This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMM.2017.2674603, IEEE Transactions on Multimedia

Instrument Learning and Sparse NMD for Automatic Polyphonic Music Transcription

Antonello Rizzi, *Member IEEE,* Mario Antonelli and Massimiliano Luzi, *Student Member IEEE,*

***Abstract*—In this paper, an Automatic Music Transcription (AMT) algorithm based on a supervised Non-negatve Matrix Decomposition (NMD) is discussed. In particular, a novel approach for enhancing the sparsity of the solution is proposed. It consists of a two-step processing in which the NMD is solved joining a ℓ_2 regularization and a threshold filtering. In the first step, the NMD is performed with the ℓ_2 regularization in order to get an overall selection of the notes most likely appearing in the monotimbral musical excerpt. In the second step, a threshold filtering followed by another ℓ_2 regularized NMD are repeatedly performed in order to progressively reduce the dictionary matrix and to refine the notes transcription. Furthermore, a user-oriented instrument learning procedure has been conceived and proposed. The proposed AMT system has been tested upon the dataset collected by the LabROSA laboratories considering the transcription of three different pianos. Moreover, it has been validated through a comparison with a regularized NMD and with three open source AMT software. The results prove the effectiveness of the proposed two-step processing in enhancing the sparsity of the solution and in improving the transcription accuracy. Moreover, the proposed system shows promising performance in both multi-F0 and note tracking tasks, obtaining in most tests better transcription accuracy than the competing algorithms.**

***Index Terms*—Automatic Music Transcription, Spectrogram Factorization, Non-negative Matrix Decomposition, Sparse Coding, Non-monotone Optimization.**

I. Introduction

MUSIC Information Retrieval (MIR) has got an increasing attention by the multimedia research community and several systems have been proposed, especially in the Content Based Retrieval (CBR) applications. Examples are the genre classification of musical excerpts [1], the retrieval of similar songs [2], the search of the occurrence of a musical excerpt in a musical database [3], the retrieval of song titles using a query by humming, a query by singing or, in general, a query by melody approach [4].

In this context, an Automatic Music Transcription (AMT) system can help in the task of recognizing the contextual musical information needed for the CBR systems. In fact, AMT algorithms aim to retrieve the musical score of a composition directly from its audio recording by detecting which notes are played, when they are played and their duration. An effective music transcription algorithm can be useful and critically important in query by melody music retrieval in order to both retrieve the melody singed by the user and to automatically generate a database of melodies [5]. Beyond the application in MIR systems and in the CBR problems cited above, AMT can be useful in other multimedia contexts. Indeed, it can help and speed up the composition job, both for expert and novice musicians; AMT can have a great value in providing a musical score to some musical styles where actually no score exists, such as music from the oral traditions or jazz [6]; furthermore, AMT can be useful in recording studio where the MIDI files generated by the transcription system can be used for driving any Virtual Instrument Software (VST).

Although the transcription task is considered to be a solved problem for monophonic music [7], [8], the transcription of polyphonic music is much more complicated because of the overlapping and interference of the harmonics related to the superimposed notes and this problem remains still open in the literature. First attempts for polyphonic transcription were done in 1977 by Moorer [9], who used comb filters and autocorrelation for transcription of duets, and by Piszczalski & Galler [10]. Since then, several works have been proposed using both temporal and frequency domain analysis, such as Rossi et al. [11], Dixon [12] and Bello et al. [13]. Most of the studies are based on frequency representation and various techniques were employed. Klapuri [14] has estimated multiple fundamental frequencies from spectral peaks using a computational model of human peripheral auditory system and a subsequent cancellation of each harmonic pattern from the spectrum; in Ryynänen & Klapuri work [15] they added Hidden Markov Model post-processing in order to get the melody lines from the f_0 estimation. Some approaches used psycho-acoustic models to perform transcription, such as Slaney & Lion [16], Ellis [17] and Klapuri [18]. Marolt [19] proposed a connectionist approach using the output of adaptive oscillators as a training set for a bank of neural networks in order to transcribe piano recordings. Another important work based on a classification approach has been proposed by Poliner & Ellis [20], who used a supervised classification system based on Support Vector Machines for piano transcription. Other works, such as those of Davy et al. [21] and Yoshii et al. [22], rely on probabilistic inference using Bayesian methods and in general Hidden Markov Models were used for note tracking due to their effectiveness in modeling sequential structures [20], [23].

In the last years, spectrogram factorization performed by Non-negative Matrix Factorization (NMF) or probabilistic latent component analysis (PLCA) got significant attention in music transcription community. Examples of NMF can be found in [24]–[27], whereas Benetos & Dixon focused principally on PLCA proposing an AMT system able to cope with different instruments and with tuning deviations [23], [28], [29]. Smaragdis & Brown firstly proposed the NMF approach in [30]an it is a non-supervised algorithm aiming

The authors are with the Department of Information Engineering, Electronics and Telecommunications (DIET), University of Rome “La Sapienza”, Via Eudossiana 18, 00184 Rome, Italy (e-mail: antonello.rizzi@uniroma1.it; mario.antonelli@gmail.com; massimiliano.luzi@uniroma1.it).

to both identify which notes are played and their temporal activities. The key aspect of the NMF approach consists in its capacity of fulfilling the physical behavior of polyphonic music. In fact, as a polyphonic musical excerpt is composed of the superimposition of different single notes, the NMF searches for the superimpositions of different single atoms, each one related to one of the played notes. However, the raw unsupervised NMF suffers of several drawbacks. Firstly, it requires to know a priori the rank of the decomposition, *i.e.* the exact number of notes to search. Secondly, it requires the notes to be played monophonically at least once in order to successfully separate all the single notes. Thirdly, it can learn some meaningless atoms. Fourthly, it requires a further recognition algorithm to be performed in order to associate each detected atom to the related note label.

The previous issues have been addressed in literature by means of a supervised or semi-supervised NMF, also called Non-negative Matrix Decomposition (NMD) [31]–[34]. The NMD approach uses a fixed dictionary matrix $\mathbf{D}$ built considering at least one atom for each of the playable notes. This results in a fully labeled and meaningful dictionary allowing to automatically separate and recognize all the notes. In the supervised version the dictionary is built offline from the samples of monophonic notes [31], whereas in the semi-supervised NMD the dictionary is firstly initialized to a generic value and then is tuned online [33].

Besides the improvements gained, the NMD approach continues to detect some shadow activities related to notes not actually played. For this reason, the latest research efforts have been focused on the development of transcription systems aiming to enhance the sparsity of the solution. Three kinds of sparsity have been addressed. Firstly, the sparsity on the notes activation has been typically imposed since only a little set of all the playable notes are actually played in a musical excerpt. Secondly, a group sparsity approach has been proposed in which the atoms of the dictionary are collected in groups and the sparsity is imposed on the overall activation of these groups [27], [34], [35]. This is because the number of notes actually played simultaneously is significantly smaller than the number of all the possible combination of notes. Finally, also a harmonic sparsity has been imposed to each atom of the dictionary, forcing to zero all the frequency bins except those related to the fundamental frequency and its partials [33].

In this paper, a novel approach for enhancing the sparsity of the solution of a supervised NMD algorithm is proposed. Instead of considering only a regularization constraint in the cost function, a two-step processing joining a ℓ_2 regularization and a threshold filtering is proposed. During the first step, the β-divergence cost function with $\beta = 2$ and the ℓ_2 regularization are used in order to perform an overall selection of the atoms most likely appearing in the composition. At the end of this step, a threshold filtering is performed deleting from the dictionary matrix $\mathbf{D}$ all the atoms showing an overall activation lower than a suited threshold. During the second step, the NMD and the threshold filtering are repeated until a suited stop condition is verified. Herein, the more effective β-divergence with $\beta = 1$ and the ℓ_2 regularization are used to furtherly refine the notes detection. In order to deal with the non-negative constraint, the NMD is performed by solving a constrained optimization problem by means of a suited non-monotone modification of the projected gradient method. Furthermore, a user-oriented instrument learning procedure has been conceived for allowing the end-user to build a dictionary upon its own instrument. The whole software has been implemented in *C#* language and represents a complete AMT system receiving a WAVE file in input and returning the transcribed score encoded in a MIDI file.

