REAL-TIME POLYPHONIC MUSIC TRANSCRIPTION WITH NON-NEGATIVE MATRIX FACTORIZATION AND BETA-DIVERGENCE

Arnaud Dessein, Arshia Cont, Guillaume Lemaitre
IRCAM – CNRS UMR 9912, Paris, France
{dessein,cont,lemaitre}@ircam.fr

ABSTRACT

In this paper, we investigate the problem of real-time polyphonic music transcription by employing non-negative matrix factorization techniques and the β-divergence as a cost function. We consider real-world setups where the music signal arrives incrementally to the system and is transcribed as it unfolds in time. The proposed transcription system is addressed with a modified non-negative matrix factorization scheme, called non-negative decomposition, where the incoming signal is projected onto a fixed basis of templates learned off-line prior to the decomposition. We discuss the use of non-negative matrix factorization with the β-divergence to achieve the real-time decomposition. The proposed system is evaluated on the specific task of piano music transcription and the results show that it can outperform several state-of-the-art off-line approaches.

1. INTRODUCTION

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 the problem of multiple-pitch estimation which has been largely investigated for music as well as speech, and for which a wide variety of methods have been proposed [8]. Non-negative matrix factorization has already been used in this context, with off-line approaches [1, 3, 20, 22–24] as well as on-line approaches [4, 6, 7, 17, 21].

Generally speaking, *non-negative matrix factorization* (NMF) is a technique for data analysis where the observed data are supposed to be non-negative [16]. The main philosophy of NMF is to build up these observations in a constructive additive manner, what is particularly interesting when negative values cannot be interpreted (*e.g.* pixel intensity, word occurrence, magnitude spectrum).

In this paper, we employ NMF techniques to develop a real-time system for polyphonic music transcription. This system is thought as a front-end for musical interactions in live performances. Among applications, we are interested in computer-assisted improvisation for instruments such as the piano. We do not discuss such applications in the paper but rather concentrate on the system for polyphonic music transcription and invite the curious reader to visit the companion website [1] for complementary information and additional resources. The proposed system is addressed with an NMF scheme called *non-negative decomposition* where the signal is projected in real-time onto a basis of note templates learned off-line prior to the decomposition.

In this context, the price to pay for the simplicity of the standard NMF is the overuse of templates to construct the incoming signal, resulting in note insertions and substitutions such as octave and harmonic errors. In [6, 7], the issue has been tackled with the standard Euclidean cost by introduction of a sparsity constraint similar to [14]. We here investigate the use of more complex costs by using the β-divergence. This is in contrast to previous systems for real-time audio decomposition which have either considered the Euclidean distance or the Kullback-Leibler divergence. NMF with the β-divergence has recently proved its relevancy for off-line applications in speech analysis [18], music analysis [11] and music transcription [3, 23]. We adapt these approaches to a real-time setup and propose a tailored multiplicative update to compute the decomposition. We also give intuition in understanding how the β-divergence helps to improve transcription. The provided evaluation show that the proposed system can outperform several off-line algorithms at the state-of-the-art.

The paper is organized as follows. In Section 2, we introduce the related background on NMF techniques. In Section 3, we focus on NMF with the β-divergence, provide a multiplicative update tailored to real-time decomposition, and discuss the relevancy of the β-divergence for the decomposition of polyphonic music signals. In Section 4, we depict the general architecture of the real-time system proposed for polyphonic music transcription, and detail the two modules respectively used for off-line learning of note templates and for on-line decomposition of music signals. In Section 5, we perform evaluations of the system for the specific task of piano music transcription.

In the sequel, uppercase bold letters denote matrices, lowercase bold letters denote column vectors, lowercase plain letters 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 $\mathbf{A} \otimes \mathbf{B}$ and $\frac{\mathbf{A}}{\mathbf{B}}$. The element-wise power p of $\mathbf{A}$ is denoted by $\mathbf{A}^{\cdot p}$.

