. 23(9), 2421–2456 (2011)
22. Badeau, R., Bertin, N., Vincent, E.: Stability Analysis of Multiplicative Update Algorithms and Application to Nonnegative Matrix Factorization. IEEE Transactions on Neural Networks. 21(12), 1869–1881 (2010)
23. Dessein, A., Cont, A., Lemaitre, G.: Real-Time Polyphonic Music Transcription with Non-Negative Matrix Factorization and Beta-Divergence. In: 11th International Society for Music Information Retrieval Conference, pp. 489–494. Utrecht, Netherlands (2010)
24. Berry, M.W., Browne, M., Langville, A., Pauca, V.P., Plemmons, R.J.: Algorithms and Applications for Approximate Nonnegative Matrix Factorization. Computational Statistics & Data Analysis. 52(1), 155–173 (2007)
25. Cichocki, A., Zdunek, R., Phan, A.H., Amari, S.-i.: Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-Way Data Analysis and Blind Source Separation. Wiley-Blackwell (2009)
26. Abdallah, S.A., Plumbley, M.D.: Polyphonic Music Transcription by Non-Negative Sparse Coding of Power Spectra. In: 5th International Conference on Music Information Retrieval, pp. 318–325. Barcelona, Spain (2004)

27. Smaragdis, P., Brown, J.C.: Non-Negative Matrix Factorization for Polyphonic Music Transcription. In: IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 177–180. New Paltz, NY, USA (2003)
28. Virtanen, T., Klapuri, A.: Analysis of Polyphonic Audio Using Source-Filter Model and Non-Negative Matrix Factorization. In: Neural Information Processing Systems Workshop on Advances in Models for Acoustic Processing (2006)
29. Raczyński, S.A., Ono, N., Sagayama, S.: Multipitch analysis with Harmonic Nonnegative Matrix Approximation. In: 8th International Conference on Music Information Retrieval, pp. 381–386. Vienna, Austria (2007)
30. Marolt, M.: Non-Negative Matrix Factorization with Selective Sparsity Constraints for Transcription of Bell Chiming Recordings. In: 6th Sound and Music Computing Conference, pp. 137–142. Porto, Portugal (2009)
31. Grindlay, G., Ellis, D.P.W.: Multi-Voice Polyphonic Music Transcription Using Eigeninstruments. In: IEEE Workshop on Applications of Signal Processing to Audio and Acoustics. New Paltz, NY, USA (2009)
32. Févotte, C., Bertin, N., Durrieu, J.-L.: Nonnegative Matrix Factorization with the Itakura-Saito Divergence with Application to Music Analysis. Neural Computation. 21(3), 793–830 (2009)
33. Févotte, C.: Itakura-Saito Nonnegative Factorizations of the Power Spectrogram for Music Signal Decomposition. In: Wang, W. (ed.) Machine Audition: Principles, Algorithms and Systems, pp. 266–296. IGI Global Press (2010)
34. Bertin, N., Févotte, C., Badeau, R.: A Tempering Approach for Itakura-Saito Non-Negative Matrix Factorization. With Application to Music Transcription. In: IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 1545–1548. Taipei, Taiwan (2009)
35. Bertin, N., Badeau, R., Vincent, E.: Enforcing Harmonicity and Smoothness in Bayesian Non-Negative Matrix Factorization Applied to Polyphonic Music Transcription. IEEE Transactions on Audio, Speech and Language Processing. 18(3), 538–549 (2010)
36. Shashanka, M., Raj, B., Smaragdis, P.: Probabilistic Latent Variable Models as Nonnegative Factorizations. Computational Intelligence and Neuroscience. 2008 (2008)
37. Smaragdis, P., Raj, B., Shashanka, M.: Sparse and Shift-Invariant Feature Extraction from Non-Negative Data. In: IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 2069–2072. Las Vegas, NV, USA (2008)
38. Mysore, G.J., Smaragdis, P.: Relative Pitch Estimation of Multiple Instruments. In: IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 313–316. Washington, DC, USA (2009)
39. Grindlay, G., Ellis, D.P.W.: Transcribing Multi-Instrument Polyphonic Music with Hierarchical Eigeninstruments. IEEE Journal of Selected Topics in Signal Processing. 5(6), pp.1159–1169 (2011)
40. Hennequin, R., Badeau, R., David, B.: Scale-Invariant Probabilistic Latent Component Analysis. In: IEEE Workshop on Applications of Signal Processing to Audio and Acoustics. New Paltz, NY, USA (2011)
