Sparse Non-negative Matrix Factor 2-D Deconvolution for Automatic Transcription of Polyphonic Music

Mikkel N. Schmidt and Morten Mørup
Technical University of Denmark
Informatics and Mathematical Modelling
Richard Petersens Plads, Building 321
DK-2800 Kgs. Lyngby, Denmark
{mns,mm}@imm.dtu.dk

December 1, 2005

Abstract

We present a novel method for automatic transcription of polyphonic music based on a recently published algorithm for non-negative matrix factor 2-D deconvolution. The method works by simultaneously estimating a time-frequency model for an instrument and a pattern corresponding to the notes which are played based on a log-frequency spectrogram of the music.

1 Introduction

Automatic transcription of polyphonic music is a very difficult and currently unsolved problem. The task is to create a system which can extract the musical score for a piece of recorded music where multiple notes are played simultaneously by different instruments. Even for trained musicians, manual music transcription is very difficult to perform. Often it is necessary to listen to the music repeatedly and transcribe one instrument at a time [6]. In the recent years a number of different approaches for automatic music transcription have been proposed. However, currently no system exists which performs as well as skilled musicians [4].

Plumbley et al. [6] distinguishes between knowledge based and data driven methods. The former denotes methods based on knowledge of the physics of music generation and the human auditory system and seeks to mimic transcription as it is performed by humans, whereas the latter denotes methods aiming at extracting information of the structure of the music directly from the audio signal.

The perception of music can be seen as the process of transforming the low-level representation of the music, the audio waveform, into a high level representation, e.g. "Mozart's Sonata in C-major". The term *mid level representation* is often used to denote the intermediate representations of signals. A good mid-level representation for audio should be able to separate individual sources, be invertible in a perceptual sense, reduce the number of components and reveal the most important attributes of the sound [1].

Current methods for automatic music transcription are often based on modeling the music spectrum as a sum of harmonic sources and estimating the fundamental frequencies of these sources. This information constitutes an ad hoc mid-level representation.

In order to succesfully create a system for automatic music transcription, the information contained in the analyzed audio signal must be combined with knowledge of the structure of music [4]. The problem of music transcription has many similarities with that of speech recognition, where successful systems com-

bine carefully selected speech features with statistical language models.

Recently, Smaragdis and Brown [8] introduced a data driven method for automatic music transcription based on non-negative matrix factorization. Their idea is to factorize a magnitude spectrogram of music into factors corresponding to models of individual notes and the times at which they are played. This method provides a very useful mid-level representation but has the disadvantage that it does not in fact model notes but rather unique events. Thus, if two notes are always played simultaneously they will be modelled as one component. Also, some components might not correspond to notes but rather model e.g. background noise.

In this paper we propose a novel method for automatic transcription of polyphonic music. The method is based on a recently introduced non-negative matrix factor 2-D deconvolution model [7, 5], which is used to compute a very useful mid-level representation of music. The idea is to simultaneously model the instruments and the notes which are played. The described method is purely data driven and can be combined with a knowledge based system to give a functioning music transcription system.

2 Method

It is well known that the scale of music is logarithmic. The twelve-tone equal tempered scale which forms the basis of modern western music divides each octave into twelve halfnotes where the frequency ratio between each successive halfnote is equal. If F_1 is the fundamental frequency of one note, then the fundamental frequency of the note which is p halfnotes above can be expressed as $F_2 = F_1 \cdot 2^{p/12}$. Taking the logarithm gives: $\log F_2 = \log F_1 + \frac{p}{12} \log 2$, thus in a log-frequency representation the notes are linearly spaced.

We assume, that an instrument can be modelled by a specific time-frequency signature modulating the sound of the instrument over the time τ. When an instrument plays a note at a certain time, this signature is displaced onto the time axis. Similarly, when an instrument plays a note with a certain pitch, ϕ, it corresponds to displacing the time-frequency signature on the log-frequency axis. This constitutes the basic idea in the recently introduced non-negative matrix factor 2-D deconvolution (NMF2D) model [7]:

$$\mathbf{V} \approx \mathbf{\Lambda} = \sum_{\tau} \sum_{\phi} \overset{\downarrow\phi}{\mathbf{W}^{\tau}} \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}, \tag{1}$$

where $\mathbf{V} \in \mathcal{R}_{+}^{M \times N}$, $\mathbf{W}^{\tau} \in \mathcal{R}_{+}^{M \times d}$ and $\mathbf{H}^{\phi} \in \mathcal{R}_{+}^{d \times N}$ are non negative matrices, $\downarrow \phi$ denotes the downward shift operator which moves each element in the matrix ϕ rows down, and $\rightarrow \tau$ denotes the right shift operator which moves each element in the matrix τ columns to the right.

