Identifying Missing and Extra Notes in Piano Recordings Using Score-Informed Dictionary Learning

Siying Wang, *Student Member, IEEE,* Sebastian Ewert, *Member, IEEE,* and Simon Dixon

***Abstract*—The goal of automatic music transcription (AMT) is to obtain a high-level symbolic representation of the notes played in a given audio recording. Despite being researched for several decades, current methods are still inadequate for many applications. To boost the accuracy in a music tutoring scenario, we exploit that the score to be played is specified and we only need to detect the differences to the actual performance. In contrast to previous work which uses score information for post-processing, we employ the score to construct a transcription method that is tailored to the given audio recording. By adapting a score-informed dictionary learning technique as used for source separation, we learn for each score pitch a spectral pattern describing the energy distribution of associated notes in the recording. In this paper, we identify several systematic weaknesses in our previous approach and introduce three extensions to improve its performance. Firstly, we extend our dictionary of spectral templates to a dictionary of variable-length spectro-temporal patterns. Secondly, we integrate the score information using soft rather than hard constraints, to better take into account that differences from the score indeed occur. Thirdly, we introduce new regularizers to guide the learning process. Our experiments show that these extensions particularly improve the accuracy for identifying extra notes, while the accuracy for correct and missing notes remains at a similar level. The influence of each extension is demonstrated with further experiments.**

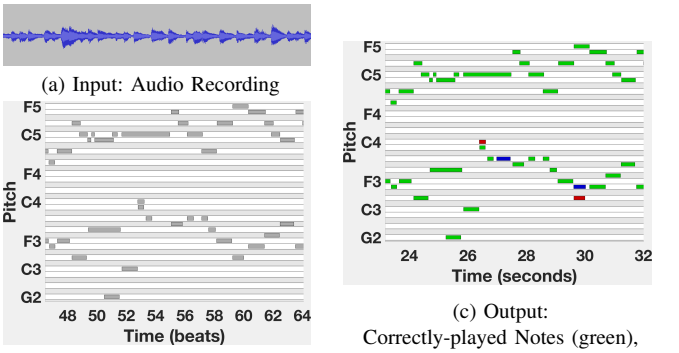
***Index Terms*—Music Transcription, Score-Informed Dictionary Learning, Non-Negative Matrix Factorization, Music Tutoring.**

I. Introduction

AUTOMATIC music transcription (AMT) has been an active research area for several decades and is often considered to be a key technology in music signal processing [1]. Its applications range from content-based music retrieval and interactive music interfaces [1] to musicological analysis [2], music education [3] and note-based audio processing [4]. While for certain applications the accuracy of state-of-the-art methods is sufficiently high, current methods still do not reach the sophistication of a transcription made by human experts. In addition, current methods seem to have reached a plateau in performance and it has become increasingly difficult to make significant improvements [1]. Therefore, many interesting applications involving AMT technologies remain infeasible.

The authors are with the Centre for Digital Music, School of Electronic Engineering and Computer Science, Queen Mary University of London, London, U.K. (e-mail: {siying.wang,s.ewert,s.e.dixon}@qmul.ac.uk.)

Manuscript received March 2, 2017; revised March 2, 2017. This work was supported in part by the China Scholarship Council and in part by the Engineering and Physical Sciences Research Council under Grants EP/J010375/1 and EP/L019981/1



(a) Input: Audio Recording

(b) Input: Score Representation

(c) Output: Correctly-played Notes (green), Missing Notes (red), Extra Notes (blue)

Fig. 1. The score-informed transcription process: Given (a) an audio recording and (b) the corresponding score, identify (c) the correctly played score notes, the missing notes and the extra notes.

One way to boost the accuracy of an AMT system is to provide additional information, e.g. originating from annotations interactively made by the user during the transcription process [5], or single note recordings giving more details about the instrument in use and the recording conditions [6]–[8]. In this paper, we investigate a particular type of prior knowledge available in a specific application scenario: a musical score. In particular, we explore the scenario of a music tutoring application, in which the system evaluates a student's performance with regards to how faithfully the score was reproduced, in order to provide feedback on when and how the student deviates. More precisely, given a digital encoding of the score of a piece of music (Fig. 1(b)) and an audio recording of a student playing that piece (Fig. 1(a)), the goal is to identify which score notes have been played correctly, which have not been played (*missing notes*) and which notes have been played that are not found in the score (*extra notes*) – see Fig. 1c.

Theoretically, standard AMT methods could be employed in this context by using them to generate a transcription from the audio and comparing the result with the given score. In practice, however, the error rates of existing methods are high and thus such a comparison would often be meaningless, with many detected extra and missing notes actually being transcription errors. To the best of the authors' knowledge, only two methods have aimed at improving upon this off-the-shelf approach [9], [10]. Benetos et al. [9] first align the score to the audio and then, after synthesizing the score

using a wavetable method, transcribe both the real and the synthesized audio using an AMT method. To lower the number of falsely detected notes for the real recording, the method discards a detected note if the same note is also detected in the synthesized recording while no corresponding note can be found in the score. Here, the underlying assumption is that in such a situation, the combination of harmonic intervals might lead to uncertainty in the spectrum, which could cause an error in their proposed method. To improve the results further, the method requires the availability of single note recordings for the instrument to be transcribed (under the same recording conditions) – a requirement not unrealistic to fulfill in this application scenario but leading to additional demands for the user. Under these additional constraints, the method lowered the number of transcription errors considerably compared to standard AMT methods.

A conceptual weakness in [9] is that the main purpose of the score information is to post-process the transcription results obtained via a standard AMT method. In contrast, the main idea in [10] is to exploit the available score information to adapt the transcription method itself to a given recording. To this end, the score is used to modify two central components of an AMT system: the set of spectral patterns used to identify note objects in a time-frequency representation, and the subsequent note detection process. More precisely, similar to strategies used in score-informed source separation [11], the method constrains the dictionary learning process in non-negative matrix factorization (NMF) using the score information. This way, the method yields for each pitch in the score template vectors describing the spectral energy distribution in the recording associated with notes of that pitch. After extrapolating the learned dictionary to pitches not in the score, the adapted dictionary is used to compute unconstrained activations for all pitches over time. Assuming that the number of playing mistakes is relatively low compared to the total number of notes, the score information is used to adapt the note detection process such that the match between detected notes and score notes is maximized in a certain way. By comparing the resulting final transcription to the score, the notes can be classified as either correct, missing or extra. Integrating the score information into the method itself, the method considerably improved upon the state of the art, even without the requirement to provide single note recordings as in [9].

The main contributions of this paper are the identification of several systematic weaknesses in the signal model used in [10] and designing corresponding improvements. First, the signal representation used in [10] is based on NMF, where spectral and temporal properties are modeled independently [12]. As demonstrated in [13], [14] this decoupling of information is generally not appropriate for non-stationary sounds – for example, one typically cannot express in standard NMF that a certain spectral template for the sustain part of a note is expected a certain time after the attack. Therefore, incorporating ideas presented in [7], we extend the concept of a dictionary of spectral templates used in [10] to a dictionary of variable-length spectro-temporal patterns to better account for the highly non-stationary behavior of piano sounds. Similar to [7], we guide the corresponding parameter estimation process using specific regularizers instead of explicit Markov-constraints [13], [14], which circumvents various issues regarding the computational efficiency and numerical properties associated with the latter, as detailed in [7].

A second weakness in [10] is that the score-information is incorporated into the NMF dictionary learning process using hard constraints – if the aligned score specifies that a certain pitch is inactive at a given time, the learning process cannot overrule this information. As a consequence, the energy associated with extra notes must be modeled with templates associated with other pitches, which in certain situations can introduce errors into the learning process, as we will see below. As a counter measure we incorporate the score information using soft constraints or regularizers into the learning process, effectively implementing rather a bias than a hard constraint. This way, we can better account for the case of a student locally deviating from the score. As a third extension, we introduce new regularizers in order to guide the learning process more explicitly, taking the typical spectro-temporal progression of piano sounds better into account [15]. Finally, we conduct systematic experiments to illustrate the influence of individual parameters and provide additional insights into the behavior of the proposed methods.