[1] http://imtr.ircam.fr/imtr/Realtime_Transcription

2. RELATED BACKGROUND

This section introduces the NMF model, the standard NMF problem, and the popular multiplicative updates algorithm used to solve it. We then present the relevant literature in sound recognition with NMF.

2.1 NMF model

The NMF model is a low-rank approximation 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:

$$\mathbf{V} \approx \mathbf{W}\mathbf{H} \tag{1}$$

In this model, the multivariate data 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 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* of the corresponding column of $\mathbf{V}$ into this basis.

2.2 Standard problem and multiplicative updates

The standard NMF model of Equation 1 provides an approximate factorization $\mathbf{W}\mathbf{H}$ of $\mathbf{V}$. The aim is then to find the factorization which optimizes a given goodness-of-fit measure called *cost function*. In the standard formulation, the Euclidean distance is used, and the NMF problem amounts to minimizing the following cost function subject to non-negativity of both $\mathbf{W}$ and $\mathbf{H}$:

$$\frac{1}{2}\|\mathbf{V} - \mathbf{W}\mathbf{H}\|_F^2 = \frac{1}{2}\sum_j \|\mathbf{v}_j - \mathbf{W}\mathbf{h}_j\|_2^2 \tag{2}$$

For this particular cost function, factors $\mathbf{W}$ and $\mathbf{H}$ can be computed with the popular *multiplicative updates* introduced in [16]. These updates are derived from a gradient descent scheme with judiciously chosen steps, as follows:

$$\mathbf{H} \leftarrow \mathbf{H} \otimes \frac{\mathbf{W}^T\mathbf{V}}{\mathbf{W}^T\mathbf{W}\mathbf{H}} \qquad \mathbf{W} \leftarrow \mathbf{W} \otimes \frac{\mathbf{V}\mathbf{H}^T}{\mathbf{W}\mathbf{H}\mathbf{H}^T} \tag{3}$$

The updates are applied in turn until convergence, and ensure both non-negativity and decreasing of the cost, but not necessarily local optimality of factors $\mathbf{W}$ and $\mathbf{H}$.

A flourishing literature exists about extensions to the standard NMF problem and their algorithms [5]. These extensions can be thought of in terms of modified cost functions (*e.g.* using divergences or adding penalty terms), of modified constraints (*e.g.* imposing sparsity), and of modified models (*e.g.* using tensors). For example, the cost function defined in Equation 2 is often replaced with the Kullback-Leibler divergence for which specific multiplicative updates have been derived [16].

2.3 Applications in sound recognition

NMF algorithms have been applied to various problems in vision, sound analysis, biomedical data analysis and text classification among others [5]. In the context of sound analysis, 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 $\mathbf{w}_i$ at the j-th time-frame.

NMF has already been used in the context of polyphonic music transcription (*e.g.* see [1, 22]). Several problem-dependent extensions have been developed to this end such as a source-filter model [24], an harmonic constraint [20], an harmonic model with temporal smoothness [3], or an harmonic model with spectral smoothness [23]. These approaches rely in general on the off-line nature of NMF, but some authors have used NMF in an on-line setup.

A real-time system to identify the presence and determine the pitch of one or more voices is proposed in [21]. This system is also adapted for sight-reading evaluation of solo instrument in [4]. Concerning automatic transcription, a similar system is used in [17] for transcription of polyphonic music, and in [19] for drum transcription. A real-time system for polyphonic music transcription with sparsity considerations is proposed in [6]. The approach is further developed in [7] for real-time coupled multiple-pitch and multiple-instrument recognition. Yet, all these approaches are based on NMF with the Euclidean distance or the Kullback-Leibler divergence. We discuss the use of the more general β-divergence as a cost function and its relevancy for decomposition of music signals in Section 3.

3. NON-NEGATIVE DECOMPOSITION WITH THE BETA-DIVERGENCE

In this section, we define the β-divergence, give some of its properties, and review its use as a cost function for NMF. We finally formulate the non-negative decomposition problem with the β-divergence and give multiplicative updates tailored to real-time for solving it.