The matrix $\mathbf{V}$ is a log-frequency magnitude spectrogram representation of a piece of polyphonic music. The columns of $\mathbf{W}^{\tau}$ correspond to the time-frequency signature of a given instrument, and the rows of $\mathbf{H}^{\phi}$ correspond to the time-pitch signature of each instrument, i.e. which notes are played by the instrument at what time.

In order to estimate the parameters of the model, we use the following least squares cost function:

$$C_{LS} = \frac{1}{2} ||\mathbf{V} - \mathbf{\Lambda}||_f^2 = \frac{1}{2} \sum_i \sum_j (\mathbf{V}_{i,j} - \mathbf{\Lambda}_{i,j})^2. \tag{2}$$

Based on gradient descent with multiplicative updates, the following recursive updates can be used to compute $\mathbf{W}^{\tau}$ and $\mathbf{H}^{\phi}$ [7, 5]:

$$\mathbf{W}^{\tau} \leftarrow \mathbf{W}^{\tau} \bullet \frac{\sum_{\phi} \overset{\uparrow\phi}{\mathbf{V}} \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}^{T}}{\sum_{\phi} \overset{\uparrow\phi}{\mathbf{\Lambda}} \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}^{T}}, \tag{3}$$

$$\mathbf{H}^{\phi} \leftarrow \mathbf{H}^{\phi} \bullet \frac{\sum_{\tau} \overset{\downarrow\phi}{\mathbf{W}^{\tau}}^{T} \overset{\leftarrow\tau}{\mathbf{V}}}{\sum_{\tau} \overset{\downarrow\phi}{\mathbf{W}^{\tau}}^{T} \overset{\leftarrow\tau}{\mathbf{\Lambda}}}, \tag{4}$$

where $A \bullet B$ denotes element-wise multiplication and $\frac{A}{B}$ denotes element-wise division. The algorithm has been proven to converge with these updates [5].

It is important to note that the NMF2D model has certain ambiguities between the factors $\mathbf{W}^{\tau}$ and

$\mathbf{H}^{\phi}$. If a time-frequency signature of an instrument is shifted in time or frequency, there exists an adverse shift of the time-pitch signature (if we disregard edge effects). In order to alleviate the shift ambiguity it can be useful to shift $\mathbf{W}^{\tau}$ and $\mathbf{H}^{\phi}$ during the recursive computation, e.g. such that the geometric mean value of the row coefficients in $\mathbf{W}^{\tau}$ and the column coefficients of $\mathbf{H}^{\phi}$ are centered.

Another ambiguity is, that NMF2D is not in general unique. If the data does not span the positive octant adequately a rotation of $\mathbf{W}^{\tau}$ and adverse rotation of $\mathbf{H}^{\phi}$ can yield equivalent results. Furthermore, the structure of a factor in $\mathbf{H}^{\phi}$ can to some extend be put into the signature of the same factor in $\mathbf{W}^{\tau}$ and vice versa [5]. The upper harmonics in the time-frequency signature of an instrument in $\mathbf{W}^{\tau}$ can for example to some extent be modelled by notes of higher pitch being present in $\mathbf{H}^{\phi}$. In order to ensure unique solutions, constraints in the form of sparseness are useful [2, 3]. In this case we wish to have all information pertaining to the instrument included i $\mathbf{W}^{\tau}$, which can be ensured by a sparsity constraint on $\mathbf{H}^{\phi}$. A sparse solution can be obtained by modifying the cost function so that we minimize the norm of $\mathbf{H}^{\phi}$ while keeping the norm of each instrument in $\mathbf{W}^{\tau}$ constant. While Hoyer [2, 3] uses the 1-norm we here use the $1/2$-norm since it more heavily penalizes small values in $\mathbf{H}^{\phi}$ [5]. To find the update equations for the sparse NMF2D (SNMF2D) we consider the following cost function [5]:

$$C_{SLS} = C_{LS} + \beta \cdot \|\mathbf{H}\|_{1/2} \quad (5)$$

$$s.t. \quad \|\mathbf{W}_d\|_2 = 1, \quad (6)$$

where:

$$\|\mathbf{H}\|_{1/2} = \left(\sum_{\phi,d,j} \left(|\mathbf{H}^{\phi}_{d,j}| \right)^{1/2} \right)^2, \quad (7)$$

$$\|\mathbf{W}_d\|_2 = \left(\sum_{\tau,i} \left(|\mathbf{W}^{\tau}_{i,d}| \right)^2 \right)^{1/2}, \quad (8)$$

and β is a sparseness parameter defining the relative weight of the sparseness term to the approximation of $\mathbf{V}$.

Based on gradient descent, the following recursive updates can be used to compute $\mathbf{W}^{\tau}$ and $\mathbf{H}^{\phi}$ [5]:

$$\mathbf{W}^{\tau} \leftarrow \mathbf{W}^{\tau} + \eta_W \left(\sum_{\phi} (\overset{\uparrow\phi}{\mathbf{V}} - \mathbf{\Lambda}) \overset{\rightarrow\tau}{\mathbf{H}^{\phi}} \right), \quad (9)$$

$$\mathbf{H}^{\phi} \leftarrow \mathbf{H}^{\phi} + \eta_H \left(\frac{\beta \cdot \mathbf{H}^{\phi^{(-1/2)}}}{\|\mathbf{H}\|^{-1/2}_{1/2}} + \sum_{\tau} \overset{\downarrow\phi}{\mathbf{W}^{\tau T}} (\overset{\leftarrow\tau}{\mathbf{V}} - \mathbf{\Lambda}) \right), \quad (10)$$

where $\mathbf{H}^{\phi^{(-1/2)}}$ denotes rasing each element of $\mathbf{H}^{\phi}$ to the power $-1/2$.

The algorithm is summarized in the following steps:

1. Initialize $\mathbf{W}^{\phi}$ and $\mathbf{H}^{\tau}$ randomly.
2. Update $\mathbf{W}^{\tau}$ according to (9).
3. Set any negative elements in $\mathbf{W}^{\tau}$ to zero.
4. Normalize $\mathbf{W}^{\tau}$ according to (6).
5. If cost function is reduced accept $\mathbf{W}$ else reject update, reduce step size, η_W, and go to step 2.
6. Update $\mathbf{H}^{\phi}$ according to (10).
7. Set any negative elements in $\mathbf{H}^{\phi}$ to zero.
8. If cost function is reduced accept $\mathbf{H}$ else reject update, reduce step size, η_H, and go to step 6.
9. Repeat from step 2 until convergence.

Since the SNMF2D requires adaption of the step sizes η_W and η_H it does not converge as fast as the NMF2D. Consequently, it can be convenient to use NMF2D to find a starting point as opposed to starting from random.

3 Experimental Results

The NMF2D and SNMF2D methods were tested on two pieces of music; a computer generated piece and a real piano recording. Both pieces of music are the

first bars of Mozart's Sonata in C-major K.545 Allegro. The computer generated music was created using a single sampled piano note.

We resampled the music to a sample rate of 16 kHz and analyzed it by the short time Fourier transform with a 2048 point Hanning windowed FFT and 50% overlap. This gave us 61 FFT slices. We grouped the spectrogram bins into 303 logarithmically spaced frequency bins in the range from 100 Hz to 8 kHz with 48 bins per octave, which corresponds to four times the resolution of the equal tempered musical scale. Then, we performed the NMF2D and SNMF2D analysis of the log-frequency magnitude spectrogram. We used one factor, d = 1, corresponding to modelling one instrument. We empirically chose to use eight convolutive components in time, $\tau = \{0, \ldots, 7\}$, corresponding to 50 milliseconds. We chose to use 128 convolutive components in pitch, $\phi = \{0, \ldots, 127\}$, corresponding to more than 2 1/2 octaves. For the SNMF2D analysis we chose a sparseness parameter, $\beta = 0.1$.