The paper is organized as follows. We discuss related work in Section II and present the baseline method in section III. Technical details of the proposed method are described in Section IV, followed by the experimental results in Section V. Conclusions and discussions of future work are given in Section VI.

II. Related work

A. Automatic Music Transcription (AMT)

A large group of methods directly related to ours consists of the various approaches to the general AMT problem. In the following, we discuss a few central contributions and refer for a more comprehensive overview to [1], [6], [16]. As a fundamental problem in music processing, a wide range of approaches has been proposed over the years. For example, [17] proposed a joint pitch estimation method which progressively combines F0 candidates into pitch or note objects. Further, various probabilistic models have been employed for AMT, such as a method using maximum a posteriori (MAP) estimation [18] or methods based on non-parametric Bayesian models [19]. Modeled as a classification or regression task, transcription has also been addressed by several discriminatively trained methods, using support vector machines [20], convolutional neural networks [21], [22] , deep belief networks [23] or recurrent neural networks [24], [25].

Among the various approaches, most state-of-the-art AMT methods are based on spectrogram factorization techniques, such as Non-negative Matrix Factorization (NMF) or its probabilistic formulation, Probabilistic Latent Component Analysis (PLCA) – see [26] for an overview. One reason for NMF's popularity is the fact that, from a machine learning perspective, it belongs to the group of generative models, which often employ interpretable parameters and thus enable a direct way to incorporate prior knowledge and adapt the method to specific

acoustic conditions [7], [27]. In the AMT context, NMF was introduced in [12] to decompose an input spectrogram into a product of a matrix containing spectral template vectors and a matrix encoding the activity of each template over time. Variants add constraints to spectral templates so as to enforce a harmonic structure [28], [29], or constraints for the activation matrix to enhance temporal continuity and sparsity properties [30]. Further variants such as non-negative matrix deconvolution (NMD) [31] employ, instead of individual spectral template vectors as used in NMF, entire spectro-temporal blocks as templates, each modeling a part of an entire segment in a time-frequency representation. Since these blocks have a fixed size, NMD has mainly been used for drum sound separation and transcription [32]. Shift-invariant PLCA enhances NMF's capability to represent changes in fundamental frequency by effectively coupling the parameter estimation for those templates associated with a specific musical pitch [33]. Finally, Markov constraints can be used to express that non-stationary sounds can often be modeled as a sequence of specific spectral templates, where the order is modeled using a graphical model [13], [14], [34]; as discussed in [7], this approach is particularly useful for modelling a few concurrent speakers but leads to various numerical issues when applied to highly polyphonic sounds such as piano recordings.

B. Score-informed Music Information Retrieval

A score is a natural source of information in music information retrieval (MIR) and has been used for a variety of tasks. For example, the methods proposed in [35], [36] employ the score to extract parameters capturing the dynamics and other expressive parameters from an audio recording of a musical performance. In [37] the authors align a given score to a corresponding audio recording and use score information to refine the note onset position in a post-processing step. With a focus on music learning and tutoring, Dannenberg et al. [38] use score following to match a student's performance (given as a MIDI file) with the score and analyze the performance based on the matching. Wang et al. [39] introduce a system for detecting pitch activity in violin performances, such that the result can be compared with the corresponding score in order to give feedback on the student's playing technique. The systems proposed in [9], [10] also aim at providing feedback on pitch accuracy but focus on a piano tuition scenario, which requires accounting for the highly non-stationary nature of the piano sound production process and the high level of polyphony in such recordings.

In our proposed method, a main use of the musical score is to guide a dictionary learning process. This concept has first been explored for source separation; see [11] for an overview. Using the pitch and timing information contained in the score, sound events can be better identified and located in a corresponding audio recording. Score-informed source separation techniques have been applied to note-based audio editing [4], remixing and upmixing of stereo music [40], instrument-wise equalization [41] and singing voice separation [42].

III. Algorithmic Framework and Baseline Method

As discussed above, in contrast to the score-informed transcription approach presented in [9] where score information is mainly used to post-process note detection results, our method employs the score information to adapt and change the transcription method itself. The main idea is to obtain a system highly tuned to transcribe exactly the piece at hand under the specific acoustic conditions in the given recording. In this section, we summarize the individual steps used in our previous approach [10]. This serves both as a baseline in our experiments, and as the algorithmic framework which we analyze and extend significantly in Section IV. Hence, we adapt the notation and model from [10] to prepare for the extensions to be introduced.

Step 1: Score-Audio Alignment

Our first step is to align a score (given as a MIDI file) to an audio recording of a student playing that score. After the alignment, we know for each score note, the expected time position including onset and offset in the corresponding performance. This way, we can reduce the influence of tempo variations on the proposed system. The alignment method we use here was proposed in [43] and combines chroma with onset indicator features to increase the temporal accuracy of the resulting alignments. In [10] we found this method to be accurate enough for our purposes despite the playing errors that are unexpected by the method (for each score note onset, the alignment deviation is 23ms on average). In particular, the next steps in our method do not rely on an exact alignment but include generous temporal tolerances to account for possible local alignment inaccuracies. However, other alignment techniques should be employed to address cases of structural differences [44]–[46] or asynchronies between voices [47]. For the purposes of this paper, however, we focus on play-through performances with missing and extra notes and leave such more advanced scenarios to future work.

Step 2: Score-Informed Adaptive Dictionary Learning

The alignment result provides, for each note in the score, information on the expected time position in the audio recording. In a next step, we use this as prior information to learn how each note manifests in a time-frequency representation of the audio recording, suitably adapting techniques used in score-informed source separation. This way, we can make fewer general assumptions and rather learn the details from data. In particular, our idea is to impose score-informed constraints to a model based on non-negative matrix factorization (NMF). Using the score constraints enables us to obtain a structured, pitch-based dictionary that is adapted to the specific audio recording and can be used to model the input with high detail, see also [11]. In the following, we assume general familiarity with NMF and refer to [48] for further details.

Let K be the number of different pitches we want to consider in our model, and L the number of individual spectral template vectors we associate with a single pitch. We define $P \in \mathbb{R}_{\geq 0}^{M \times (K \cdot L)}$ as the *spectral dictionary matrix*, where M

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/TASLP.2017.2724203, IEEE/ACM Transactions on Audio, Speech, and Language Processing

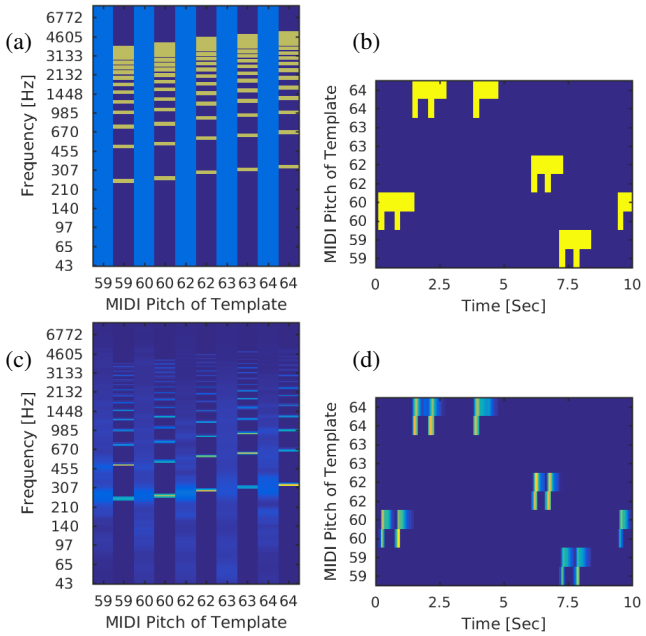


Fig. 2. Score-Informed Dictionary Learning: Using multiplicative updates in non-negative matrix factorization, constraints can easily be enforced by setting individual entries to zero (dark blue): (a) Templates and (b) activations after the initialization; (c) Templates and (d) activations after the optimization process.

is the number of frequency bins. Each column in P defines a *(spectral) template* vector. Accordingly, let $A \in \mathbb{R}^{(K \cdot L) \times N}$ be the activation matrix, where N is the number of time frames in the given audio recording. In the following, we will also use a tensor-like notation to access individual elements in P and A in the sense that $P_{m,k,\ell} := P_{m,(k-1)\cdot L+\ell}$. The magnitude spectrogram $V \in \mathbb{R}^{M \times N}$ of a given audio recording is modeled as the product of P and A. Our goal is to obtain P and A minimizing a distance between V and PA. More precisely, we derive P and A by minimizing a cost function $c(P, A)$, which is a weighted sum of a reconstruction error term $d(V||P, A)$, and regularizer terms encouraging a certain structure in the activations (i.e. $c_i(A)$) and templates (i.e. $\widetilde{c}_j(P)$),

$$c(P, A) = d(V||P, A) + \sum_i \alpha_i c_i(A) + \sum_j \beta_j \widetilde{c}_j(P),$$

where α_i and β_j are the weights for the corresponding regularizer terms.