The algorithm has been tested considering two virtual pianos corresponding to a *Steinway Grandpiano* and a *Yamaha C2* and one real piano played by a *Yamaha Disklavier* system. The dataset consists of the MIDI and the WAVE files collected by Poliner & Ellis [20] of the LabROSA laboratories [36]. Firstly, the impact of each stage of the algorithm on the final transcription accuracy has been investigated. Then, the effectiveness of the proposed two-step processing in enhancing both the sparsity and the accuracy has been highlighted through a comparison with a regularized NMD. Finally, the proposed algorithm has been validated through a direct comparison with the open source AMT software related to the works of Marolt [19], Vincent [32] and Benetos & Dixon [29]. The results show promising performance with the proposed system obtaining almost always better results both in the multi-F0 and in the note tracking tasks. In addition, the time complexity is in line with state of art transcribing systems.

The paper is organized as follows: in the next section the whole AMT algorithm is presented. In Sect. III the testing procedure, the adopted metrics and the performed tests are shown and commented. Finally, Sect. IV contains some conclusive remarks.

II. The Proposed AMT Algorithm

A. Solving Method

Let consider the magnitude spectrogram $\mathbf{Y} \in \mathbb{R}_{+}^{M \times N}$ of the input audio signal $y[n]$ where M is the number of frequency bins and N is the number of temporal frames. Given the factorization rank $K \ll M$, the NMF algorithm aims to approximate the matrix $\mathbf{Y}$ by the product of the two non-negative matrices $\mathbf{D} \in \mathbb{R}_{+}^{M \times K}$ and $\mathbf{X} \in \mathbb{R}_{+}^{K \times N}$ such that:

$$\mathbf{Y} \simeq \mathbf{DX} \tag{1}$$

In such a decomposition, $\mathbf{D}$ and $\mathbf{X}$ are the dictionary and the activation matrices, respectively. In particular, the columns $\boldsymbol{d}_k$ of $\mathbf{D}$ are called *atoms* and each of them contains the spectral representation of one of the note recognized in the musical piece, whereas the rows of $\mathbf{X}$ represent the temporal activity of these atoms. The non-negative matrices $\mathbf{D}$ and $\mathbf{X}$ are evaluated by minimizing a suited cost function $\mathcal{F}$:

$$\mathbf{D}, \mathbf{X} \leftarrow \min_{\mathbf{D}, \mathbf{X}} \mathcal{F}\left(\mathbf{Y}, \mathbf{DX}\right) \tag{2}$$

Currently, the β-divergence (Eq.(3)) is preferred as cost function since it generalizes several functions mostly used in signal processing, such as the Euclidean distance ($\beta = 2$), the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMM.2017.2674603, IEEE Transactions on Multimedia

Kullback-Lieber (KL) divergence ($\beta = 1$) and the Itakuro-Saito (IS) divergence ($\beta = 0$).

$$\mathcal{B}(\boldsymbol{y}_n, \boldsymbol{z}_n, \beta) = \begin{cases} \sum_m \frac{y_{n_m}}{z_{n_m}} - \log \frac{y_{n_m}}{z_{n_m}} - 1, & \beta = 0 \\ \sum_m y_{n_m} \left(\log \frac{y_{n_m}}{z_{n_m}} - 1 \right) + z_{n_m}, & \beta = 1 \\ \sum_m \frac{y_{n_m}^{\beta} + (\beta-1) z_{n_m}^{\beta} - \beta y_{n_m} z_{n_m}^{\beta-1}}{\beta(\beta-1)}, & \text{o/w} \end{cases} \tag{3}$$

In the above formula, the β-divergence is evaluated for the n-th temporal frame with $\boldsymbol{z}_n = \mathbf{D}\boldsymbol{x}_n$ referring to the current estimation, and y_{n_m}, z_{n_m} referring to the m-th frequency bin of the respective vectors.

When the dictionary matrix $\mathbf{D}$ is fixed, the NMF is reduced to NMD and only the activation matrix $\mathbf{X}$ has to be evaluated. In this case the NMD is performed by solving the optimization problem (2) only for $\mathbf{X}$. Furthermore, it is possible to proceed with a frame by frame analysis instead of estimating the whole matrix $\mathbf{X}$ all at once by means of the well known multiplicative rules [30], [32]. This fact allows to parallelize and to simplify the whole transcription algorithm. For clarity, the optimization problem to solve is expressed as follows:

$$\begin{cases} \boldsymbol{x}_n \leftarrow \min\limits_{\boldsymbol{x}_n} \mathcal{F}\left(\mathbf{y_n}, \mathbf{D}\boldsymbol{x}_n\right) \\ \text{s.t. } \boldsymbol{x}_n \geq \mathbf{0} \end{cases} \quad n = 0, \ldots, N-1 \tag{4}$$

In order to handle non-negative constraints, the problem (4) can be solved by using any constrained optimization algorithm. In the last years, many advanced algorithms have been presented whose implementations are available on internet, such as the *LANCELOT* solver [37]. In the present paper a modification of the projected gradient algorithm has been used; this approach uses the Barzilai-Borwein formulas to compute the step along the anti-gradient direction in combination with a non-monotone line search technique used for evaluating the step along the projected direction [38]. This constrained optimization algorithm has been chosen since non-monotone projected gradient methods are preferred when the feasible set is defined by bounds or non-negative constraints; furthermore, their low memory requirements and low computational cost make them very attractive for large scale problems or time demanding applications if compared to Newton or Quasi-Newton methods [39]. Further details of the algorithm used to solve the optimization problem (4) can be found in [40] and a Matlab version can be downloaded at [41].

B. Spectral Representation

Among the available spectrogram representations, the Constant Q Transform (CQT) has been considered in this work. This kind of spectrogram is generally preferred in musical signal processing due to the logarithmic progression of the frequency bins. Theorized by Youngberg in [42] and adapted to musical processing by Brown in [43], it is characterized by having the frequency bins arranged in the fundamental frequencies of notes, instead of keeping them linearly distributed in the frequency range. In particular, the positions of these bins are given by:

$$f_0^{(p)} = 440 * 2^{p/B}\, Hz \tag{5}$$

where $440\, Hz$ is the pitch of note ‘A’ at the 4-th octave, B is the number of considered bins per octave and p is the position of the p-th note with respect to note ‘A$_{4\text{th}}$’. Furthermore, the CQT uses a variable spectral resolution having an inversely proportional relationship with the increasing frequency. The advantage of this spectral representation consists in a substantial reduction of the number of bins, producing a more effective frequency representation for music processing.

In this work the CQT is evaluated with the Brown algorithm [44], assuming $B = 24$ bins per octave corresponding to a value of $Q = 35$. In order to speed up the CQT computation, a higher bound to the window length fixed in $10\, ms$ has been set up.

C. Instrument Learning Procedure

A user-oriented instrument learning procedure has been conceived in order to build the dictionary matrix $\mathbf{D}$. This procedure aims to investigate the harmonic behavior of the current instrument in order to synthesize a meaningful dictionary from a limited set of monophonic exemplars.

In general, any instrument is characterized by imposing a typical harmonic timbre to the sounds it produces. This harmonic timbre is related to the ratio between the power of the fundamental frequency and those of the related harmonics. Although each playable note could show a different harmonic timbre, it could be recognized a macroscopic harmonic behavior of the instrument that will characterize its acoustic. The proposed learning method aims to retrieve this macroscopic behavior and to use it for building a dictionary matrix tailored upon the current instrument. Thus, the instrument learning procedure is composed of two phases: the harmonic timbre retrieval and the dictionary synthesis.

In order to perform the first phase, it is required that the end-user records a very limited number L of different monophonic notes. A good choice could be the twelve notes belonging to the most used octave of the instrument to learn. For each recording, a unique spectral representation is obtained as the vector resulting from the frame by frame averaging of the related CQT. Then, a peak analysis is performed upon the obtained spectral representation in order to get the power of each harmonic; this power is expressed as the value of the spectral representation at the harmonic peak. Given the l-th note recording, the retrieved powers of the harmonics are arranged in a vector $\boldsymbol{c}_l$ in which the h-th element represents the power of the h-th harmonic and $h = 0$ refers to the fundamental frequency. Finally, the macroscopic harmonic timbre of the instrument is evaluated as the average of the vectors $\boldsymbol{c}_l$:

$$\boldsymbol{c} = \frac{1}{L} \sum_{l=0}^{L-1} \boldsymbol{c}_l \tag{6}$$

The just detected harmonic timbre $\boldsymbol{c}$ can be used for synthesizing a generic audio signal related to every note produced by the instrument. Thus, the signal related to the note of pitch f_0 is expressed as follows:

$$s[n] = \sum_h c_h \sin\left(2\pi h f_0 \frac{n}{T_s}\right) \tag{7}$$

where T_s is the chosen sampling time. Note that the phase information can be neglected since only the magnitude spectrogram is considered for the AMT task.