The results of the analyses of the computer generated music and the real music are shown in Figure 1 and Figure 2 respectively. The figures show the score for the piece of music and log amplitude of $\mathbf{H}^{\phi}$, $\mathbf{W}^{\tau}$, and the modelled spectrogram, $\mathbf{\Lambda}$, for the NMF2D and SNMF2D analysis. In all analyses we note, that the instrument time-frequency signature, $\mathbf{W}^{\tau}$, reveals a clear harmonic structure as would be expected. The time-pitch signature in the NMF2D analyses show that part of the harmonics and background noise are modelled here in addition to the notes which are played. In the SNMF2D analyses however, only the notes are present in the time-pitch signature.

4 Discussion

The results show, that the proposed SNMF2D method is very well suited for automatic transcription of polyphonic music. The analysis of the computer generated music in Figure 1 shows that all notes can be identified. The same is the case for the real piano recording shown in Figure 2, except for the second low C, which is missing in the sparse analysis, because the note is played very softly and is removed by the sparsity constraint.

In our examples, we do not translate the result of the analysis into an actual score. The frequencies of the notes can be found by computing the fundamental frequency of the time-frequency signature of the instrument, e.g. by fitting a harmonic model and compare this to the relative pitch of the notes in $\mathbf{H}$. In order to perform automatic transcription based on the SNMF2D decomposition, it should be combined with a suitable knowledge based system, which can perform quantization, rejection of spurious notes etc.

The NMF2D has a large number of parameters, depending on the number of time shifts τ and pitch shifts ϕ. In the analyzed examples above, the number of elements in $\mathbf{V}$ was 18438 while $\mathbf{W}$ and $\mathbf{H}$ had a total of 10232 elements, thus the number of parameters in the model is more than half that of the data. Even when the number of parameters in the model is greater than that of the data, i.e. the problem is overcomplete, the SNMF2D model can give reasonable results because of the sparsity constraint on $\mathbf{H}^{\phi}$.

In all the experiments, the estimated model fitted the data very well; in all our experiments, both using NMF2D and SNMF2D, the explained variation was above 95%. It is also worth to mention, that for the four second music examples described here the algorithm took less than one minute to run. The complexity of the algorithm is approximately linear i the length of the signal.

In these examples the analyzed instrument was a piano. Our studies show that the method also works very well when analyzing other instruments such as guitars, flutes and violins. For many real instruments, the assumption that all notes can be modelled by a time-frequency signature which is simply shifted on the time- and frequency axes does hold reasonably well. However, instruments for which it does not hold, can e.g. be modelled by multiple factors which each span a shorter range of the scale.

One aspect which is not directly modelled is the length of the notes. Since each note is modelled by a fixed time-frequency signature, the note length is captured in the time-pitch signature, where the notes appear as lines. However, sometimes the lines of a single tone is broken and can thus falsely be identified