41. Fuentes, B., Badeau, R., Richard, G.: Adaptive Harmonic Time-Frequency Decomposition of Audio Using Shift-Invariant PLCA. In: 36th International Conference on Acoustics, Speech, and Signal Processing, pp. 401–404. Prague, Czech Republic (2011)
42. Benetos, E., Dixon, S.: Multiple-Instrument Polyphonic Music Transcription Using a Convolutive Probabilistic Model. In: 8th Sound and Music Computing Conference, pp. 19–24. Padova, Italy (2011)

43. Karvanen, J., Cichocki, A.: Measuring Sparseness of Noisy Signals. In: 4th International Symposium on Independent Component Analysis and Blind Signal Separation, pp. 125–130. Nara, Japan (2003)
44. Hoyer, P.O.: Non-Negative Matrix Factorization with Sparseness Constraints. Journal of Machine Learning Research. 5, 1457–1469 (2004)
45. Eggert, J., Körner, E.: Sparse Coding and NMF. In: IEEE International Joint Conference on Neural Networks, pp. 2529–2533. Budapest, Hungary (2004)
46. Albright, R., Cox, J., Duling, D., Langville, A.N., Meyer, C.D.: Algorithms, Initializations, and Convergence for the Non Negative Matrix Factorization. Technical Report, NC State University (2006)
47. Hoyer, P.O.: Non-Negative Sparse Coding. In: 12th IEEE Workshop on Neural Networks for Signal Processing, pp. 557–565. Martigny, Switzerland (2002)
48. Heiler, M., Schnörr, C.: Learning Sparse Representations by Non-Negative Matrix Factorization and Sequential Cone Programming. Journal of Machine Learning Research. 7, 1385–1407 (2006)
49. Virtanen, T.: Monaural Sound Source Separation by Nonnegative Matrix Factorization with Temporal Continuity and Sparseness Criteria. IEEE Transactions on Audio, Speech and Language Processing. 15(3), 1066–1074 (2007)
50. Kompass, R.: A Generalized Divergence Measure for Nonnegative Matrix Factorization. Neural Computation. 19(3), 780–791 (2007)
51. Wang, D., Brown, G.J.: Computational Auditory Scene Analysis: Principles, Algorithms and Applications. Wiley-IEEE Press (2006)
52. Klapuri, A., Davy, M.: Signal Processing Methods for Music Transcription. Springer, New York, NY, USA (2006)
53. Bay, M., Ehmann, A.F., Downie, J.S.: Evaluation of Multiple-F0 Estimation and Tracking Systems. In: 10th International Society for Music Information Retrieval Conference, pp. 315–320. Kobe, Japan (2009)
54. Emiya, V., Badeau, R., David, B.: Multipitch Estimation of Piano Sounds Using a New Probabilistic Spectral Smoothness Principle. IEEE Transactions on Audio, Speech, and Language Processing. 18(6), 1643–1654 (2010)
55. Yeh, C., Roebel, A., Rodet, X.: Multiple Fundamental Frequency Estimation and Polyphony Inference of Polyphonic Music Signals. IEEE Transactions on Audio, Speech, and Language Processing. 18(6), 1116–1126 (2010)
56. Goto, M., Hashiguchi, H., Nishimura, T., Oka, R.: RWC Music Database: Popular, Classical, and Jazz Music Databases. In: 3rd International Conference on Music Information Retrieval, pp. 287–288. Paris, France (2002)
57. Badeau, R.: Gaussian Modeling of Mixtures of Non-Stationary Signals in the Time-Frequency Domain (HR-NMF). In: IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 253–256. New Paltz, NY, USA (2011)
58. Mysore, G., Smaragdis, P., Raj, B.: Non-Negative Hidden Markov Modeling of Audio with Applications to Source Separation. In: 9th International Conference on Latent Variable Analysis and Signal Separation, pp. 140–148 (2010)
59. Nakano, M., Le Roux, J., Kameoka, H., Kitano, Y., Ono, N., Sagayama, S.: Nonnegative Matrix Factorization with Markov-Chained Bases for Modeling Time-Varying Patterns in Music Spectrograms. In: 9th International Conference on Latent Variable Analysis and Signal Separation, pp. 149–156 (2010)
60. Benetos, E., Dixon, S.: A Temporally-Constrained Convolutive Probabilistic Model for Pitch Detection. In: IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 133–136. New Paltz, NY, USA (2011)