During the dictionary synthesis phase, the signal (7) is synthesized for each note from ‘C$_{\text{2nd}}$’ to ‘B$_{\text{7th}}$’ and for each of them the related spectral representation is evaluated with the same procedure explained above. The obtained vectors represent the atoms $\boldsymbol{d}_k$ and will populate the columns of $\mathbf{D}$.

This instrument learning procedure aims to provide an answer to some of the challenges claimed by Benetos et al. in [6]. According to [6], one of the principal lack of the current AMT systems is the absence of an end-user application that can help in improving the AMT accuracy. Moreover, there is the general tendency to apply AMT in a general purpose way, not considering the possibility of tailoring the transcribing system on the actual playing instrument. Considering these issues, the instrument learning procedure has been designed to be as user-oriented as possible looking for the best trade-off between accuracy and usability. In fact, although also other works allow the end-user to create a dictionary upon his own instrument, typically their procedures require at least one monophonic exemplar for each playable note [28], [31]. Thus, these procedures could not be considered user-friendly due to the very high number of recordings the user must perform and often a generic dictionary has to be used. Conversely, the proposed instrument learning procedure allows to build meaningful dictionaries starting from a very limited number of recordings. This allows the end-user to build autonomously the dictionary upon his own instruments and consequently to tailor the transcription on the actual playing instrument, obtaining a more instrument-specific transcribing system.

At the same time, the dictionary synthesis phase allows to build the dictionary matrix ensuring a sparse harmonic representation for each atom. In fact, the expression (7) allows to filter out all the noise affecting the recordings and to put a frequency contribution only in correspondence of the bins related to a harmonic component.

D. Pre-processing

With the aim of generalizing and facilitating the recognition and the tracking of the notes, a pre-processing stage is performed before executing the transcription algorithm.

Firstly, a linear normalization of the audio signal is performed in order to set the samples values in the range $[-1, 1]$.

$$y[k] = y[k] / \max_{k=0,\ldots,K-1} \left\{ \left| y[k] \right| \right\} \tag{8}$$

Successively, a Hard Dynamic Range Compression (HDRC) is performed. The HDRC aims to level the intensity of the audio signal in order to limit the loudness differences between louder and softer notes, helping in detecting the softer ones. In order to perform the HDRC, the Root Mean Square (RMS) value is evaluated for each sample considering the preceding temporal window of $10\,ms$ length. If the RMS value is greater than a threshold fixed to the RMS of the whole normalized audio signal, then the sample is attenuated with a multiplicative coefficient inversely proportional to the threshold surplus.

After the HDRC, another linear normalization is performed with the aim of recovering the signal range to $[-1, 1]$.

E. Transcription Procedure

The transcription procedure is performed in two steps, each one involving the solution of a proper NMD of the input spectrogram $\mathbf{Y}$. Jointly with a ℓ_2 regularization and a threshold filtering, the proposed procedure aims to enhance the sparsity of the solution resulting in a more accurate transcription.

1) Step One: During the first step the NMD is solved considering the whole dictionary matrix $\mathbf{D}$ and by minimizing a convex cost function in order to perform a preliminary selection of the most likely active atoms in the composition. In particular, the NMD is solved for each frame by using the method discussed in Sect. II-A considering the Euclidean distance ($\beta = 2$) and a ℓ_2 regularization. For clarity, the optimization problem is summarized in the following:

$$\begin{cases} \boldsymbol{x}_n \leftarrow \min\limits_{\boldsymbol{x}_n} \mathcal{B}(\boldsymbol{y}_n, \mathbf{D}\boldsymbol{x}_n, \beta = 2) + \dfrac{\lambda}{2} \left\| \boldsymbol{x}_n \right\|_2^2 \\ \text{s.t. } \boldsymbol{x}_n \geq \mathbf{0} \\ n = 0, \ldots, N-1 \end{cases} \tag{9}$$

where $\lambda > 0$ is the regularization coefficient.

In the first step, the use of a less effective cost function is preferred for reducing the computational cost of the optimization algorithm. Indeed, during the Step One it is only necessary to perform a first rough selection of the active atoms, whereas the Step Two will accurately detect the activation of the actually played notes. At the end of Step One, the obtained activation matrix $\mathbf{X}$ is linearly normalized in order to generalize the threshold to be used during the threshold filtering of Step Two:

$$x_{kn} = x_{kn} / \max_{k=0,\ldots,K-1} \left\{ \max_{n=0,\ldots,N-1} \{x_{kn}\} \right\} \tag{10}$$

2) Step Two: During the second step a threshold filtering followed by a further NMD are iterated until a proper stop condition is verified. The aim of this step is to progressively increase the sparsity of $\mathbf{X}$ and to refine the estimation of the residual atoms activation.

The key operation is the threshold filtering and the consequent reduction of the dictionary matrix $\mathbf{D}$. Given a row of the activation matrix $\mathbf{X}$ representing the temporal activity of one atom, the whole row is deleted if its maximum value is lower than a suited threshold X_{th}. This way, if the threshold is properly set, the rows describing the activation of actually played notes are left untouched, whereas most of the rows containing only noise are completely filtered out. Similarly, all the atoms $\boldsymbol{d}_k$ related to the filtered rows of $\mathbf{X}$ are deleted from the dictionary matrix $\mathbf{D}$, obtaining a reduced version $\tilde{\mathbf{D}}$ of the dictionary. Being $\tilde{\mathbf{D}}_d$ the dictionary at the d-th sub-iteration of Step Two, the whole step is repeated until the dictionary matrix does not change between two consecutive sub-iterations, indicating that it is no more reducible.

The cost function considered for Step Two is the more effective KL divergence ($\beta = 1$) and again a ℓ_2 regularization is adopted. In this case, the objective function is locally adaptive since, besides the linear distance term, there is a nonlinear (logarithmic) term, weighted by the target value which properly measures the relevance of the local error. Although the nonlinearities of the KL divergence upshift the computational

Algorithm 1 Pseudo-code of the two-step processing

```
Split $\mathbf{Y}$ in slices of 10 seconds length $\mathbf{Y}^{(s)}$
for all Slices $s$ do
    Step One
    $\mathbf{X}_0^{(s)} \leftarrow \mathrm{NMD}(\mathbf{Y}^{(s)}, \mathbf{D}, \beta = 2, \ell_2)$
    Step Two
    Set $\tilde{\mathbf{D}}_0 = \mathbf{D}$ and $d = 1$
    while $\tilde{\mathbf{D}}_d \neq \tilde{\mathbf{D}}_{d-1}$ do
        $\tilde{\mathbf{D}}_d \leftarrow \mathrm{threshold}(\tilde{\mathbf{D}}_{d-1}, \mathbf{X}_{d-1}^{(s)})$
        $\mathbf{X}_d^{(s)} \leftarrow \mathrm{NMD}(\mathbf{Y}^{(s)}, \tilde{\mathbf{D}}_d, \beta = 1, \ell_2)$
        $d$++
    end while
end for
```

cost of the optimization algorithm, this higher computational demand is counterbalanced by the smaller dimension of the reduced dictionary $\tilde{\mathbf{D}}_d$. Moreover, the better performance of this cost function allows to improve furtherly the notes detection. In order to apply the projected gradient method described in Sect. II-A, the gradient of the KL divergence is evaluated as follows:

$$\nabla_{\boldsymbol{x}_n} \mathcal{B}\left(\boldsymbol{y}_n, \tilde{\mathbf{D}}_d \boldsymbol{x}_n, \beta = 1\right) = \tilde{\mathbf{D}}_d^T \left(\mathbf{1} - \frac{\boldsymbol{y}_n}{\tilde{\mathbf{D}}_d \boldsymbol{x}_n}\right) \tag{11}$$

where $\mathbf{1}$ is a vector with all the elements equal to one.