3.1 Definition and properties of the beta-divergence

The β-divergences form a parametric family of distortion functions [9]. For any $\beta \in \mathbb{R}$ and any points $x, y \in \mathbb{R}_{++}$, the β-divergence from x to y is defined as follows:

$$d_\beta(x|y) = \frac{1}{\beta(\beta-1)}\left(x^\beta + (\beta-1)y^\beta - \beta x y^{\beta-1}\right) \tag{4}$$

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:

$$d_{\beta=0}(x|y) = d_{IS}(x|y) = \frac{x}{y} - \log\frac{x}{y} - 1 \tag{5}$$

$$d_{\beta=1}(x|y) = d_{KL}(x|y) = x\log\frac{x}{y} + y - x \tag{6}$$

For $\beta = 2$, the β-divergence specializes to the widely used half squared Euclidean distance:

$$d_{\beta=2}(x|y) = d_E(x|y) = \frac{1}{2}(x-y)^2 \tag{7}$$

Concerning their properties, all β-divergences are non-negative and vanish iff $x = y$. However, they are not necessary distances in the strict terms since they are not symmetric and do not satisfy the triangle inequality in general. A property of the β-divergences relevant to the present work is that for any scaling factor $\lambda \in \mathbb{R}_{++}$ we have:

$$d_\beta(\lambda x|\lambda y) = \lambda^\beta d_\beta(x|y) \tag{8}$$

We discuss further the interest of this scaling property for decomposition of polyphonic music signals in Section 3.3.

3.2 NMF and the beta-divergence

The β-divergence was first used with NMF to interpolate between the Euclidean distance and the Kullback-Leibler divergence [15]. Starting with the scalar divergence in Equation 4, a matrix divergence can be constructed as a *separable* divergence, *i.e.* by summing the element-wise divergences. The NMF problem with the β-divergence then amounts to minimizing the following cost function subject to non-negativity of both $\mathbf{W}$ and $\mathbf{H}$:

$$\mathcal{D}_\beta(\mathbf{V}|\mathbf{WH}) = \sum_{i,\,j} d_\beta(v_{ij}\,|\,[\mathbf{WH}]_{ij}) \tag{9}$$

For $\beta = 2$, this cost function specializes to the cost defined in Equation 2 for standard NMF.

As for standard NMF, several algorithms including multiplicative updates have been derived for NMF with the β-divergence and its extensions [5, 15]. The β-divergence has also proved its relevancy as a cost function for audio off-line applications in speech analysis [18], music analysis [11] and music transcription [3, 23].

3.3 Problem formulation and multiplicative update

We now formulate the problem of non-negative decomposition with the β-divergence. We assume that $\mathbf{W}$ is a fixed dictionary of note templates onto which we seek to decompose the incoming signal $\mathbf{v}$ as $\mathbf{v} \approx \mathbf{Wh}$. The problem is therefore equivalent to minimizing the following cost function subject to non-negativity of $\mathbf{h}$:

$$\mathcal{D}_\beta(\mathbf{v}|\mathbf{Wh}) = \sum_{i} d_\beta(v_i\,|\,[\mathbf{Wh}]_i) \tag{10}$$

To solve this problem, we update $\mathbf{h}$ iteratively by using a vector version of the corresponding multiplicative update proposed in the literature [5, 15]. As $\mathbf{W}$ is fixed, we never apply its respective update. The algorithm thus amounts to repeating the following update until convergence:

$$\mathbf{h} \leftarrow \mathbf{h} \otimes \frac{\mathbf{W}^T\left((\mathbf{Wh})^{.\beta-2} \otimes \mathbf{v}\right)}{\mathbf{W}^T(\mathbf{Wh})^{.\beta-1}} \tag{11}$$

This scheme ensures non-negativity of $\mathbf{h}$, but not necessarily local optimality. Unfortunately, no proof has been found yet to show that the cost function is non-increasing under this update for a general parameter β, even if it has been observed in practice [11]. However, even if such theoretical issues need to be investigated further, the simplicity of this scheme makes it suitable for real-time applications and gives good results in practice.