In our previous work [10], we did not make heavy use of regularizers but incorporated the score information as hard constraints into NMF. More precisely, we allocated two spectral templates to each pitch in the score, one for the attack and one for the sustain part, i.e. $L = 2$ and K equal to the number of unique pitches in the score. To implement constraints in NMF, we exploited the fact that, due to the use of multiplicative update rules, entries in P or A set to zero will remain zero throughout the NMF learning process. This way, we can enforce a harmonic structure in templates associated with the sustain part of a specific pitch: given the pitch, we roughly know the position of the fundamental and the harmonics and can set template entries between these positions to zero, as here no or little energy is expected [11], [49]. Leaving a small non-zero neighborhood around the expected partial positions enables learning of the exact positions of each partial. The attack templates are initialized with a uniform energy distribution to account for their broadband properties. Fig. 2(a) shows an example of such template initializations.

The activations are constrained in a similar way using the score information. If a pitch is expected to be inactive in a time segment according to the aligned score, the corresponding activation entries are set to zero, while the remaining entries are initialized with random positive values. To account for alignment inaccuracies, we use relatively generous tolerances of ± 0.5s to define the temporal boundaries for active pitches. To account for a lack of constraints on the attack templates, the corresponding activations are only allowed in a close vicinity around expected onset positions, see Fig. 2(b) for an example.

After these initializations, we learn the unconstrained areas of the template matrix P and activation matrix A with the commonly used Lee-Seung update rules [48]. The cost function $c(P, A)$ to minimize only contains a reconstruction error term in the form of a generalized Kullback-Leibler divergence:

$$d(V||P, A) = \sum_{m,n} V_{m,n} \log \left(\frac{V_{m,n}}{(PA)_{m,n}} \right) - V_{m,n} + (PA)_{m,n}$$

In our experiments, however, we observed that using only d, the attack templates sometimes capture too much of the energy associated with the sustain phase, which would interfere with the later note detection process. To discourage peaks in the attack templates, which typically correspond to partials of the sustain part, we encourage smoothness in amplitude along the frequency dimension using a spectral continuity regularizer:

$$\widetilde{c}_1(P) = \sum_k \sum_m \sum_{\ell \in \mathcal{A}} (P_{m,k,\ell} - P_{m-1,k,\ell})^2$$

where $\mathcal{A} \subset \{1, \ldots, L\}$ denotes the index set of the attack templates for a pitch. It corresponds to a specific form of total variation in frequency direction and by minimizing it, we encourage energy in attack templates to be distributed smoothly across the entire frequency range.

Figs. 2(c) and (d) show P and A after convergence. Compared with Fig. 2(a) and (b), the unconstrained areas have been refined to reflect the acoustical properties of the recording. Further, the attack templates show broadband characteristics thanks to the spectral continuity regularizer, while still capturing the non-uniform, pitch dependent energy distribution typical for piano attacks.

Step 3: Dictionary Extrapolation and Residual Modelling

The dictionary learned in the previous step contains only templates for pitches used in the score. In particular, pitches outside this set cannot properly be represented, which potentially includes extra notes played by the student. In this step, we therefore extrapolate the learned dictionary to the full piano range, in order to model pitches not used in the score. By employing a time-frequency representation V using a logarithmic frequency scale, we can implement this step by a simple shift operation: for a pitch not in the score, we find the closest pitch used in the score and shift the associated

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/TASLP.2017.2724203, IEEE/ACM Transactions on Audio, Speech, and Language Processing

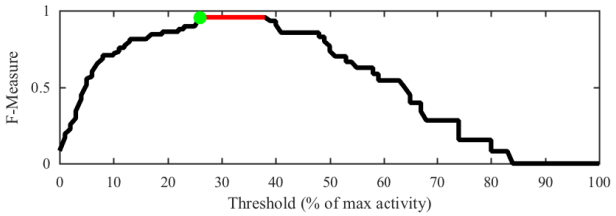


Fig. 3. Adaptive and pitch-dependent thresholding: For each pitch we choose the smallest threshold maximizing the F-measure we obtain by comparing the detected onsets against the aligned nominal score. The red entries show threshold candidates having maximal F-measure for a certain pitch and the green dot is the threshold we choose for this pitch.

templates by the number of frequency bins corresponding to the difference between the two pitches. If there are two closest pitches, we arbitrarily chose the one with the lower pitch. The higher pitch could have been used as well, however, we do not expect any differences in performance related to this choice. This complete dictionary is then fixed to compute a new and unconstrained activation matrix A for all pitches. For the initialization of A, we add rows for the newly extrapolated pitches and remove the zero constraints by adding a small value to all entries.

Step 4: Onset Detection Using Score-Informed Adaptive Thresholding

The result of the previous step is an activation matrix for a dictionary highly tuned to model the given input recording. In this step, we use the score again to adapt the decision process responsible for analyzing the activation matrix and detecting onsets. A first idea would be to detect peaks in the activations associated with attack templates of individual pitches. However, while the learned attack templates are indeed pitch-dependent (compare Fig. 2(c)), their energy distribution is relatively flat and often leads to confusion about which templates should be active in a given frame. Therefore, the method in [10] analyses only the activity for sustain templates. To this end, we define $\hat{A} \in R_{\geq 0}^{K \times N}$ via $\hat{A}_{k,n} := A_{k,2,n}$, i.e. a version of A with the activities for attack templates removed.

Next, instead of using a global threshold for all pitches as commonly done in standard AMT methods, we exploit the score information again to choose pitch-dependent thresholds to distinguish real onsets from spurious activity. In particular, as the loudness perception in the human auditory system is frequency dependent and highly complex for non-sinusoidal sounds, a pianist is likely to play each key with a different intensity resulting in different energy levels. Therefore the transcription accuracy benefits directly from choosing individualized detection thresholds. To find a suitable threshold for each k that is associated with a pitch used in the score, we generate multiple candidates which are uniformly distributed between 0 and $\max_n \hat{A}_{k,n}$. We use each candidate as a threshold in peak picking to detect onsets and calculate an F-measure comparing the detected onsets with the expected onset positions taken from the aligned score (using a temporal tolerance[1] of $T_1 = \pm 0.5$s). Among all candidates, we choose the lowest threshold that maximizes this F-measure, as illustrated in Fig. 3. Further, to improve the robustness for pitches with few notes in the score, we calculate this F-measure jointly for several neighboring pitches. For pitches not in the score, we interpolate a threshold from those for the two closest surrounding score pitches. This way, we choose a threshold that produces the best match between the detected onsets and the given score. With these thresholds, we obtain a final transcription result for the given recording. We refer to [10] for further details on this procedure.

Step 5: Score-Informed Onset Classification

As a last step, we classify the resulting onsets into *correctly played notes*, *missing notes* and *extra played notes*. To do so, we compare the transcription result with the aligned score to check for each detected onset whether there is a correspondence with the aligned score (again using a tolerance of $T_1 = \pm 0.5$s as in the last step). More precisely, if there is a correspondence between a detected note onset and a note in the score, then the detected note is classified as a correct note, otherwise as an extra note. On the other hand, if there is no correspondence between a score note and any detected onset, then the score note is classified as unplayed. In this case, the onset for the missing note is set using the alignment result, i.e. the onset corresponds to the expected position in the performance. Note that we check the correspondence by a simple local search, which may lead to mistakes in the classification for cases such as repeated notes or arpeggiated chords. A more sophisticated symbolic alignment method (such as [46]) may help to avoid such mistakes. Fig. 1(c) illustrates a classification result using a different color for each class.