In order to further improve the effectiveness of the proposed approach, the input spectrogram is split in slices of 10 seconds length and the two-step processing is applied separately to each slice. This solving procedure comes from the observation that the average number of the total amount of played notes in a musical piece is significantly lower than the average number of actually played notes in a ten seconds length excerpt (42 and 8 in the LabROSA dataset, respectively). Thus, the proposed two-step processing is able to perform a very wide reduction of the dictionary matrix. In fact, ideally it is possible to obtain a final dictionary of about 8 atoms for each slice starting from the 72 atoms of the full dictionary. In conclusion, the pseudo-code of the proposed AMT system is shown in Algorithm 1, whereas the impact of the second step in enhancing the sparsity of $\mathbf{X}$ is shown in Fig.1.

Differently to other NMD approaches where the ℓ_1/ℓ_2 regularization constraints and/or the group sparsity are considered, the proposed AMT aims to improve the overall sparsity of $\mathbf{X}$ by means of an iterative reduction of the dictionary matrix $\mathbf{D}$. In fact, although the ℓ_1/ℓ_2 regularization constraints succeed in forcing to zero the activation of most of the actually not played atoms, the activation matrix shows some "shadow" activity for some not played notes, resulting in false positive and substitution errors. This shadow activity is attributable to noise effects and to the harmonic interference between played notes due to the sharing of some harmonics between them. The proposed two-step procedure, instead, succeeds in selecting only the actually played notes resulting in a more effective filtering of the shadow effects, in a more sparse activation matrix $\mathbf{X}$ and in a better transcription accuracy. Furthermore, the solving method discussed in Sect. II-A jointly with the parallel processing and the progressive reduction of $\mathbf{D}$ allows to perform the NMD in a very efficient way.

A similar dictionary recuction procedure can be found in [31] in which the authors performs a pre-selection of the active atoms by means of a peak analysis executed on the spectrum of the current frame to transcribe. However, their procedure is performed *a priori* only once, causing the risk to prematurely cancel some atoms related to notes actually played or to let survive many unwanted atoms. Conversely, the proposed dictionary reduction is performed *a posteriori* by analyzing the detected notes activity. Thus, the estimated temporal activity of each note allows to better identify which atoms are actually active and which are not, getting a more accurate reduction of the dictionary matrix with respect to a *a priori* selection. Moreover, the selection procedure is repeatedly performed during the Step Two, allowing to maximize the reduction of the dictionary and to progressively improve the notes detection.

F. Post-processing: Note Tracking

In order to complete the transcription task, a post-processing procedure performing the note tracking and the final encoding in a MIDI file is needed.

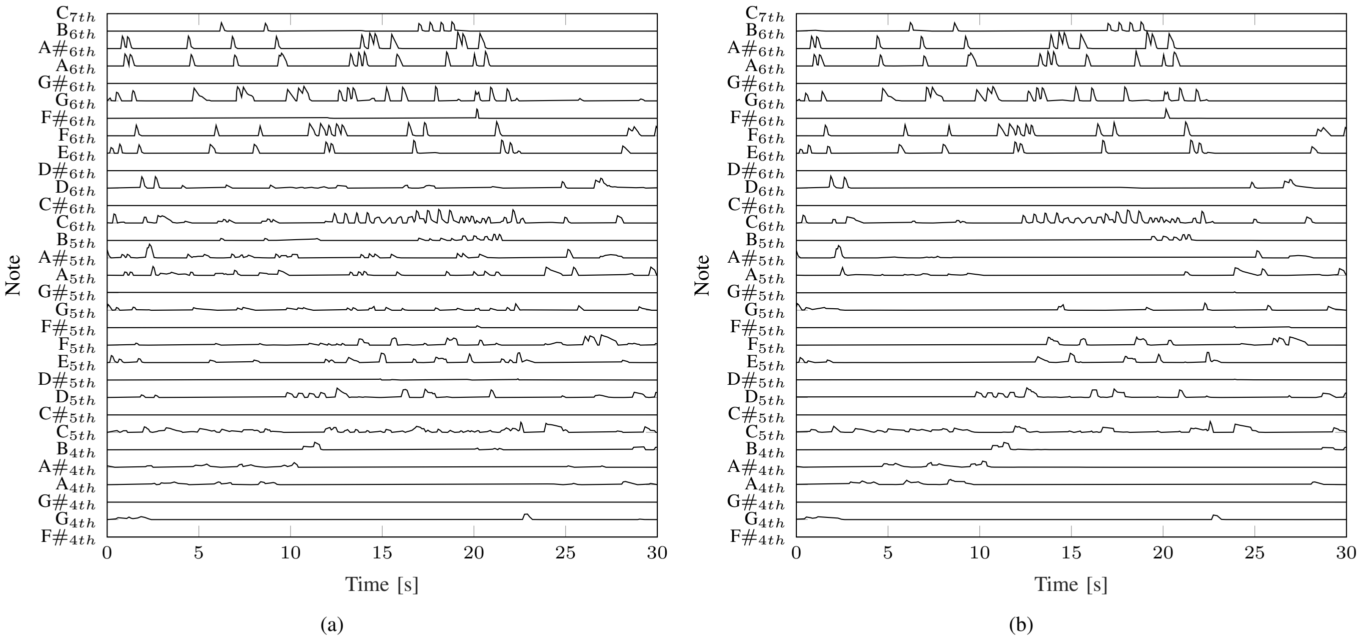
In this work the post-processing has been implemented by means of the Hidden Markov Model (HMM) note tracking method. This approach has been already succesfully used in the literature, *e.g.* by Poliner & Ellis [20] and Benetos & Dixon [28]; it has been chosen because it guarantees the best trade-off between accuracy and generalization capability for different instruments. In particular, the technique proposed by Benetos & Dixon in [28], [29] has been implemented. In the following this method is briefly reviewed.

A two state HMM is considered for each note, where the hidden states refer to *note on* and *note off* conditions. Using the rows of $\mathbf{X}$ as symbol emission sequences of the HMM, the method aims to estimate the maximum likelihood state sequence *note on*/*note off* for each note by means of the Viterbi algorithm. The transition probabilities have been estimated by analyzing the sequence of activation and deactivation of each note in the MIDI files belonging to the LabROSA dataset [36]. Instead, the emission probabilities have been inferred from the detected activation matrix performing the following sigmoid smoothing on the rows of $\mathbf{X}$:

$$\mathcal{P}(x_{kn}|q_{kn} = on) = \frac{1}{1 + e^{-(x_{kn} - \gamma)\theta}} \tag{12}$$

$$\mathcal{P}(x_{kn}|q_{kn} = off) = 1 - \frac{1}{1 + e^{-(x_{kn} - \gamma)\theta}} \tag{13}$$

where n and k refer to the n-th temporal frame and to the k-th note, respectively, q_{kn} is the current state, x_{kn} is the temporal activity, whereas γ and θ controls the offset and the sensitivity of the smoothing. In particular, γ affects the minimum activation value that will be considered as *note on* and θ smooths the transition between *note on* and *note off* states. A higher value of γ will result in a greater probability to discard the notes with a lower activation, whereas a higher value of θ increases the number of transitions between the two states. By tuning γ and θ it is possible to maximize the performance on the note onset and note offset detection.



(a)

(b)

Fig. 1. Transcription of 30 seconds of Mozart K.333 performed considering only the Step One (a) and the complete two-step processing (b). It is noticeable the filtering performed by the second step on the residual shadow activities of Step One (*e.g.* notes C_{5th}, G_{5th} and D_{6th}).

III. Performance Evaluation

A. Testing Procedure

In order to achieve an accurate and rigorous performance evaluation, the following scheme is applied. The ground truth score of the composition to transcribe is firstly encoded in a MIDI file. In order to obtain a time aligned recording, the MIDI file is read and played by a suitable VST. This recording is then encoded in a one channel PCM WAVE file sampled at $8\,KHz$ with $16\,bps$ and it is transcribed by the AMT software. Once the transcription has been performed, the AMT software generates an output MIDI file encoding the transcribed musical score. Thus, having both the original MIDI file and the transcribed one, the performance metrics are evaluated considering a direct comparison between the two MIDIs.