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

$$\mathbf{h} \leftarrow \mathbf{h} \otimes \frac{\left(\mathbf{W} \otimes (\mathbf{v}\mathbf{e}^T)\right)^T (\mathbf{Wh})^{.\beta-2}}{\mathbf{W}^T(\mathbf{Wh})^{.\beta-1}} \tag{12}$$

where $\mathbf{e}$ is a vector full of ones. This helps to reduce the computational cost of the update scheme as the matrix $\left(\mathbf{W} \otimes (\mathbf{v}\mathbf{e}^T)\right)^T$ needs only to be computed once.

The scaling property in Equation 8 may give an insight in understanding the relevancy of the β-divergence in our context. For $\beta = 0$, the Itakura-Saito divergence is the only β-divergence to be scale-invariant as it was remarked in [11]. 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. Considering music signals, this amounts to giving the same importance to high-energy and to low-energy frequency components. When $\beta > 0$, more emphasis is put on the frequency components of higher energy, and the emphasis augments with β. When $\beta < 0$, the effect is the converse. In our context of music decomposition, we try to reconstruct an incoming music 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. The parameter β can thus help to control this trade-off.

4. GENERAL ARCHITECTURE OF THE SYSTEM

In this section, we present the real-time system proposed for polyphonic music transcription. The general architecture is shown schematically in Figure 1. The right side of the figure represents the music signal arriving in real-time, and its decomposition onto notes whose descriptions are provided *a priori* to the system as templates. These templates are learned off-line, as shown on the left side of the figure, and constitute the dictionary used during real-time decomposition. We describe the two modules hereafter.

4.1 Note template learning

The learning module aims at building a dictionary $\mathbf{W}$ of note templates onto which the polyphonic music signal is projected during the real-time decomposition phase.

In the present work, we use a simple rank-one NMF with the standard cost function as a learning scheme. We suppose that the user has access to isolated note samples of the instruments to transcribe, from which the system learns characteristic templates. The whole note sample k is first

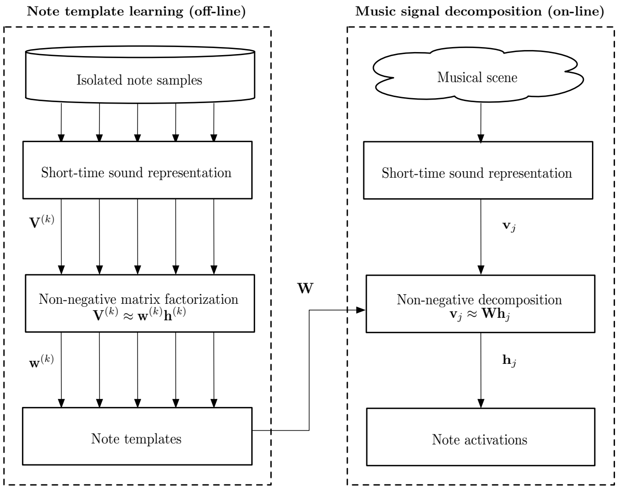

Note template learning (off-line)
Isolated note samples
Short-time sound representation
$\mathbf{V}^{(k)}$
Non-negative matrix factorization
$\mathbf{V}^{(k)} \approx \mathbf{w}^{(k)}\mathbf{h}^{(k)}$
$\mathbf{w}^{(k)}$
Note templates
$\mathbf{W}$
Music signal decomposition (on-line)
Musical scene
Short-time sound representation
$\mathbf{v}_j$
Non-negative decomposition
$\mathbf{v}_j \approx \mathbf{W}\mathbf{h}_j$
$\mathbf{h}_j$
Note activations