IV. Analysis and Extensions

By using the score to adapt the transcription method itself to a given recording, the approach summarized in Section III considerably improves upon the state of the art [9], as demonstrated in [10]. In this section, however, we will identify several conceptual weaknesses in [10] and design corresponding countermeasures to improve the performance of our system.

A. Example of Failure

Our previous work [10] achieved a relatively high level of accuracy, and so we focus on cases where the previous method failed. The first case is an excerpt from the piece in our dataset, which led to the lowest accuracy in [10]. As illustrated in Fig. 4(a) and (b), the system is confused by a systematic error in the student's performance. More precisely, misreading the key signature in the score (left of Fig. 4(a)), the student replaces all F#3 notes with F3 notes in the performance, as illustrated on the right of Fig. 4(a). With no real F#3 in the audio, the dictionary learning process fails to learn correct templates for

[1] T_1 is mainly used to account for alignment inaccuracies and it could be adjusted for different scenarios. For example, we would increase T_1 if the student cannot yet follow the rhythm faithfully which leads to asynchronies between concurrent notes for non-musical reasons.

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/TASLP.2017.2724203, IEEE/ACM Transactions on Audio, Speech, and Language Processing

the F#3. However, since the dictionary interpolation step is not aware of this situation, the inaccurate F#3 templates are shifted to represent the F3 templates as well. The resulting P after dictionary interpolation is shown on the left of Fig. 4(b) – the template errors are visible as off-center peaks in the partials of F3 and F#3, circled in yellow. The right of Fig. 4(b) shows the activations A obtained using this dictionary. We can clearly see spurious activations for F#3 and missing activations for F3. Further, since the energy associated with the actually played F3 notes is not modeled well using these templates, there are even additional incorrect activations for the E3, which captures some of that residual energy. This systematic error is a worst-case scenario for our dictionary-learning method, as there is no correct data from which the omitted pitch can be learned. Similar situations also arise if a pitch is used only once (or a few times) on the score and the student makes a mistake playing this note (e.g. forgetting to play a very high or low pitched note). To account for these and related problems, we propose several extensions in the next subsections.

B. From Spectral Templates to Time-Frequency Patterns

The signal model used in Section III is NMF-based. A unifying aspect of most NMF and sparse coding methods is that time and frequency information are strictly separated. In typical NMF-based methods, we can neither model that a specific template for the sustain part follows an attack template after a certain amount of time, nor that the energy in a note decays in a specific way [15]. Adopting an idea from [7], we now extend our dictionary based on individual templates to a dictionary of time-frequency patterns. In particular, instead of using $L = 2$ templates, we now set L to the average length of all notes in the aligned score, in frames. In particular, L is the same for every pitch in the dictionary. For example, in Figs. 4(c)-(f), the pattern length is $L = 29$. With such a drastic increase in the number of parameters, it is a priori not clear whether the learning process will still function correctly. Therefore, to assist the score in providing a strong guidance for the learning process, we will introduce additional regularizers to improve the stability.

Similarly to [10] we apply different constraints to the attack and the sustain templates and employ the templates corresponding to first time frame (23ms) to model the attack. Fig. 4(c) shows the zero-based constraints using red frames in P and in A (the blue frames are only informational and indicate where the extra F3 notes are active). Note that the temporal constraints in A now have a diagonal structure to account for our intended spectro-temporal interpretation of each pattern. As our cost function we use the same $c(P, A)$ as in Section III, i.e. a reconstruction error term $d(V||P, A)$ and a spectral continuity regularizer $\widetilde{c}_1(P)$ on the attack templates. As can be seen from Fig. 4(c), the template matrix after the dictionary learning step captures more details compared to our previous method (b). However, due to the hard constraints, there is no activation for the actually played F3 notes. The F#3 notes that are on the score get activated with the result that the F#3 templates learn to represent the pitch F3, as shown by the energy at the bottom of the first partial of F#3 in Fig. 4(c). So,

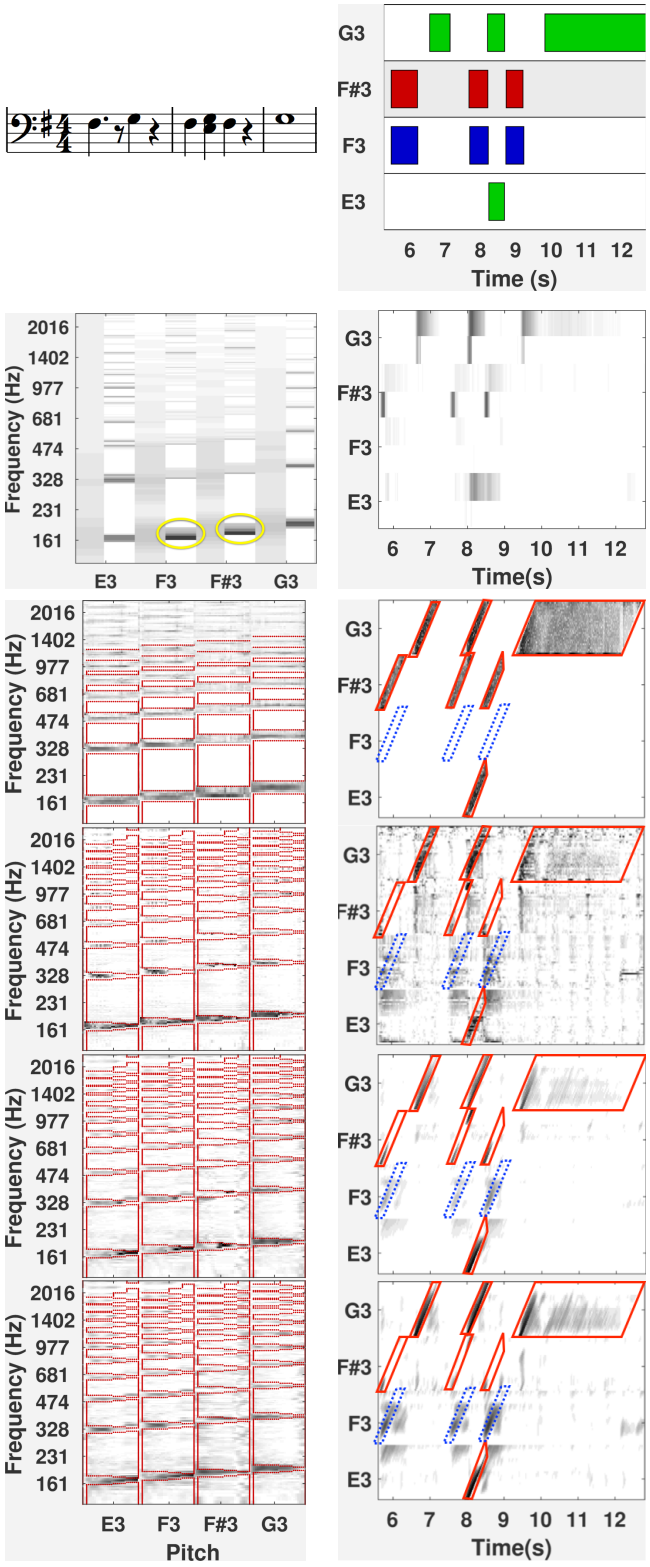


Fig. 4. A problematic case where the method introduced in [10] failed while our proposed method works correctly. (a) The input score (left) and expected output (right; green - correct notes; red - missing notes; blue: extra notes). (b) The templates (left) and activation (right) matrices obtained using [10] (template errors are circled in yellow). For each pitch, there are two columns in the template matrix and two rows in the activation matrix. (c)-(d): Extensions used in our proposed method (blue doted frames on activations: expected extra notes). (c) Extended spectro-temporal pattern dictionary using hard constraints (red frames). (d) Softening the constraints. (e) Encouraging sparsity and diagonal continuity structure in the activation matrix. (f) Encouraging energy decay in learned note patterns.

while this new signal model has potential to represent more detail, it does not resolve the above problem.