The proposed AMT system has been tested principally for piano transcription because of the great amount of widely polyphonic musical excerpts available for this instrument. Several tuning tests have been performed and the best results have been obtained with the parameters listed in TABLE I.

TABLE I
Algorithm Parameters

CQT		**Algorithm**		**HMM**	
B	Q	λ	X_{th}	γ	θ
24	35	10^{-6}	0.2	0.18	75

The dataset is composed of the MIDI files collected by the LabROSA laboratories [36]. It counts thirty musical excerpts of one minute length, split in two sets: the Training Set counting twenty MIDIs and the Test Set counting ten MIDIs. In this work, the entire LabROSA dataset has been used for estimating the HMM parameters, whereas only the ten compositions belonging to the Test Set have been used for the performance evaluation.

Three different kind of pianos have been considered. The first one and the second one are synthesized through two VST software simulating a *Steinway Grandiano* and a *Yamaha C2*, respectively. The third one is a real piano played by a *Yamaha Disklavier* system. The ten MIDIs of the LabROSA Test Set have been played by each of the three pianos obtaining a total of thirty recordings of one minute length for a total of thirty minutes of transcribed music. The recordings of the *Yamaha Disklavier* have been conducted by the LabROSA laboratories and the related WAVE files can be downloaded at [36].

The instrument learning procedure has been performed for both the *Steinway Grandpiano* and the *Yamaha C2* and two different dictionaries have been built. In particular, the twelve notes belonging to the 4-th octave starting from 'C_{4th}' to 'B_{4th}' have been used for the learning procedure of both the pianos. The transcription campaign has been performed considering the two dictionaries separately in order to highlight the generalization capability of the algorithm upon different dictionaries.

Four tests have been performed. In the first one, the impact of pre-processing, of Step One and of Step Two on the transcription accuracy has been investigated. In the second test, the proposed two-step processing have been compared with both the ℓ_1 and the ℓ_2 regularized NMD in order to highlight the improved sparsity and transcription accuracy of the proposed algorithm. In the third test, the performances of the proposed AMT system have been compared with those of three open source AMT software with the aim of validating the effectiveness of the proposed system with respect to state of art. Finally, the algorithm has been tested upon a violin in order to highlight its natural capability of dealing with different instruments.

B. Metrics

In this work the performance metrics defined in [20] and officially approved by the MIREX [45] are considered. In the following these metrics are briefly reviewed.

1) Multi-F0 Evaluation: The frame based multi-F0 metrics were introduced by Poliner & Ellis in [20] and they are used in the MIREX multiple-F0 evaluation task. The output and the ground truth scores are aligned, divided in $10\,ms$ length frames and compared frame by frame.

The main metric is the overall transcription accuracy defined by the Dixon formula:

$$Acc = \frac{\mathrm{ct}}{\mathrm{ct} + \mathrm{fn} + \mathrm{fp}} \tag{14}$$

where ct is the number of correctly transcribed notes, fn is the number of false negative errors and fp is the number of false positive errors.

The Dixon accuracy can be rewritten as:

$$Acc = \frac{\sum_n N_{ct}[n]}{\sum_n \left(N_{ct}[n] + N_{fn}[n] + N_{fp}[n]\right)} \tag{15}$$

where $N_{ct}[n]$, $N_{fn}[n]$ and $N_{fp}[n]$ are the total number of correctly transcribed notes, the total number of false negative errors and the total number of false positive errors at the n-th frame, respectively. Acc ranges in $[0, 1]$, with 1 representing the perfect transcription.

Furthermore, Poliner & Ellis have defined several error metrics in order to identify the cause of the wrong note detection: the measure of substitution errors E_{subs}; the measure of false negative errors E_{fn}; the measure of false positive errors E_{fp}.

$$E_{subs} = \frac{\sum_n \left(\min\left\{N_{ref}[n], N_{out}[n]\right\} - N_{ct}[n]\right)}{\sum_n N_{ref}[n]} \tag{16}$$

$$E_{fn} = \frac{\sum_n \max\left\{0, N_{ref}[n] - N_{out}[n]\right\}}{\sum_n N_{ref}[n]} \tag{17}$$

$$E_{fp} = \frac{\sum_n \max\left\{0, N_{out}[n] - N_{ref}[n]\right\}}{\sum_n N_{ref}[n]} \tag{18}$$

$$E_{tot} = E_{subs} + E_{fn} + E_{fp} \tag{19}$$

where $N_{ref}[n]$ and $N_{out}[n]$ are the total number of active notes at the n-th frame in the ground truth score and in the transcribed score, respectively.

2) Note Tracking Evaluation: This metric is used for the MIREX note tracking task. A note is considered correctly transcribed if its onset is within $\pm\,50\,ms$ of the related ground truth onset. The formula (14) is used again in order to define the overall accuracy measure. Furthermore, Precision P, Recall R and the F-measure F_m are defined as follows:

$$P = \frac{\mathrm{ct}}{\mathrm{ct} + \mathrm{fp}} \qquad R = \frac{\mathrm{ct}}{\mathrm{ct} + \mathrm{fn}} \qquad F_m = \frac{2PR}{P + R} \tag{20}$$

where P, R and F_m measures the amount of false positive errors, of false negative errors and the balance between false positives and false negatives, respectively.

C. Algorithm Testing

1) Comparison between Algorithm Stages: In order to evaluate the impact of each stage (HDRC, Step One and Step Two) on the transcription accuracy, four tests have been performed. In the first test only the Step One has been considered; in the second test the HDRC has been added; during the third test only the Step One and the Step Two have been performed; finally, the full algorithm has been evaluated in the fourth test. By analyzing the achieved performances, it has been possible to highlight how each of the algorithm stages affects the final accuracy. The obtained results are shown in TABLE II.

TABLE II
COMPARISON BETWEEN ALGORITHM STAGES

(a) Dictionary *Steinway*

	Multi-F0					Note Tracking			
Stages	Acc	E_{sub}	E_{fn}	E_{fp}	E_{tot}	Acc	P	R	F_m
Step One	34,5	4,1	57,3	5,5	66,8	61,0	91,3	64,7	75,3
HDRC + Step One	52,0	7,5	29,5	13,3	50,1	68,3	85,4	77,1	80,7
Step One + Step Two	38,0	3,5	54,0	6,0	63,4	63,9	93,3	66,7	77,3
Full Algorithm	54,6	5,5	29,6	12,3	47,4	72,1	90,5	77,6	83,3

(b) Dictionary *Yamaha C2*

	Multi-F0					Note Tracking			
Stages	Acc	E_{sub}	E_{fn}	E_{fp}	E_{tot}	Acc	P	R	F_m
Step One	33,6	4,2	58,4	4,8	67,5	60,6	91,2	64,4	75,0
HDRC + Step One	51,1	7,6	30,6	12,6	50,8	68,4	85,8	77,0	80,9
Step One + Step Two	37,7	3,8	53,9	6,0	63,7	63,1	92,7	66,2	76,7
Full Algorithm	55,1	5,5	28,9	12,8	47,2	72,2	90,0	78,2	83,4

The proposed AMT system shows a good note tracking just only with the Step One stage, performing a note tracking accuracy of 61.0% with the dictionary *Steinway* and of 60.6% with the dictionary *Yamaha C2*. However, the multi-F0 accuracy is not likewise good. The principal error is attributable to false negatives, as proven by the high value of E_{fn} (57.3% and 58.4%) and the low value of R (64.7% and 64.4%). This happens because of the wide loudness differences between the louder and the softer notes, involving a greater difficulty on detecting the softer ones. Moreover, the offset detection is made worse for the same reason and the algorithm fails in detecting the notes when their energy decreases. The HDRC succeeds in solving this phenomenon. In fact, the overall accuracy significantly increases when this pre-processing is performed, as shown in both the second rows of TABLE II part (a) and part (b). In particular, for the dictionary *Steinway* the multi-F0 and the note tracking accuracies are improved by 17.5 and 7.3 percentage points, respectively, whereas for the dictionary *Yamaha C2* they are improved by 17.5 and 7.8 percentage points, respectively. This is because the HDRC helps in detecting the softer notes and to improve the offset detection. This allows to reduce E_{fn} by 27.8 points for both the dictionaries and to increase R by 12.4 and $12,6$ points. By adding the Step Two, it can be noticed as actually the threshold filtering succeeds in furtherly increasing the sparsity with respect to the Step One, where only the ℓ_2 regularization is performed. The second step allows to filter out most of the residual noises and harmonic interference and to refine the notes detection. This results in a general reduction of the substitution errors E_{sub}, an increasing of P and R and an overall improvement of both the multi-F0 and the note tracking performances.