Figure 1. Schematic view of the general architecture.

processed in a short-time sound representation supposed to be non-negative and approximatively additive (*e.g.* a short-time magnitude spectrum). The representations are stacked in a matrix $\mathbf{V}^{(k)}$ where each column $\mathbf{v}_j^{(k)}$ is the sound representation of the j-th time-frame. We then solve standard NMF with $\mathbf{V}^{(k)}$ and a rank of factorization $r = 1$, using the multiplicative updates in Equation 3. This learning scheme simply gives a template $\mathbf{w}^{(k)}$ for each note sample (the information in the row vector $\mathbf{h}^{(k)}$ is discarded).

4.2 Music signal decomposition

Having learned the templates, we stack them in columns to form the dictionary $\mathbf{W}$. The problem of real-time transcription then amounts to projecting the incoming music signal $\mathbf{v}_j$ onto $\mathbf{W}$, where $\mathbf{v}_j$ share the same representational front-end as the note 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 successive activations of the different notes in the music signal. Following the discussion in Section 3, we learn the vectors $\mathbf{h}_j$ by employing the β-divergence as a cost function and the multiplicative update tailored to real-time decomposition in Equation 11.

As such, the system reports only a frame-level activity of the notes. Some post-processing is thus needed to extract more information about the eventual presence of the notes, and provide a symbolic representation of the music signal for transcription. This post-processing potentially includes activation thresholding, onset detection, temporal modeling, etc. It is however not thoroughly discussed in this paper where we use a simple threshold-based detection followed by a minimum duration pruning.

5. EVALUATION AND RESULTS

In this section, we evaluate the system on polyphonic transcription of piano music. We provide a subjective evaluation with musical excerpts synthesized from MIDI references. We also perform an objective evaluation with a real piano music database and standard evaluation metrics.

5.1 Subjective evaluation

As sample examples, we transcribed two musical excerpts synthesized from MIDI references with real piano samples from the Real World Computing (RWC) database [12].

For the non-negative decomposition, β was set to 0.5 since this value was shown optimal for music transcription in [23] and provided good results in our tests. The threshold for detection was set to 2 and no minimum duration pruning was applied. For the dictionary, one note template was learned and max-normalized for each of the 88 notes of the piano using corresponding samples taken from RWC. We used a simple short-time magnitude spectrum representation, 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 decomposition was computed in real-time simulation under MATLAB on a 2.40 GHz laptop with 4.00 Go of RAM, and was about three times faster than real-time.

The results of the decomposition are shown in Figure 2. Figures 2(a) and 2(b) depict the piano-roll representations of the two piano excerpts. The ground-truth references are represented with rectangles and the transcriptions with black dots. Overall, this shows that the system is able to match reliably the note templates to the music signals. During note attacks, more templates are used due to transients but some post-processing such as minimum duration pruning would help to remove these errors. We also remark a tendency to shorten sustained notes which may be due to a different spectral content during note releases.