C. From Hard to Soft Constraint Regions

A main reason why the dictionary learning in Section IV-A fails is that the templates needed to represent the F3 cannot be activated due to the use of hard constraints (also discussed in a source separation context in [4]). Therefore, we now change the hard constraints used in Section IV-A into arbitrarily strong regularizers, which encourage zeros but allow for exceptions if necessary. More precisely, instead of using hard zero-based constraints, we now apply terms encouraging sparsity (similar to [30]) to the score-specified constraint regions:

$$\widetilde{c}_2(P) := \sum_{m,k,\ell} P_{m,k,\ell} \cdot (1 - M^P_{m,k,\ell})$$

$$c_1(A) := \sum_{k,\ell,n} A_{k,\ell,n} \cdot (1 - M^A_{k,\ell,n})$$

where $M^P \in \{0,1\}^{M \times (K \cdot L)}$ and $M^A \in \{0,1\}^{(K \cdot L) \times N}$ denote with ones the unconstrained areas of P and A, respectively, shown by the red frames in Fig. 4(d). Further, note that we additionally tapered the constraint region on P for each partial, which allows us to encourage harmonics to transition from being a little more broadband at the beginning to completely harmonic at the end of the note. $\widetilde{c}_2$ and c_1 are essentially (potentially strong) ℓ_1 regularizers that we selectively apply to the zero-values regions encoded in M^P and M^A, respectively.

Using these soft constraints also allows us to merge the dictionary learning and extrapolation procedures (steps 2 and 3): we simply learn time-frequency patterns for all pitches during the dictionary learning step. Since the pitches not found in the score typically also will not occur in the performance, we need to apply additional constraints to obtain correct results. To this end, we constrain each time-frequency pattern for a pitch not used in the score to be a shifted version of a pattern for a pitch found in the score. In our case, we couple the non-score pattern with the pattern of the closest score-pitch. Technically, this is related to shift-invariant dictionary learning [50] and more precisely to transformation-invariant NMF [51].

As shown on the left of Fig. 4(d), due to these changes, templates for F#3 no longer contain energy associated with the F3 (in P) and are activated less overall (in A). However, the whole activation matrix is not very structured and contains a lot of noise, rendering onset detection quite difficult. Therefore, we next introduce more regularizers to encourage further structure in P and A.

D. Encouraging Temporal Continuity in A

The idea behind our first regularizer in this subsection stems from the following observation: if the ℓ-th template for a pitch is active in frame n, the $(\ell{+}1)$-th template should be active in frame $(n{+}1)$ (given that the note is still active in frame $(n{+}1)$). Therefore, similar to [7], the following regularizer term enhances diagonal structures and discourages vertical and horizontal structures as well as unnecessary fluctuations between neighboring entries:

$$c_2(A) = \sum_{k,\ell,n} (A_{k,\ell+1,n+1} - A_{k,\ell,n})^2$$

It can be seen as an anisotropic variant of the total variation regularizer used in image processing [52] and is related to temporal smoothness terms as used in [30].

Further, we add an additional ℓ^1 regularizer to encourage sparseness across all entries in A, in order to have fewer but stronger diagonals resulting from c_2 (note that c_1 is related to c_3 but confined to our constraint regions).

$$c_3(A) = \sum_{k,\ell,n} A_{k,\ell,n}$$

As shown in Fig. 4 (e), after learning, the activation matrix becomes cleaner compared to (d) and we can see the diagonal structure of three played notes appearing for F3. However, there is still energy at the F#3 pitch, which was not played. As another problem, we find in the corresponding templates (shown on the left side of (e)) that their energy does not decay with time. For example, in the first partial of E3, F3 and F#3, the energy is higher in the last few templates than in the earlier ones, which does not reflect the temporal energy progression in piano tones.

E. Encouraging Energy Decay in the Template Matrix

With the next regularizer, we impose a decay structure onto the spectral templates associated with the sustain phase:

$$\widetilde{c}_3(P) = \sum_k \sum_m \sum_{\ell \in \mathcal{B}} f(P_{m,k,\ell} - P_{m,k,\ell-1}),$$

where $\mathcal{B} \subset \{1, \ldots, L\}$ denotes the index set of the sustain templates of the time-frequency pattern of one pitch. $f : \mathbb{R} \to \mathbb{R}$ is a function encouraging a smooth decrease in energy while penalising sudden energy increases in time direction. We define

$$f(x) := (\gamma x - 1) e^{(\gamma x - 1)}$$

with $\gamma > 0$ being a non-linear parameter, see also Fig. 5. Using the differentiable $\widetilde{c}_3$, a decrease in energy in time direction is not penalized, while an increase is strongly discouraged. As shown in Fig. 4, after learning, we can observe energy decays in time direction for individual patterns in P. With these more accurate patterns, the three played F3 notes are finally active in the activation matrix, while the F#3 notes (not played) are correctly no longer activated.

It should be noted that this regularizer is not intended to model all details of the decay process found in piano sounds. In particular, different partials decay at different rates and thus various decay patterns are possible. Further, the coupling between strings adds another layer of complexity to the decay pattern of a piano note, resulting in beating and other fluctuations in energy that overlay the overall energy decay [53]. Instead of modeling these details, the main purpose of this regularizer is to assist in the identification of the main effect, i.e. a strong exponential energy decay.

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/TASLP.2017.2724203, IEEE/ACM Transactions on Audio, Speech, and Language Processing

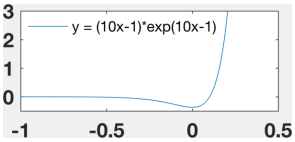


Fig. 5. Plot of function $f(x) = (\gamma x - 1)e^{(\gamma x - 1)}$ for $\gamma = 10$.

F. Parameter Estimation

The cost function we need to minimize in order to obtain P and A is,

$$c(P,A) = d(V||P,A) + \alpha_1 c_1(A) + \alpha_2 c_2(A) + \alpha_3 c_3(A) + \beta_1 \widetilde{c}_1(P) + \beta_2 \widetilde{c}_2(P) + \beta_3 \widetilde{c}_3(P).$$

We use a similar multiplicative updating strategy to [30] to alternately update P and A until convergence. The gradients of terms in the cost function with respect to A are given by

$$\begin{aligned} \nabla^A d(V||P,A) &= P^\mathsf{T} 1 - P^\mathsf{T} \frac{V}{PA} \\ \nabla^A c_1(A) &= 1 - M_A \\ [\nabla^A c_2(A)]_{k,\ell,n} &= 4A_{k,\ell,n} - 2A_{k,\ell-1,n-1} - 2A_{k,\ell+1,n+1} \\ \nabla^A c_3(A) &= 1 \end{aligned}$$

We write the gradient of the cost function as the difference between element-wise positive terms and negative terms:

$$\begin{aligned} [\nabla^A c(P,A)]_{k,\ell,n} &= [\nabla^A c^+(P,A)]_{k,\ell,n} - [\nabla^A c^-(P,A)]_{k,\ell,n} \\ [\nabla^A c^+(P,A)]_{k,\ell,n} &= \sum_m P_{m,k,\ell} + \alpha_1 + 4\alpha_2 A_{k,\ell,n} + \alpha_3 \\ [\nabla^A c^-(P,A)]_{k,\ell,n} &= \sum_m P_{m,k,\ell} \frac{V_{m,n}}{(PA)_{m,n}} + \alpha_1 (M_A)_{k,\ell,n} \\ &\quad + 2\alpha_2 (A_{k,\ell-1,n-1} + A_{k,\ell+1,n+1}) \end{aligned}$$

The update rule for A is given by:

$$A_{k,\ell,n} \leftarrow A_{k,\ell,n} \cdot \frac{[\nabla^A c^-(P,A)]_{k,\ell,n}}{[\nabla^A c^+(P,A)]_{k,\ell,n}}$$