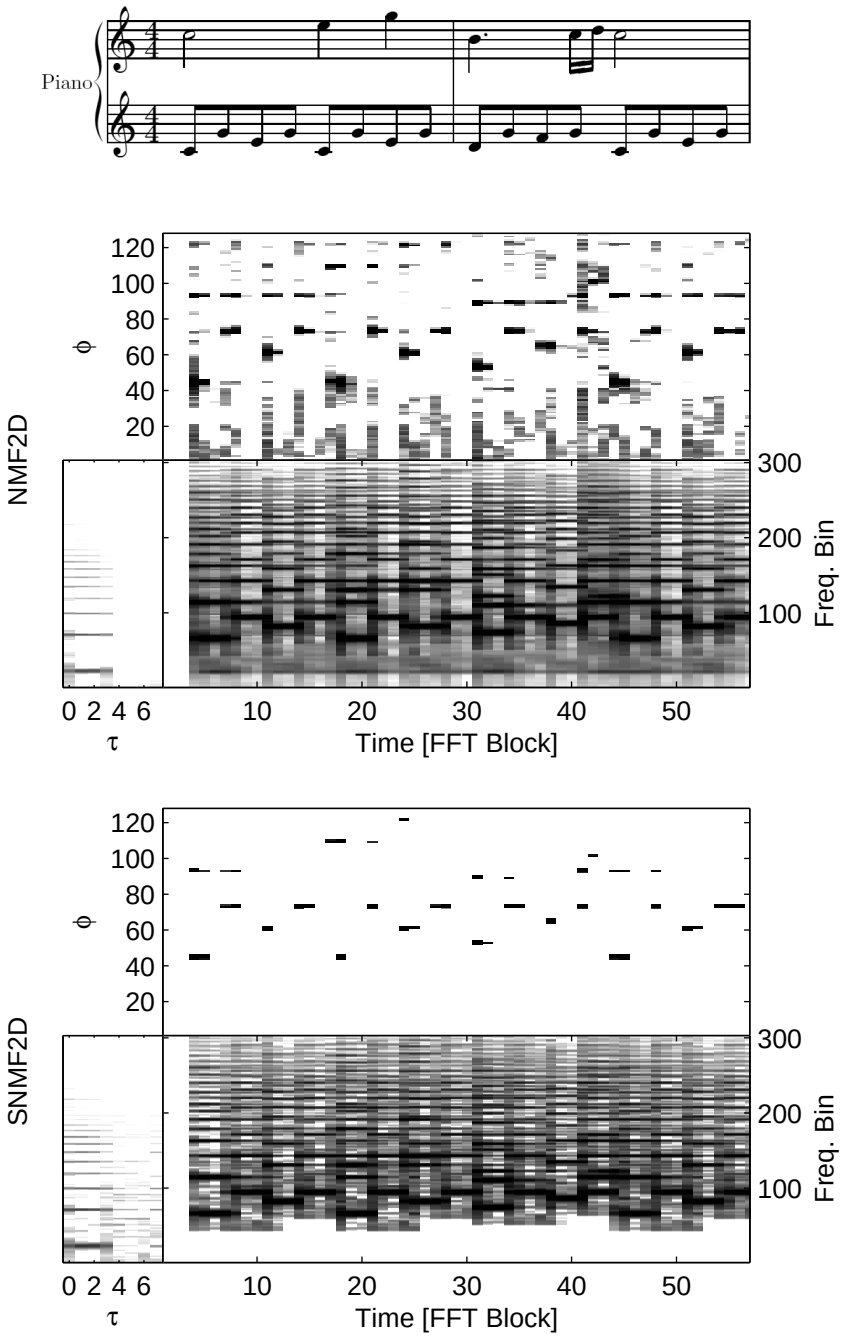

Figure 1: Computer generated music: Musical score and analysis by NMF2D and SNMF2D. The two analysis plots show the log amplitude of $\mathbf{H}^{\phi}$ (top), $\mathbf{W}^{\tau}$ (bottom left), and $\mathbf{\Lambda}$ (bottom right).