2) Comparison with Regularized NMD: The effectiveness of the two-step processing in enhancing the sparsity and the accuracy of the transcription with respect to a regularized NMD has been investigated in this test. In particular, the performances of the proposed AMT system have been compared with those obtained by a NMD in which both the $\beta = 1$ and $\beta = 2$ cost functions and both the ℓ_1 and ℓ_2 regularization constraints are considered. The results are shown in TABLE III.

TABLE III
COMPARISON WITH SPARSE NMD

(a) Dictionary *Steinway*

	Multi-F0					**Note Tracking**			
Stages	Acc	E_{sub}	E_{fn}	E_{fp}	E_{tot}	Acc	P	R	F_m
Proposed	54,6	5,5	29,6	12,3	47,4	72,1	90,5	77,6	83,3
$\beta = 1$ and ℓ_1	54,1	5,1	31,2	11,7	48,0	71,7	90,3	77,3	83,0
$\beta = 1$ and ℓ_2	54,2	5,1	31,1	11,7	47,9	71,7	90,4	77,3	83,1
$\beta = 2$ and ℓ_1	52,0	7,5	29,5	13,2	50,2	68,2	85,4	77,0	80,7
$\beta = 2$ and ℓ_2	52,0	7,5	29,5	13,2	50,1	68,3	85,4	77,1	80,7

(b) Dictionary *Yamaha C2*

	Multi-F0					**Note Tracking**			
Stages	Acc	E_{sub}	E_{fn}	E_{fp}	E_{tot}	Acc	P	R	F_m
Proposed	55,1	5,5	28,9	12,8	47,2	72,2	90,0	78,2	83,5
$\beta = 1$ and ℓ_1	54,2	5,2	30,7	12,2	48,1	71,5	89,6	77,8	83,0
$\beta = 1$ and ℓ_2	54,2	5,2	30,7	12,3	48,1	71,6	89,6	77,9	83,0
$\beta = 2$ and ℓ_1	51,1	7,7	30,5	12,6	50,8	68,4	85,8	77,0	80,9
$\beta = 2$ and ℓ_2	51,1	7,6	30,6	12,6	50,8	68,4	85,8	77,0	80,9

It can be seen as the proposed two-step processing succeeds always in achieving a more accurate transcription with respect to all the regularized NMD configurations. In particular, it is noticeable that the main improvement with respect to the NMD with $\beta = 2$ is related to E_{sub}, with the proposed AMT system getting a lower value of about 2 points with respect to the NMD. This is because the proposed procedure succeeds in achieving a more sparse activation matrix $\mathbf{X}$ and consequently a more accurate transcription. Conversely, the NMD with $\beta = 1$ achieves slightly worse performances with respect to the proposed two-step processing. However, it requires about $370\,s$ for transcribing a one minute length musical excerpts in front of the $60\,s$ of the NMD with $\beta = 2$ and of the $85\,s$ of the two-step processing. Thus, besides the achieved improvements on the transcription accuracy, it is noticeable also the competing time efficiency of the proposed two-step processing.

3) Comparison with Open Source Software: The proposed system has been validated through a direct comparison with three open source AMT software. The use of open source software allowed a rigorous comparison between the algorithms upon the same Test Set and the same testing procedure discussed in Sect. III-A. The considered AMT systems are: the software *SONIC*, developed by Marolt and discussed in [19]; the *VAMP* plugin *Silvet Note Transcription*, developed by Benetos & Dixon and discussed in [28], [29]; the Constrained NMF algorithm (CNMF), developed by Vincent and discussed in [32]. The competing AMT systems have been tuned to the best parameters configuration, where applicable. In particular, *Silvet Note Transcription* has been run in the *intensive processing mode* that ensures the higher transcription quality; furthermore, the most suitable instrument dictionary has been selected. The CNMF has been tuned with the best parameters set widely discussed by the authors in [32], whereas SONIC does not have any tuning parameter.

The average results performed in all the thirty musical excerpts are shown in TABLE IV, in which the outcomes of the proposed method are presented separately for the two dictionaries.

TABLE IV
COMPARISON WITH OPEN SOURCE ALGORITHMS

	Multi-F0					**Note Tracking**			
Algorithm	Acc	E_{sub}	E_{fn}	E_{fp}	E_{tot}	Acc	P	R	F_m
Proposed *Steinway*	54,6	5,5	29,6	12,3	47,4	72,1	90,5	77,6	83,3
Proposed *Yamaha C2*	55,1	5,5	28,9	12,8	47,2	72,2	90,0	78,2	83,5
Marolt SONIC	52,4	6,8	26,1	19,9	52,8	70,6	86,1	78,9	81,9
Benetos-Dixon Silvet	46,5	6,7	39,7	8,1	54,4	56,5	78,2	64,8	70,7
Vincent CNMF	40,4	20,4	20,3	32,4	73,0	20,6	31,8	35,7	33,4

It can be seen as the proposed algorithm shows better results with respect to the others with both the dictionaries and in both the multi-F0 and the note tracking tasks. In particular, the low values of E_{sub} and of E_{fp} highlight as the two-step processing succeeds in filtering out most of the harmonic interference, improving the sparsity and the overall transcription accuracy. On the other hand, the high value of E_{fn}, jointly with the higher note tracking accuracy, suggests that the main weakness is the offset detection.

In order to compare the performances in front of a real piano, the results related to only the *Yamaha Disklavier* are listed in TABLE V.

TABLE V
RESULTS RELATED TO THE YAMAHA DISKLAVIER RECORDINGS

	Multi-F0					**Note Tracking**			
Algorithm	Acc	E_{sub}	E_{fn}	E_{fp}	E_{tot}	Acc	P	R	F_m
Proposed *Steinway*	46,1	7,7	38,2	8,5	54,4	63,8	85,5	71,1	77,4
Proposed *Yamaha C2*	46,8	7,6	37,2	9,1	53,9	64,5	85,3	72,1	77,9
Marolt SONIC	48,1	7,5	33,8	12,8	54,1	60,5	82,7	68,5	74,4
Benetos-Dixon Silvet	43,1	6,7	44,5	6,0	57,2	54,6	78,6	63,0	69,8
Vincent CNMF	35,9	19,6	30,4	23,0	73,1	21,3	33,0	37,2	34,7

The same previous consideration can be underlined, with both the dictionaries obtaining better results than the other open source algorithms. The only exception happens for the multi-F0 evaluation, in which the software SONIC is slightly better. Again the results in E_{fn} and the best note tracking accuracy suggests that the worse multi-F0 accuracy is related to errors in the offset detection. Evidently, it represents the principal weakness of the proposed method.

4) Violin Transcription: In the last test, the performances of the proposed system have been evaluated for violin transcription in order to briefly investigating its capability of dealing with instruments different from the piano. During this test, the tuning parameters are not changed although they have been set for piano transcription.

The violin Test Set is composed of 11 MIDIs of one minute length, downloaded at [46]. These musical excerpts include monophonic and polyphonic parts and also staccato and non-staccato playing. In this case, only one VST has been used and once again the twelve notes of the 4-th octave have been considered for the instrument learning. The results are shown in TABLE VI where also the outcomes of the multi-instrument algorithms of Benetos & Dixon and of Vincent are listed.

TABLE VI
VIOLIN TRANSCRIPTION RESULTS

	Multi-F0					**Note Tracking**			
Algorithm	Acc	E_{sub}	E_{fn}	E_{fp}	E_{tot}	Acc	P	R	F_m
Proposed	62,0	7,2	11,2	25,7	44,0	68,0	73,0	90,4	80,6
Benetos-Dixon Silvet	65,3	3,6	20,7	12,0	36,4	72,8	82,5	83,7	83,0
Vincent CNMF	40,5	22,3	12,5	41,8	76,6	15,6	22,1	35,2	26,6

Although the proposed system is slightly outperformed by *Silvet Note Transcription*, it is remarkable the high accuracy performed both in the multi-F0 and in the note tracking tasks. Indeed, the algorithm reaches a performance close to one of the best multi-instrument AMT software without performing any particular tailoring on the current instrument except for the learning procedure. These results suggest that a proper tuning of the HMM on the typical scores of the violin will produce better results.