Similarly, the gradient of the cost function with regard to P can be split as,

$$\begin{aligned} [\nabla^P c^+(P,A)]_{m,k,\ell} &= \sum_n A_{k,\ell,n} + \mathcal{I}_{\mathcal{A}}(\ell) 4\beta_1 P_{m,k,\ell} + \beta_2 \\ &\quad + \mathcal{I}_{\mathcal{B}}(\ell)\beta_3 \gamma^2 \big(P_{m,k,\ell} e^{\gamma(P_{m,k,\ell} - P_{m,k,\ell-1}) - 1} \\ &\quad + P_{m,k,\ell} e^{\gamma(P_{m,k,\ell+1} - P_{m,k,\ell}) - 1} \big) \end{aligned}$$

$$\begin{aligned} [\nabla^P c^-(P,A)]_{m,k,\ell} &= \sum_n A_{k,\ell,n} \frac{V_{m,n}}{(PA)_{m,n}} \\ &\quad + \mathcal{I}_{\mathcal{A}}(\ell) 2\beta_1 P_{m+1,k,\ell} + \mathcal{I}_{\mathcal{A}}(\ell) 2\beta_1 P_{m-1,k,\ell} + \beta_2 (M_P)_{m,k,\ell} \\ &\quad + \mathcal{I}_{\mathcal{B}}(\ell)\beta_3 \gamma^2 \big(P_{m,k,\ell-1} e^{\gamma(P_{m,k,\ell} - P_{m,k,\ell-1}) - 1} \\ &\quad + P_{m,k,\ell+1} e^{\gamma(P_{m,k,\ell+1} - P_{m,k,\ell}) - 1} \big) \end{aligned}$$

where $\mathcal{I}_{\mathcal{A}}$ and $\mathcal{I}_{\mathcal{B}}$ are the indicator functions for $\mathcal{A}$ and $\mathcal{B}$, respectively. The update rule for P is given by

$$P_{m,k,\ell} \leftarrow P_{m,k,\ell} \cdot \frac{[\nabla^P c^-(P,A)]_{m,k,\ell}}{[\nabla^P c^+(P,A)]_{m,k,\ell}}$$

To represent a pitch that is not in the score, we use a shifted version of the templates of the closest pitch in the score. While this is not problematic for the update of A (by actually creating a shifted copy), the update for P needs to be adapted as the shifted and unshifted versions need to be coupled, i.e. have to be updated jointly. Fortunately, as discussed in [51], the gradients given above for score and non-score pitches can easily be merged and thus be used to create a joint update. Due to space constraints we refer to [51] for details.

Once the activation matrix A is computed, we follow almost the same strategy as in Section III for the onset detection and note classification. We set $\hat{A}_{k,n} = \sum_\ell A_{k,\ell,n}$, i.e. we sum all activation values associated with pitch k in frame n. In particular, we found the activations resulting from our spectro-temporal dictionary to be more discriminative for the attack part compared to our previous method. Therefore, in contrast to [10], we found it useful to include the attack part in $\hat{A}$ as well.

TABLE I
PIECES FOR EVALUATION

ID	Composer	Title
1	Josef Haydn	Symphony No. 94: Andante (Hob I:94-02)
2	James Hook	Gavotta (Op. 81 No. 3)
3	Pauline Hall	Tarantella
4	Felix Swinstead	A Tender Flower
5	Johann Krieger	Sechs musicalische Partien: Bourrée
6	Johannes Brahms	The Sandman (WoO 31 No. 4)
7	Tim Richards (arr.)	Down by the Riverside

V. EXPERIMENTS

In this section, we report on experiments we conducted to evaluate the performance of our extended method compared to the previous method [10]. Further, we investigate the influence of each parameter on the extended method.

A. Dataset, Setting & Evaluation Measure

1) Dataset: We use a dataset of seven pieces shown in Table I, which were selected from the Associated Board of the Royal Schools of Music 2011/12 syllabus for grades 1 and 2. The dataset was originally introduced in [9] and the pieces were played on a Yamaha U3 Disklavier, with the pianist intentionally introducing mistakes to simulate a student deviating from the original score. In total, there are 1600 correctly played notes, 111 missing notes and 116 extra notes. For each piece, there is one audio recording, one MIDI file of the original score, and three MIDI files annotating the correctly played, missing and extra notes. The annotation files were slightly corrected in our previous work, and are available online[2]. For further details, we refer to [10].

2) Audio Input: The audio data has a sampling rate of 44100 samples per second. It was converted to a spectrogram using a Hann window, with a window size of 4096 and half overlap. Using a weighted sum, the spectrogram is converted to a log-frequency scale using a resolution of 36 bins per octave.

3) Evaluation Measure: To evaluate a method, we calculate four performance metrics, Precision, Recall, F-Measure and Accuracy as used in the Mirex evaluation campaign [54] – however, separately for each class of notes. To this end, the

[2]https://code.soundsoftware.ac.uk/projects/score-informed-piano-transcription-dataset

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/TASLP.2017.2724203, IEEE/ACM Transactions on Audio, Speech, and Language Processing

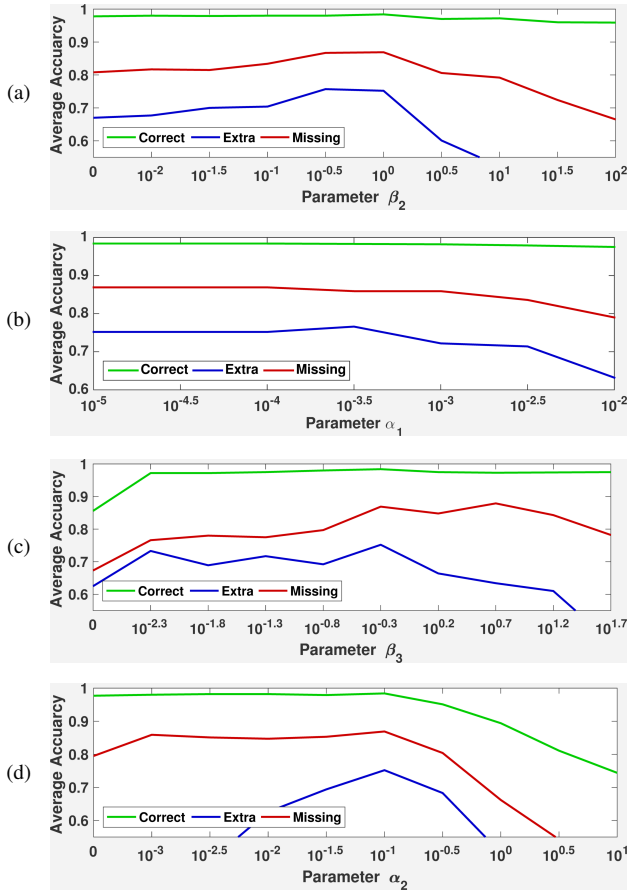


Fig. 6. Average accuracy as a function of different parameters. (a) template constraint β_2; (b) activation constraint α_1; (c) decay regularizer β_3; (d) diagonal structure regularizer α_2.

annotation MIDI files provide for the correctly played and extra notes the onset positions in the performance. For each missing note, the corresponding MIDI file provides an onset position indicating where that note would have been expected in the performance. Our proposed method ignores these annotation files and only employs the original, uninterpreted score MIDI file. More precisely, as detailed in Section III, our method compares its note detections to the original score to obtain sets of notes classified as either correctly played, extra and missing. By comparing the onset positions obtained by a method and those annotated in the ground truth, we can get the performance metrics separately for each class of notes. The allowed onset deviation for our evaluation is $T_2 = \pm 0.2$s.

B. Influence of Individual Parameters

To indicate the influence of different terms in our cost function, we conducted four groups of experiments, and for each group we change one of the parameters and fix all others. Note that we use six regularizers and corresponding weights in total. Due to space limitation, we focus on our novel terms and omit a detailed discussion of the spectral continuity regularizer $\widetilde{c}_1$ and the activation sparsity regularizer c_3 and refer instead to [10] and [30], respectively, for further details and a discussion of their behavior.

1) Soft Mask-Constraint Regularizer on Template Matrix: Fig. 6 (a) shows the average accuracy for different values of the parameter β_2, i.e. the weight balancing the importance of the term $\widetilde{c}_2$ which implements a soft mask-constraint on P, see Section IV-C. The average accuracy of identifying extra and missing notes increases with β_2 and peaks at $\beta_2 = 1$, while the average accuracy for correctly played notes remains steady. If β_2 is increased beyond this point, the accuracy for all three note classes drops, especially for the extra and missing note classes.