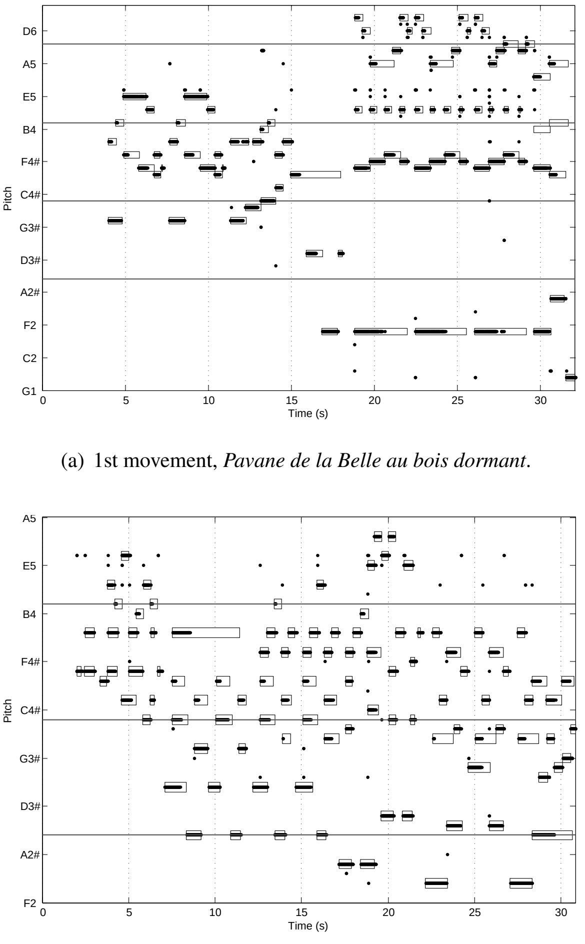
5.2 Objective evaluation

For a more rigorous evaluation, we considered the standards of the Music Information Retrieval Evaluation eXchange (MIREX) [2] and focused on two subtasks: (1) a frame-level estimation of the present events in terms of musical pitch, and (2) a note-level tracking of the present notes in terms of musical pitch, onset and offset times.

For the evaluation dataset, we chose the MIDI-Aligned Piano Sounds (MAPS) database [10]. MAPS contains real recordings of piano pieces with ground-truth references. We selected 25 pieces and truncated each of them to 30 s.

Concerning parameters, β was set to 0.5. The thresholds for detection were set empirically to 1 and 2 for the frame and note levels respectively. The minimum duration for pruning was set to 50 ms. The templates were learned from MAPS with the same representation front-end as above. This algorithm is referenced by BND.

In addition, we tested the system with the standard Euclidean decomposition algorithm referenced by END, and with the sparse algorithm of [14] with projection onto the cone of sparsity $s = 0.9$. For these two algorithms, the detection thresholds were set to 2 and 4 for the frame and note levels respectively. To compare results, we also performed the evaluation for two off-line systems at the state-of-the-art: one based on NMF but with an harmonic model



(a) 1st movement, *Pavane de la Belle au bois dormant*.

(b) 4th movement, *Les entretiens de la Belle et de la Bête*.

Figure 2. Transcription of two piano excerpts from *Ma mère l'Oye, Cinq pièces enfantines pour piano à quatre mains* (1908-1910), Maurice Ravel (1875-1937).

and spectral smoothness [23], and another one based on a sinusoidal analysis with a candidate selection exploiting spectral features [25].

We report the evaluation results per algorithm in Tables 1 and 2 at the frame and note levels respectively. Standard evaluation metrics from the MIREX are used as described in [2]: precision $\mathcal{P}$, recall $\mathcal{R}$, F-measure $\mathcal{F}$, accuracy $\mathcal{A}$, total error $\mathcal{E}_{tot}$, substitution error $\mathcal{E}_{subs}$, missed error $\mathcal{E}_{miss}$, false alarm error $\mathcal{E}_{fa}$, mean overlap ratio $\mathcal{M}$. At the note level, the subscripts 1 and 2 represent respectively the onset-based and the onset/offset-based results.

Overall, the results show that the proposed real-time system performs comparably to the state-of-the-art off-line algorithms of [23, 25]. Using the β-divergence, the system BND even outperforms the other algorithms. The sparse algorithm of [14] reduces insertions and substitutions, but augments the number of missed notes so that it actually does not perform better than the standard scheme END. The standard Euclidean cost also shows its limits for transcription where more complex costs with the β-divergence give better results. We finally remark that the mean overlap ratio scores corroborate the observation that sustained notes tend to be shortened.