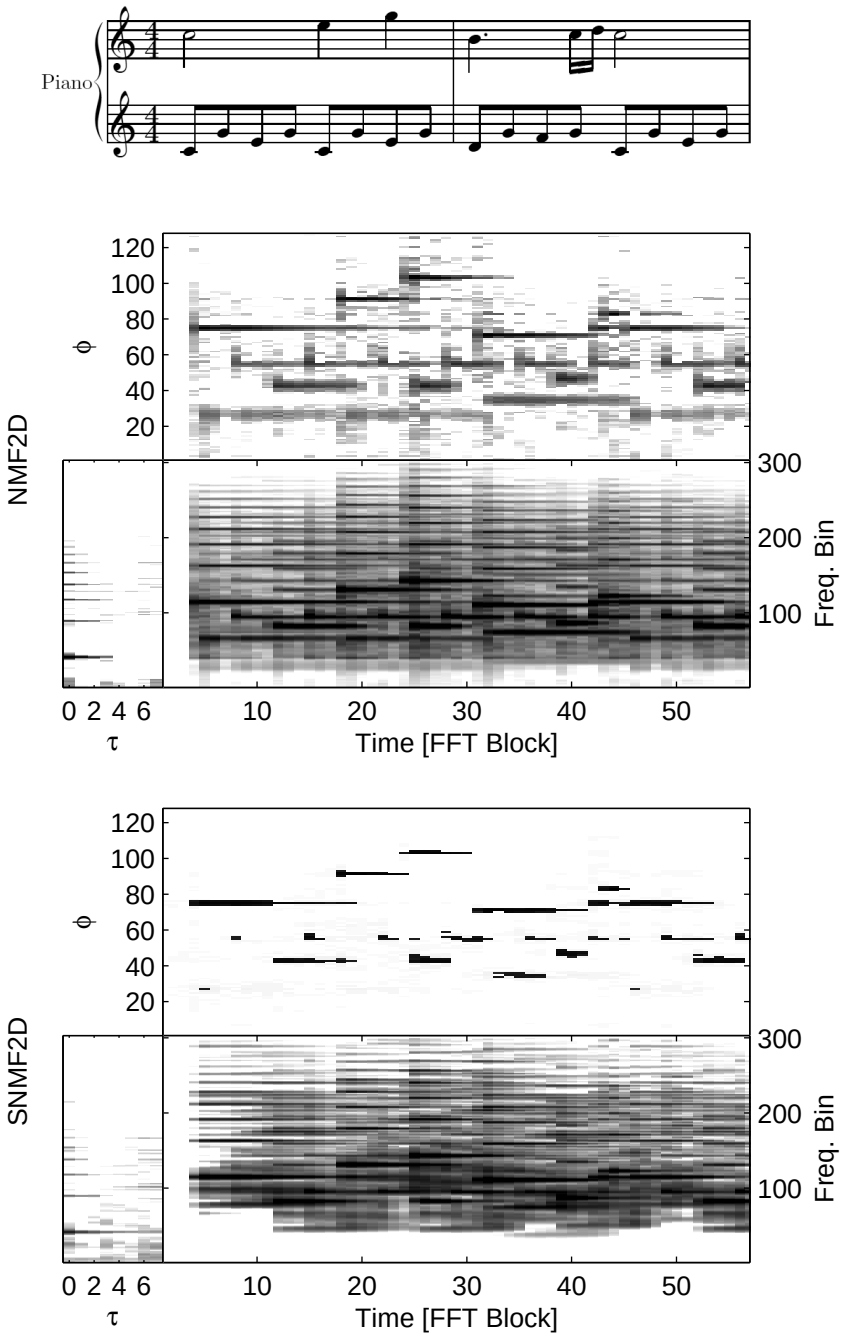

Figure 2: Recorded piano music: Musical score and analysis by NMF2D and SNMF2D. The two analysis plots show the log amplitude of $\mathbf{H}^{\phi}$ (top), $\mathbf{W}^{\tau}$ (bottom left), and $\mathbf{\Lambda}$ (bottom right).

as two notes. The model could gain from including a more direct way of modelling note lengths. Also the method could be improved by modelling the way the time-frequency profile of an instrument changes from the low notes to the high notes.

Compared to the method proposed by Smaragdis and Brown [8], the sparse NMF2D method has the advantage that it directly provides a representation which can be used to determine the notes played by an instrument, and that it can identify notes which do not occur in isolation. Furthermore, the method can be used to model multiple simultaneous instruments.

Initial studies show, that the SNMF2D algorithm also works well for transcription of simple computer generated multiple instrument polyphonic music, but cannot directly be used for transcription of complex real polyphonic music with multiple instruments. Future work will focus on extending the NMF2D model on multiple instrument transcription using supervised approaches.

5 Conclusion

We have presented a method for automatic transcription of polyphonic music based on a novel sparse non-negative matrix factor 2-D deconvolution model. We have demonstrated the method for two pieces of polyphonic piano music with good results. SNMF2D seems to be a promising method for automatic music transcription.

References

[1] D. Ellis and D. Rosenthal. Mid-level representations for computational auditory scene analysis. In *Proc. Workshop on Computational Auditory Scene Analysis, Int. Joint Conf. on Artif. Intell., Montreal*, 1995.

[2] P.O. Hoyer. Non-negative sparse coding. *Neural Networks for Signal Processing, 2002. Proceedings of the 2002 12th IEEE Workshop on*, pages 557–565, 2002.

[3] P.O. Hoyer. Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, 2004.

[4] A.P. Klapuri. Automatic music transcription as we know it today. *Journal of New Music Research*, 33(3):269–282, 2004.

[5] M. Mørup and M. N. Schmidt. Nonnegative matrix factor 2-D deconvolution (nmf2d) and sparse nmf2d (snmf2d). Technical report, Institute for Mathematical Modelling, Technical University of Denmark, 2005.

[6] M. D. Plumbley, S. A. Abdallah, J. P. Bello, M. E. Davies, G. Monti, and M. B. Sandler. Automatic music transcription and audio source separation. *Cybernetics and Systems*, 33(6):603–627, 2002.

[7] M. N. Schmidt and M. Mørup. Nonnegative matrix factor 2-D deconvolution for blind single channel source separation. In *ICA2006*, 2005.

[8] P. Smaragdis and J.C. Brown. Non-negative matrix factorization for polyphonic music transcription. *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 177–180, October 2003.