2) Soft Mask-Constraint Regularizer on Activation Matrix: The influence of the parameter α_1 is illustrated in Fig. 6(b). The weight controls c_1 which corresponds to a soft mask-constraint on A. The best results are obtained for $\alpha_1 = 10^{-3.5}$. Similar to the template constraint term, the average accuracy for all three note classes declines if the activity constraint becomes too strong – i.e. if activity outside the expected positions is heavily penalized and thus extra notes cannot be modeled anymore.

3) Decay Structure Regularizer on Template Matrix: The influence of the parameter β_3, which balances the importance of decay structure regularizer $\widetilde{c}_3$, is illustrated in Fig. 6 (c). The average accuracy for the correctly played note class remains relatively static after a short increase. The average accuracies for missing and extra notes show an upwards trend with β_3 first, followed by a decrease for the extra notes with $\beta_3 > 10^{-0.3}$ and a slight decrease for the missing notes with $\beta_3 > 10^{0.7}$. The overall best results are obtained for $\beta_3 = 10^{-0.3}$, which seems to represent a reasonable trade-off between model capacity and learning stability.

4) Diagonal Structure Regularizer on Activation Matrix: Fig. 6 (d) shows a plot of the average accuracy against different values for the weight α_2, which is associated with the diagonal structure regularizer c_2. The average accuracy of correctly played and missing notes only changes slightly for $\alpha_2 \in [0, 10^{-1}]$. On the contrary, the average accuracy for extra notes grows considerably with α_2, peaking at $\alpha_2 = 10^{-1}$.. The overall best results are obtained for $\alpha_2 = 10^{-1}$. These results seem to indicate that the score information alone might not be enough to guide the learning process in such a way that a physically correct diagonal structure occurs in the activations and that this regularizer is indeed needed.

5) Discussion: Overall, varying the parameters has the least influence on the class of correctly played notes and the largest influence on the extra notes. This is not really surprising as the score provides strong information about the correctly played notes and thus our regularizers are not required to provide additional help. For unexpected events, such as extra notes, however, the regularizers are of much higher importance.

One surprising observation is that the influence of the mask-constraint c_1 on A is not more pronounced in the results, as it essentially carries much of the temporal information provided by the score. Indeed, using a low value for α_1 we observed more noise in A and yet the results do not differ much. Several aspects are important to explain this behavior. First, even with a low value for α_1, we still use the score information to initialize A – which already adds a strong bias for the final result (note that, from an optimization point of view, NMF is a bi-convex problem and as such the error surface

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/TASLP.2017.2724203, IEEE/ACM Transactions on Audio, Speech, and Language Processing

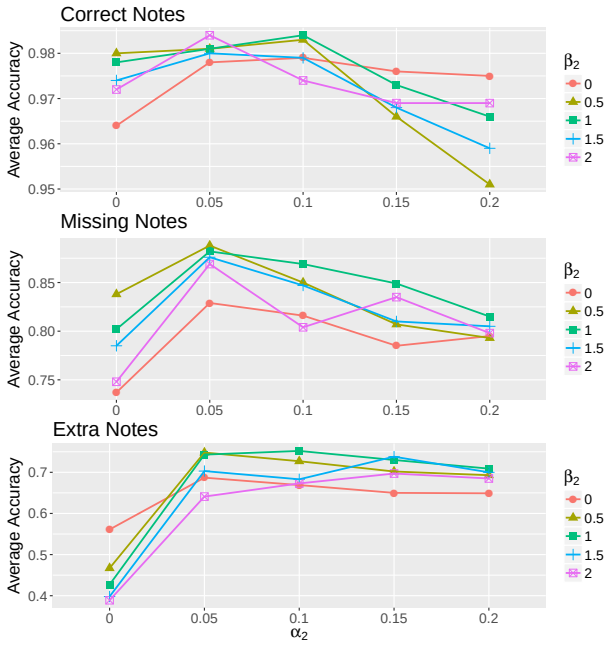


Fig. 7. Interaction between parameter α_2 and β_2 as they influence identification accuracy for all three types of notes.

contains various local minima). A second important aspect is the adaptive thresholding, which is part of our onset detection. While c_1 incorporates the note information as a soft constraint, the adaptive thresholding as described in Section IV-F employs the same information in a more binary form. In particular, our adaptive method takes the noise level in A into account when choosing a threshold. Therefore, the stability of our method with respect to α_1 can partially be attributed to the quality of the adaptive thresholding and its noise insensitivity. For more complex pieces beyond the beginner level (i.e. intermediate levels and beyond), however, we would expect the noise level to have a stronger impact on the results – such scenarios, however, were not within the scope of this paper. However, with additional data, this might be an interesting direction for the future.

With six regularizers in total, the optimization of the joint parameter space is not trivial as parameters might influence each other. Even using only five different settings for each parameter leads to $5^6 = 15625$ different configurations in a grid search. However, assuming that most dependencies are already observable in pairs of parameters (i.e. in contrast to dependencies that only occur comparing two groups of parameters jointly), we can decrease the search space drastically. In particular, with six parameters there are only $\binom{6}{2} = 15$ different combinations. Testing five different values per parameter leads to 25 configurations to be tested for each combination and thus to a total of 375 configurations and corresponding evaluations on the whole dataset.

Overall, we did not find interaction effects between most parameters, which justifies optimizing them individually. However, we found a somewhat complex interaction between the template mask-constraint parameter β_2 and the diagonal structure parameter α_2. Fig. 7 shows various plots illustrating

TABLE II
AVERAGE PERFORMANCE METRICS OF THREE METHODS FOR CORRECTLY PLAYED NOTES(C), EXTRA NOTES (E) AND MISSING NOTES (M)

Method	Class	Prec.	Recall	F-Meas.	Accur.
Extended	C	0.996	0.988	0.992	0.984
	E	0.876	0.840	0.849	0.752
	M	0.884	0.980	0.926	0.869
Previous [10]	C	0.994	0.991	0.993	0.986
	E	0.814	0.750	0.770	0.640
	M	0.928	0.970	0.945	0.899
[9]	C	-	-	-	0.932
	E	-	-	-	0.605
	M	-	-	-	0.492

the performance of our method for several combinations of the two parameters – plotted separately for the three note classes. If there would be no interaction, the plots would not cross. However, in particular for higher values of β_2, we observe interaction. For example, for extra notes, we see that for high values of β_2 the accuracy improves with higher values of α_2. For low values of β_2, the opposite happens. While this interaction might just be a property of our dataset and its size, it might also be explained by the fact that a strong β_2 leads to stronger constraints on P, which might not always be appropriate. A stronger value for α_2 might make sure in this case that 'unexplained' residual energy (resulting from enforcing incorrect constraints) is suppressed in A by other means and thus does not lead to wrong detections.

C. Comparison to the Previous Method

Next, we compare the performance of our extended method and our previous work [10]. We set the parameters to a configuration we found to perform well as described above: $\alpha_1 = 10^{-3.5}, \alpha_2 = 0.1, \alpha_3 = 0.1, \beta_1 = 10, \beta_2 = 1, \beta_3 = 0.5$.

Due to the space limitation, we only show performance metrics averaged over all seven pieces in Table II. In particular, the reported F-measure is an average over the individual F-measures values, rather than computed from the average precision and recall given in Table II. As a reference, we also include the average accuracy for the method proposed in [9] – however, note that we use here slightly modified ground truth annotations as mentioned in Section V-A.

For the correctly played notes (C), the extended method shows a similar performance as our previous work [10] on all four evaluation measures. For extra notes (E), however, our extended method considerably outperforms our previous work regarding all measures. For example, the average accuracy improves by 17%, from 0.640 to 0.752. For the missing notes (M), the extended method is slightly worse (by 3%) compared to [10] but essentially remains at a similar level.