Alg.	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{F}$	$\mathcal{A}$	$\mathcal{E}_{tot}$	$\mathcal{E}_{subs}$	$\mathcal{E}_{miss}$	$\mathcal{E}_{fa}$
BND	63.9	67.3	**65.5**	**48.7**	**58.9**	11.9	20.8	26.2
END	55.3	58.6	**56.9**	**39.8**	71.4	17.3	24.1	29.9
[14]	58.5	55.2	56.8	39.7	**67.1**	16.8	28.0	22.3
[23]	61.0	66.7	63.7	46.8	65.6	10.4	22.9	32.3
[25]	60.0	70.8	**65.0**	**48.1**	**60.0**	16.3	12.8	30.8

Table 1. Frame-level transcription results per algorithm.

Alg.	$\mathcal{P}_1$	$\mathcal{R}_1$	$\mathcal{F}_1$	$\mathcal{A}_1$	$\mathcal{M}_1$	$\mathcal{P}_2$	$\mathcal{R}_2$	$\mathcal{F}_2$	$\mathcal{A}_2$
BND	75.5	67.1	**71.1**	**55.1**	**56.7**	30.0	26.6	**28.2**	**16.4**
END	57.9	58.2	**58.1**	**40.9**	53.9	21.4	21.6	**21.5**	**12.0**
[14]	57.2	56.3	56.8	39.6	**54.1**	21.0	20.7	20.8	11.6
[23]	58.1	73.7	**65.0**	**48.1**	**57.7**	20.7	26.3	**23.2**	**13.1**
[25]	33.0	58.8	42.3	26.8	55.1	11.6	20.7	14.9	8.0

Table 2. Note-level transcription results per algorithm.

6. CONCLUSION

This paper addressed the problem of real-time polyphonic music transcription by employing NMF techniques. We discussed the use of the β-divergence as a cost function for non-negative decomposition tailored to real-time transcription. The obtained results show that the proposed system can outperform state-of-the-art off-line approaches, and are encouraging for further development.

A problem in our approach is that templates are inherently considered as stationary. One way to tackle this is to consider representations that capture variability over a short time-span as in [7]. We could also combine NMF with a state representation and use templates for each state.

The template learning method can be further improved by using extended NMF problems and algorithms to learn one or more templates for each note. Such issues have not been developed but interesting perspectives include learning sparse or harmonic templates. Using the β-divergence during template learning in our experience did not improve the results. Further considerations are needed on this line.

In a live performance setup such as ours, the templates can be directly learned from the corresponding instrument. Yet in other setups, the issue of generalization must be carefully considered and will be discussed in future work. We think of considering adaptive templates by adapting an approach proposed in [13] to real-time decomposition.

We would like also to improve the robustness against noise, by keeping information from the activations during template learning, or by using noise templates as in [7]. In addition, we want to develop more elaborate sparsity controls than in [6, 7, 14]. In our approach, sparsity is controlled implicitly during decomposition. Yet in some applications, specially for complex problems such as auditory scene analysis, controlling explicitly sparsity becomes crucial. A forthcoming paper will address this issue.

Last but not least, the proposed system is currently under development for the Max/MSP real-time computer music environment and will be soon available for free download on the companion website.

7. ACKNOWLEDGMENTS

This work was partially funded by a doctoral fellowship from the UPMC (EDITE). The authors would like to thank C. Yeh and R. Badeau for their valuable help, V. Emiya for kindly providing the MAPS database, as well as P. Hoyer and E. Vincent for sharing their source code.

8. REFERENCES

[1] S. A. Abdallah and M. D. Plumbley. Polyphonic music transcription by non-negative sparse coding of power spectra. In *Proc. of ISMIR 2004*, pages 318–325, Barcelona, Spain, October 2004.

[2] M. Bay, A. F. Ehmann, and J. S. Downie. Evaluation of multiple-$F0$ estimation and tracking systems. In *Proc. of ISMIR 2009*, Kobe, Japan, October 2009.

[3] N. Bertin, R. Badeau, and E. Vincent. 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, March 2010.

[4] C.-C. Cheng, D. J. Hu, and L. K. Saul. Nonnegative matrix factorization for real time musical analysis and sight-reading evaluation. In *Proc. of ICASSP 2008*, pages 2017–2020, Las Vegas, NV, USA, March/April 2008.

