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 depict our system submitted to MIREX 2010 for the multiple fundamental frequency estimation and tracking evaluation task [1] . The system is actually tailored to real-time transcription of piano music for live performances. We consider real-world setups where the music signal arrives incrementally to the system and is transcribed as it unfolds in time. The problem of real-time transcription 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 employ non-negative matrix factorization with the β-divergence to achieve the real-time decomposition.

1. INTRODUCTION

In general terms, *non-negative matrix factorization* (NMF) is a technique for data analysis where the observed data are supposed to be non-negative [6]. 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:

$$\mathbf{V} \approx \mathbf{WH} \tag{1}$$

The aim is thus to find the factorization which optimizes a given goodness-of-fit measure, called *cost function*, such as the Euclidean distance used in the standard formulation.

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 invite the curious reader to visit the companion website [2] 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.

The paper is organized as follows. In Section 2, we focus on NMF with the β-divergence and provide a multiplicative update tailored to real-time decomposition. In Section 3, we depict the general architecture of the real-time system proposed for polyphonic 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 $\mathbf{A}^{\cdot p}$.

2. NON-NEGATIVE DECOMPOSITION WITH THE BETA-DIVERGENCE

In this section, we define the β-divergence and then discuss 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.

2.1 Definition of the beta-divergence

The β-divergences form a parametric family of distortion functions [3]. 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:

$$d_\beta(x|y) = \frac{1}{\beta(\beta-1)}\left(x^\beta + (\beta-1)y^\beta - \beta x y^{\beta-1}\right) \tag{2}$$

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:

$$d_{\beta=0}(x|y) = d_{IS}(x|y) = \frac{x}{y} - \log\frac{x}{y} - 1 \tag{3}$$

$$d_{\beta=1}(x|y) = d_{KL}(x|y) = x \log\frac{x}{y} + y - x \tag{4}$$

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

$$d_{\beta=2}(x|y) = d_E(x|y) = \frac{1}{2}(x-y)^2 \tag{5}$$

2.2 NMF and the beta-divergence

Starting with the scalar divergence in Equation 2, a matrix divergence can be constructed as a *separable* divergence, *i.e.* by summing the element-wise divergences. The NMF

[1] This paper is an extended abstract of our ISMIR 2010 paper [2].

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

problem with the β-divergence then amounts to minimizing the following cost function subject to non-negativity of both $\mathbf{W}$ and $\mathbf{H}$:

$$\mathcal{D}_\beta(\mathbf{V}|\mathbf{WH}) = \sum_{i,\,j} d_\beta(v_{ij}\,|\,[\mathbf{WH}]_{ij}) \tag{6}$$

2.3 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 $\mathbf{h}$:

$$\mathcal{D}_\beta(\mathbf{v}|\mathbf{Wh}) = \sum_{i} d_\beta(v_{i}\,|\,[\mathbf{Wh}]_{i}) \tag{7}$$

To solve this problem, we update $\mathbf{h}$ iteratively by using a vector version of the corresponding multiplicative update proposed in the literature [1]. As $\mathbf{W}$ is fixed, we never apply its respective update. The algorithm thus amounts to repeating the following update until convergence:

$$\mathbf{h} \leftarrow \mathbf{h} \otimes \frac{\mathbf{W}^T\left((\mathbf{Wh})^{\cdot\beta-2} \otimes \mathbf{v}\right)}{\mathbf{W}^T(\mathbf{Wh})^{\cdot\beta-1}} \tag{8}$$

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:

$$\mathbf{h} \leftarrow \mathbf{h} \otimes \frac{\left(\mathbf{W} \otimes (\mathbf{v}\mathbf{e}^T)\right)^T(\mathbf{Wh})^{\cdot\beta-2}}{\mathbf{W}^T(\mathbf{Wh})^{\cdot\beta-1}} \tag{9}$$

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.

3. 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 and the tuning of the system hereafter.

3.1 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 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 of standard NMF. This learning scheme simply gives a template $\mathbf{w}^{(k)}$ for each note sample.

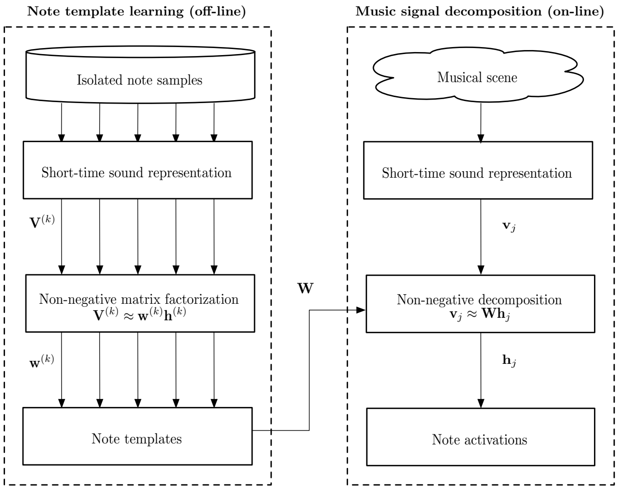
Note template learning (off-line)
Music signal decomposition (on-line)
Isolated note samples
Musical scene
Short-time sound representation
Short-time sound representation
$\mathbf{V}^{(k)}$
$\mathbf{v}_j$
W
Non-negative matrix factorization $\mathbf{V}^{(k)} \approx \mathbf{w}^{(k)}\mathbf{h}^{(k)}$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

Figure 1. Schematic view of the general architecture.

3.2 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 $\mathbf{v}_j \approx \mathbf{Wh}_j$ 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. 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 given in Equation 8.

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. We use here a simple threshold-based detection followed by a minimum duration pruning.

3.3 System tuning

The proposed system is actually tailored to piano music transcription. The system was tuned manually on a test dataset of 25 musical excerpts from the MIDI-Aligned Piano Sounds (MAPS) database [4].

For template learning, we used isolated piano samples from MAPS and from the Real World Computing (RWC) database [5]. One template was learned for each of the 88 notes of the piano using corresponding isolated samples.

More precisely, for each note of the piano, we concatenated nine samples corresponding to three different pianos: two pianos from MAPS and one piano from RWC, and to three different dynamics: *piano*, *mezzo forte* and *forte*. A standard rank-one NMF with normalization of the encodings coefficients was run on each of the concatenated samples to learn the respective note templates.

We employed a simple short-time magnitude spectrum representation, with a frame size of $50\,\mathrm{ms}$ leading to 630 samples at a sampling rate of $12600\,\mathrm{Hz}$, and computed with a zero-padded Fourier transform of 1024 bins. The frames were windowed with a Hann function, and the hop-size was set to $25\,\mathrm{ms}$ for template learning and refined to $10\,\mathrm{ms}$ for decomposition.

For the non-negative decomposition, β was set to 0.5. The threshold for detection was set empirically to 0.02 and the minimum duration for pruning was set to $50\,\mathrm{ms}$.

4. CONCLUSION

In this paper, we have presented a real-time system for polyphonic music transcription based on NMF techniques with the β-divergence as a cost function. The interested reader can find additional information on the companion website and in our ISMIR 2010 paper [2]. 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.

5. 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 to test the proposed system, as well as P. Hoyer and E. Vincent for sharing their source code for comparison.

6. REFERENCES

[1] 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.

[2] A. Dessein, A. Cont, and G. Lemaitre. Real-time polyphonic music transcription with non-negative matrix factorization and beta-divergence. In *Proc. of ISMIR 2010*, Utrecht, Netherlands, August 2010.

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

[4] 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.

[5] 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.

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