D. Time Complexity

In TABLE VII the average times needed to perform the transcription of a musical excerpt of one minute length are listed. All the transcriptions have been performed on a machine mounting an *Intel i5 3230M*, 2.6 GHz and 8 GB of RAM.

TABLE VII
AVERAGE TIME TO TRANSCRIBE ONE MINUTE OF MUSIC

Proposed	Marolt SONIC	Benetos-Dixon Silvet	Vincent CNMF
85 s	234 s	88 s	45 s

Once again, the proposed AMT system shows competitive performances with respect to the state of art algorithms. Moreover, considering that the implementation of the proposed AMT software includes parallel programming, these results suggest that on better performing workstations, exploiting a deep multi-core architecture, music transcription could be also performed in real time.

IV. CONCLUSION

In this paper, an AMT algorithm based on the matrix factorization technique has been proposed and validated. The main contribution consists in a novel approach for enhancing the sparsity of the supervised NMD method. In order to achieve this improvement, the transcription is performed using a two-step processing. The Step One aims to perform an overall selection of the most likely played notes, solving the NMD by minimizing the β-divergence with $\beta = 2$ and considering a ℓ_2 regularization. During the Step Two, the threshold filtering followed by another NMD are repeatedly performed in order to progressively reduce the dictionary matrix and to refine the notes transcription. Jointly with the more effective $\beta = 1$ and the ℓ_2 regularization, the second step allows to improve the sparsity of the solution and to increase the transcription accuracy. Differently from other AMT systems in which the matrices $\mathbf{D}$ and $\mathbf{X}$ are estimated as a whole by means of multiplicative rules, the proposed approach solves a constrained optimization problem for each frame by means of a non-monotone modification of the projected gradient method. This solving approach allows to deal efficiently with the non-negative constraints and to perform in parallel the transcription of each frame. Furthermore, a minor contribution consists in the conceived instrument learning procedure. This procedure has been designed to be as user-oriented as possible and it requires to record a very limited number of monophonic exemplars in order to allow the building of a dictionary matrix tailored upon the actual instrument to transcribe.

A rigorous testing campaign has been performed evaluating the transcription performance upon the Test Set collected by the LabROSA laboratories and considering three different pianos. An appropriate testing procedure has been defined and the metrics officially approved by the MIREX have been considered for the performance evaluation. The results prove the effectiveness of the proposed two-step processing, showing an actual improvement in the transcription accuracy. Furthermore, the algorithm has been validated through a comparison with three open source AMT software. The results show promising performance, with the proposed system obtaining almost always better transcription accuracy, both in the multi-F0 and in the note tracking tasks. The main weakness appears to be the offset detection, as proven by the high value obtained for E_{fn} jointly with the best note tracking accuracy. This proves that the algorithm detects the majority of notes, but it is not very accurate in recognizing the offset position. The violin transcription results show that the proposed algorithm achieves a performance very close to that of *Silvet Note Transcription* although it has not been performed any tuning on the violin instrument. Besides the above observation, it is remarkable that only twelve notes have been sufficient for building a meaningful dictionary able to perform with such a great accuracy.

Some improvements should be implemented. First of all, a group sparsity approach could be included in each of the two steps with the aim of furtherly reducing the effect of harmonic interference and of the overtone errors. Then, a semi-supervised approach could help in adaptively update the dictionary built with the proposed learning procedure. Finally, other efforts could be made for developing an automatic tailoring of the HMM to the current instrument. This, jointly with the proposed instrument learning procedure, with a semi-supervised technique and with an automatic instrument recognition, could bring to further improvements in the transcription accuracy.

OPEN SOURCE REPOSITORY

The source code, the Windows executable, the MATLAB scripts used for the performance evaluation and the whole dataset, including both the ground truth MIDIs, the ground truth WAVEs and the transcribed MIDIs, are available at https://bitbucket.org/MassimilianoLuzi/notesphinder/src. The repository is released under the terms of the GNU General Public License as published by the Free Software Foundation and the authors agree to the use of this software exclusively for research purposes and for reproducing the results shown in this paper.

References

[1] K. Umapathy, S. Krishnan, and S. Jimaa, "Multigroup classification of audio signals using time-frequency parameters," *IEEE Transactions on Multimedia*, vol. 7, no. 2, pp. 308–315, April 2005.

[2] Y. Yu, R. Zimmermann, Y. Wang, and V. Oria, "Scalable content-based music retrieval using chord progression histogram and tree-structure lsh," *IEEE Transactions on Multimedia*, vol. 15, no. 8, pp. 1969–1981, Dec 2013.

[3] M. Clausen and F. Kurth, "A unified approach to content-based and fault-tolerant music recognition," *IEEE Transactions on Multimedia*, vol. 6, no. 5, pp. 717–731, Oct 2004.

[4] H.-M. Yu, W.-H. Tsai, and H.-M. Wang, "A query-by-singing system for retrieving karaoke music," *IEEE Transactions on Multimedia*, vol. 10, no. 8, pp. 1626–1637, Dec 2008.

[5] T. De Mulder, J.-P. Martens, S. Pauws, F. Vignoli, M. Lesaffre, M. Leman, B. De Baets, and H. De Meyer, "Factors affecting music retrieval in query-by-melody," *IEEE Transactions on Multimedia*, vol. 8, no. 4, pp. 728–739, Aug 2006.

[6] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri, "Automatic music transcription: challenges and future directions," *Journal of Intelligent Information Systems*, vol. 41, no. 3, pp. 407–434, 2013.

[7] J. Brown and B. Zhang, "Musical frequency tracking using the methods of conventional and narrowed autocorrelation," *Journal of Acoustical Society of America*, vol. 89, no. 5, pp. 2346–2354, 1991.

[8] J. Brown, "Musical fundamental frequency tracking using a pattern recognition method," *Journal of Acoustical Society of America*, vol. 92, no. 3, 1992.

[9] J. A. Moorer, "On the trancription of musical sound by computer," *Computer Music Journal*, vol. 1, no. 4, pp. 32–38, Nov. 1977.

[10] M. Piszczalski and B. Galler, "Automatic music transcription," *Computer Music Journal*, vol. 1, no. 4, pp. 24–31, Nov. 1977.

[11] L. Rossi, G. Girolami, and M. Leca, "Identification of polyphonic piano signals," *Acustica*, vol. 83, no. 6, pp. 1077–1084, 1997.

[12] S. Dixon, "On the computer recognition of solo piano music," in *Australasian Computer Music Conference*, Brisbane, Australia, Jul. 2000, pp. 31–37.

[13] J. Bello, L. Daudet, and M. Sandler, "Time-domain polyphonic transcription using self-generating databases," in *112st Convention of Audio Engineerig Society*, Munich, Germany, May 2002.

[14] A. Klapuri, "A perceptually motivated multiple-f0 estimation method," in *IEEE Workshop on Applications of Signal Processin to Audio and Acoustics*, New Palts, NY, USA, Oct. 2005, pp. 291–294.

[15] M. Ryynänen and A. Klapuri, "Polyphonic music transcription using note event modeling," in *IEEE Workshop on Applications of Signal Processin to Audio and Acoustics*, New Palts, NY, USA, Oct. 2005, pp. 319–322.

[16] M. Slaney and R. Lyon, "A perceptual pitch detector," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Apr. 1990, pp. 357–360.

[17] D. Ellis, "Prediction-driven computational auditory scene analysis," Ph.D. dissertation, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, Jun. 1996.

[18] A. Klapuri, "Multipitch analysis of polyphonic music and speech signals using an auditory model," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 255–266, Feb. 2008.

[19] M. Marolt, "A connectionist approach to automatic transcription of polyphonic piano music," *IEEE Transactions on Multimedia*, vol. 6, no. 3, pp. 439–449, Jun. 2004, http://lgm.fri.uni-lj.si/matic/SONIC/sonic.zip.