[5] A. Cichocki, R. Zdunek, A. H. Phan, and S.-i. Amari. *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*. Wiley-Blackwell, 2009.

[6] A. Cont. Realtime multiple pitch observation using sparse non-negative constraints. In *Proc. of ISMIR 2006*, Victoria, Canada, October 2006.

[7] A. Cont, S. Dubnov, and D. Wessel. Realtime multiple-pitch and multiple-instrument recognition for music signals using sparse non-negative constraints. In *Proc. of DAFx 2007*, Bordeaux, France, September 2007.

[8] A. de Cheveigné. *Computational Auditory Scene Analysis: Principles, Algorithms and Applications*, chapter Multiple $F0$ Estimation, pages 45–72. Wiley-IEEE Press, 2006.

[9] S. Eguchi and Y. Kano. Robustifying maximum likelihood estimation. Technical report, Institute of Statistical Mathematics, Tokyo, Japan, 2001.

[10] V. Emiya, R. Badeau, and B. David. Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle. *IEEE Transactions on Audio, Speech and Language Processing*, To appear.

[11] C. Févotte, N. Bertin, and J.-L. Durrieu. Nonnegative matrix factorization with the Itakura-Saito divergence: with application to music analysis. *Neural Computation*, 21(3):793–830, March 2009.

[12] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. RWC music database: popular, classical, and jazz music databases. In *Proc. of ISMIR 2002*, pages 287–288, October 2002.

[13] M. Heiler and C. Schnörr. Learning sparse representations by non-negative matrix factorization and sequential cone programming. *J. of Machine Learning Research*, 7:1385–1407, July 2006.

[14] P. O. Hoyer. Non-negative matrix factorization with sparseness constraints. *J. of Machine Learning Research*, 5:1457–1469, November 2004.

[15] R. Kompass. A generalized divergence measure for nonnegative matrix factorization. *Neural Computation*, 19(3):780–791, 2007.

[16] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, October 1999.

[17] B. Niedermayer. Non-negative matrix division for the automatic transcription of polyphonic music. In *Proc. of ISMIR 2008*, pages 544–549, Philadelphia, PA, USA, September 2008.

[18] P. D. O'Grady and B. A. Pearlmutter. Discovering speech phones using convolutive non-negative matrix factorisation with a sparseness constraint. *Neurocomputing*, 72:88–101, 2008.

[19] J. Paulus and T. Virtanen. Drum transcription with non-negative spectrogram factorisation. In *Proc. of EUSIPCO 2005*, Antalya, Turkey, September 2005.

[20] S. A. Raczyński, N. Ono, and S. Sagayama. Harmonic nonnegative matrix approximation for multipitch analysis of musical sounds. In *Proc. of ASJ Autumn Meeting*, pages 827–830, September 2007.

[21] F. Sha and L. K. Saul. Real-time pitch determination of one or more voices by nonnegative matrix factorization. In *Proc. of NIPS 2004*, volume 17, pages 1233–1240, Cambridge, MA, USA, 2005.

[22] P. Smaragdis and J. C. Brown. Non-negative matrix factorization for polyphonic music transcription. In *Proc. of WASPAA 2003*, pages 177–180, New Paltz, NY, USA, October 2003.

[23] E. Vincent, N. Bertin, and R. Badeau. Adaptive harmonic spectral decomposition for multiple pitch estimation. *IEEE Transactions on Audio, Speech and Language Processing*, 18(3), March 2010.

[24] T. Virtanen and A. Klapuri. Analysis of polyphonic audio using source-filter model and non-negative matrix factorization. In *Proc. of NIPS Workshop AMAC 2006*, 2006.

[25] C. Yeh. *Multiple fundamental frequency estimation of polyphonic recordings*. PhD thesis, Université Pierre et Marie Curie, Paris, France, June 2008.