[20] G. E. Poliner and D. Ellis, "A discriminativemodel for polyphonic piano transcription," *EURASIP Journal on Advances in Signal Processing*, pp. 154–162, 2007.

[21] M. Davy, S. J. Godsill, and J. Idier, "Bayesian analysis of polyphonic western tonal music," *Journal of the Acoustical Society of America*, vol. 119, pp. 2498–2517, 2006.

[22] K. Yoshii and M. Goto, "A nonparametric bayesian multipitch analyzer based on infinite latent harmonic allocation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 3, pp. 717–730, March 2012.

[23] E. Benetos and T. Weyde, "An efficient temporally-constrained probabilistic model for multiple-instrument music transcription," in *16th International Society for Music Information Retrieval Conference*, Malaga, Spain, Oct. 2015, pp. 701–707.

[24] S. Abdallah and M. Plumbley, "Polyphonic music transcription by non-negative sparse coding of power spectra," Centre for Digital Music, Queen Mary, University of London, Tech. Rep., 2004.

[25] A. Dessein, A. Cont, and G. Lemaitre, "Real-time polyphonic music transcription with non-negative matrix factorization and beta-divergence," in *11th International Society for Music Information Retrieval Conference*, Utrecht, Netherlands, Aug. 2010, pp. 489–494.

[26] G. Grindlay and D. Ellis, "Transcribing multi-instrument polyphonic music with hierarchical eigeninstruments," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 6, pp. 1159–1169, 2011.

[27] K. O'Hanlon and M. D. Plumbley, "Polyphonic piano transcription using non-negative matrix factorisation with group sparsity," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2014, pp. 3112–3116.

[28] E. Benetos and S. Dixon, "A shift-invariant latent variable model for automatic music transcription," *Computer Music Journal*, vol. 36, pp. 81–94, 2012, https://code.soundsoftware.ac.uk/attachments/download/1626/silvet-win32-v1.1.0.1.zip.

[29] ——, "Multiple-instrument polyphonic music transcption using a temporally constrained shift-invariant model," *The Journal of the Acoustical Society of America*, vol. 133, no. 3, pp. 1727–1741, Mar. 2013.

[30] P. Smaragdis and J. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, Oct. 2003, pp. 177–180.

[31] C.-T. Lee, Y.-H. Yang, and H. Chen, "Multipitch estimation of piano music by exemplar-based sparse representation," *IEEE Transactions on Multimedia*, vol. 14, no. 3, pp. 608–618, June 2012.

[32] E. Vincent, N. Bertin, and R. Badeau, "Adaptive harmonic spectral decomposition for multiple pitch estimation," vol. 18, pp. 528–537, 2010, http://www.loria.fr/~evincent/multipitch_tracking.m.

[33] M. Genussov and I. Cohen, "Multiple fundamental frequency estimation based on sparse representations in a structured dictionary," *Digital Signal Processing*, vol. 23, no. 1, pp. 390–400, 2013.

[34] K. O'Hanlon, H. Nagano, N. Keriven, and M. D. Plumbley, "Non-negative group sparsity with subspace note modelling for polyphonic transcription," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 3, pp. 530–542, Mar. 2016.

[35] S. I. Adalbjörnsson, A. Jakobsson, and M. G. Christensen, "Multi-pitch estimation exploiting block sparsity," *Signal Processing*, vol. 109, pp. 236–247, 2015.

[36] LabROSA, *Automatic Piano Transcription*. Columbia University, New York: http://labrosa.ee.columbia.edu/projects/piano/, 2006.

[37] A. R. Conn, N. I. M. Gould, and P. L. Toint, *Lancelot: A Fortran Package for Large-Scale Nonlinear Optimization (Release A)*, 1st ed. Springer Publishing Company, Incorporated, 2010.

[38] B. Bartesekas, *Nonlinear Programming*, A. Scientifi, Ed., 1999.

[39] E. G. Birgin, J. . M. Martínez, and M. Raydan, "Nonmonotone spectral projected gradient methods on convex sets," *SIAM Journal on Optimization*, pp. 1196–1211, 2000.

[40] M. Antonelli and A. Rizzi, "A non-monotone optimization algorithm for IIR filter design," in *IEEE Workshop on Machine Learning for Signal Processing*, Aug. 2007, pp. 372–377.

[41] M. Antonelli, *Nonmonotone projected gradient algorithm*, http://www.marioantonelli.it/non-monotone-optimiziation-algorithms.

[42] J. Youngberg, "Constant-q signal analysis and synthesis," in *IEEE International Conference Acoustics, Speech, and Signal Processing*, vol. 3, 1978, pp. 375–378.

[43] J. C. Brown, "Calculation of a constant Q spectral transform," *The Journal of the Acoustical Society of America*, vol. 89, no. 1, pp. 425–434, 1991.

[44] J. C. Brown and M. S. Puckette, "An efficient algorithm for the calculation of a constant Q transform," *J. Acoustic Soc. America*, 1992.

[45] Music Information Retrieval Evaluation eXchange, 2016, http://www.music-ir.org/mirex/wiki/MIREX_HOME.

[46] Violin Test Set, http://www.jsbach.net/midi/midi_solo_violin.html and http://www.kunstderfuge.com/classical/p.htm.

Antonello Rizzi (S'98–M'04) received the Ph.D. in Information and Communication Engineering in 2000, from the University of Rome "La Sapienza". In September 2000 he joined the "Information and Communication" Department (INFO-COM Dpt., "La Sapienza") as an Assistant Professor. Since July 2010 he joined the "Information Engineering, Electronics and Telecommunications" Dpt., in the same University. His major research interests are in the area of Soft Computing, Pattern Recognition and Computational Intelligence, including supervised and unsupervised data driven modeling techniques, neural networks, fuzzy systems and evolutionary algorithms. His research activity concerns the design of automatic modeling systems, focusing on classification, clustering, function approximation and prediction problems. Currently, he is working on different research topics and projects, such as smart grids and micro-grids modeling and control, intelligent systems for sustainable mobility, battery management systems, Granular Computing, Data Mining and Knowledge Discovery, Content Based Retrieval Systems, classification and clustering systems for structured patterns, graph and sequence matching, agent-based clustering. Since 2008, he serves as the scientific coordinator and technical director of the R&D activities in the Intelligent Systems Laboratory within the Research and Technology Transfer Center for Sustainable Mobility of Lazio Region. Dr. Rizzi (co-)authored more than 130 international journal/conference articles and book chapters. He is a member of IEEE.

Mario Antonelli was born in Tivoli (RM) on October 15, 1977. He received the Dr. Eng. degree in Telecommunication Engineering, in May 2004, from the University of Rome "La Sapienza" and the Ph.D in Information and Communication Engineering, in 2008, from the same University. An important part of his research activity, during Ph.D studies, was dedicated to the definition and implementation of Evolutionary Optimization Algorithms for Classification, Automatic Recognition Algorithms, Computational Intelligence and Signal Processing. After the Ph.D, he worked first as System Developer in the field of Defense Systems and Security Systems, then as System Analyst on the same fields. His works has centered around Intelligent Signal Processing, Tactical Algorithms, Image Porcessing, Automatic Classification and Recognition. Currently, he is working as a Senior System Analyst in IDS - Ingneria dei Sietemi on different projects and research activities concerning Computer Aided Engineering, Operations Research, Evolutionary Algorithms, Antenna Design, Air Traffic Management.

Massimiliano Luzi was born in Ronciglione (VT), Italy, on August 20, 1987. He received the Bachelor's Degree in Electronic Engineering in September 2010 and the Master's Degree in Electronic Engineering with specialization in Signal Processing in November 2014, both from the University "La Sapienza" of Rome. Currently he is a PhD student in "Information and Communication Technology" at the Department of Information Engineering, Electronics and Telecommunications (DIET) of the same University. His major research activity regards the area of Soft Computing, Pattern Recognition and Computational Intelligence. In particular, he is studying the application of Soft Computing approaches, such as neural networks and evolutionary algorithms, in several fields of interest, such as multimedia and content based retrieval, modeling and optimization for energy efficiency, data mining and knowledge discovery. Since 2016 he is part of the Intelligent Systems Laboratory within the Research and Technology Transfer Center for Sustainable Mobility of Lazio Region, Italy. He is a student member of IEEE.