These numbers illustrate that our proposed method is now better prepared to detect unexpected events. In particular, the soft constraints enable the modeling of extra notes already during the dictionary learning process, leading to fewer errors in the template estimations. Further, the diagonal structure regularizer leads to a physically more plausible interpretation of A and eliminates many spurious activations (e.g. singular, horizontal and vertical activations). In particular, Fig.6 (d)

showed the relative importance of the diagonal structure regularizer for identifying extra notes.

Overall, we see that our proposed method trades some precision for missing notes off in favor of recall compared to our previous work [10]. This is an effect of using a soft mask-constraint on A. In particular, as discussed above, a soft constraint leads to more noise in A. As a result, the adaptive thresholding is biased towards higher thresholds, which leads to a bias for correctly played notes in the performance not being detected and thus to an increase of missing note detections – which are incorrect. However, taking all metrics into account, the drastic improvement in terms of the detection of extra notes (the weakest aspect of our previous work) is the dominant effect we observe.

VI. Conclusion

In this paper, we introduce a score-informed transcription method to identify missing and extra notes in piano recordings. By incorporating score information into the dictionary learning process, our method yields spectral patterns for each pitch that are closely adapted to the given recording. We extended our previous work to better account for the specific characteristics of piano sounds and local deviations of the performance from the score. As demonstrated by our experiments with a dataset of pieces for piano beginners, our methods achieve high accuracy compared to a state-of-the-art method, while the extensions further improve the accuracy of our system for identifying extra notes.

One issue we would like to address in future work is the lack of data. We plan to create a new dataset to test our score-informed transcription method across a variety of scenarios. For example, in performance analysis, the number of playing mistakes can be expected to be less compared to a music tutoring application, while deviations due to musical interpretation might increase. Since our alignment method was not designed to deal with strong local changes in the order of notes, such as broken or strongly arpeggiated chords, our score constraints might provide incorrect information to the transcription process in such scenarios. Similarly, in the music tutoring application, an extremely large number of playing mistakes might occur in some cases, which might lead to the alignment getting lost and thus corrupting the transcription result. Therefore, we plan to investigate more strategies to make the score information adapt better to different application scenarios.

Another issue we would like to work on is the computational complexity. Since we extend the dictionary to L templates for each pitch (compared to two templates as in [10]), the computational costs for the dictionary learning step are $L/2$ times higher. For a music tutoring system, there might be constraints on the run-time and thus we plan to investigate strategies to lower the computational cost without affecting the overall accuracy of the system.

References

[1] 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.

[2] J. P. Bello Correa, "Towards the automated analysis of simple polyphonic music: A knowledge-based approach," Ph.D. dissertation, University of London, 2003.

[3] C. Dittmar, E. Cano, J. Abeßer, and S. Grollmisch, "Music information retrieval meets music education," in *Multimodal Music Processing*, ser. Dagstuhl Follow-Ups. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2012, vol. 3, pp. 95–120.

[4] J. Driedger, H. Grohganz, T. Prätzlich, S. Ewert, and M. Müller, "Score-informed audio decomposition and applications," in *Proceedings of the ACM International Conference on Multimedia (ACM-MM)*, Barcelona, Spain, 2013, pp. 541–544.

[5] H. Kirchhoff, S. Dixon, and A. Klapuri, "Multi-template shift-variant non-negative matrix deconvolution for semi-automatic music transcription," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2012, pp. 415–420.

[6] A. P. Klapuri and M. Davy, Eds., *Signal Processing Methods for Music Transcription*. New York: Springer, 2006.

[7] S. Ewert and M. Sandler, "Piano transcription in the studio using an extensible alternating directions framework," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 1983–1997, Nov 2016.

[8] A. Cogliati, Z. Duan, and B. Wohlberg, "Context-dependent piano music transcription with convolutional sparse coding," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 12, pp. 2218–2230, 2016.

[9] E. Benetos, A. Klapuri, and S. Dixon, "Score-informed transcription for automatic piano tutoring," in *Proceedings of the European Signal Processing Conference (EUSIPCO)*, 2012, pp. 2153–2157.

[10] S. Ewert, S. Wang, M. Müller, and M. Sandler, "Score-informed identification of missing and extra notes in piano recordings," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, New York, USA, 2016, pp. 30–36.

[11] S. Ewert, B. Pardo, M. Müller, and M. D. Plumbley, "Score-informed source separation for musical audio recordings: An overview," *IEEE Signal Processing Magazine*, vol. 31, no. 3, pp. 116–124, May 2014.

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

[13] A. Ozerov, C. Févotte, and M. Charbit, "Factorial scaled hidden Markov model for polyphonic audio representation and source separation," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2009, pp. 121–124.

[14] E. Benetos and S. Dixon, "Multiple-instrument polyphonic music transcription using a temporally constrained shift-invariant model," *Journal of the Acoustical Society of America*, vol. 133, no. 3, pp. 1727–1741, 2013.

[15] T. Cheng, S. Dixon, and M. Mauch, "Modelling the decay of piano sounds," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2015, pp. 594–598.

[16] M. G. Christensen and A. Jakobsson, *Multi-Pitch Estimation*. Synthesis Lectures on Speech and Audio Processing, Morgan and Claypool Publishers, 2009.

[17] C. Yeh, A. Roebel, and X. Rodet, "Multiple fundamental frequency estimation and polyphony inference of polyphonic music signals," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 6, pp. 1116–1126, 2010.

[18] 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*, vol. 18, no. 6, pp. 1643–1654, 2010.

[19] 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, 2012.

[20] G. E. Poliner and D. P. Ellis, "A discriminative model for polyphonic piano transcription," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, no. 1, 2007.

[21] M. Marolt, "A connectionnist approach to automatic transcription of polyphonic piano music," *IEEE Transactions on Multimedia*, vol. 6, no. 3, pp. 439–449, 2004.

[22] R. Kelz, M. Dorfer, F. Korzeniowski, S. Böck, A. Arzt, and G. Widmer, "On the potential of simple framewise approaches to piano transcription," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2016, pp. 475–481.

[23] J. Nam, J. Ngiam, H. Lee, and M. Slaney, "A classification-based polyphonic piano transcription approach using learned feature representations," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2011, pp. 175–180.
[24] S. Böck and M. Schedl, "Polyphonic piano note transcription with recurrent neural networks," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Kyoto, Japan, 2012, pp. 121–124.
[25] S. Sigtia, E. Benetos, and S. Dixon, "An end-to-end neural network for polyphonic music transcription," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 5, pp. 927–939, 2015.
[26] T. Virtanen, J. F. Gemmeke, B. Raj, and P. Smaragdis, "Compositional models for audio processing: Uncovering the structure of sound mixtures," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 125–144, 2015.
[27] A. Ozerov, E. Vincent, and F. Bimbot, "A general flexible framework for the handling of prior information in audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 4, pp. 1118–1133, 2012.
[28] E. Vincent, N. Bertin, and R. Badeau, "Adaptive harmonic spectral decomposition for multiple pitch estimation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 528–537, 2010.
[29] 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*, vol. 18, no. 3, pp. 538–549, 2010.
[30] T. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.
[31] P. Smaragdis, "Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs," in *Proceedings of the International Conference on Independent Component Analysis and Blind Signal Separation (ICA)*, Grenada, Spain, 2004, pp. 494–499.
[32] A. Roebel, J. Pons, M. Liuni, and M. Lagrangey, "On automatic drum transcription using non-negative matrix deconvolution and itakura saito divergence," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 414–418.
[33] E. Benetos and S. Dixon, "A shift-invariant latent variable model for automatic music transcription," *Computer Music Journal*, vol. 36, no. 4, pp. 81–94, 2012.
[34] S. Ewert, M. D. Plumbley, and M. Sandler, "A dynamic programming variant of non-negative matrix deconvolution for the transcription of struck string instruments," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Brisbane, Australia, 2015, pp. 569–573.
[35] E. Scheirer, "Using musical knowledge to extract expressive performance information from audio recordings," in *Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI) - Workshop on Computational Auditory Scene Analysis*, 1995, pp. 153–160.
[36] S. Ewert and M. Müller, "Estimating note intensities in music recordings," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Prague, Czech Republic, 2011, pp. 385–388.
[37] B. Niedermayer and G. Widmer, "A multi-pass algorithm for accurate audio-to-score alignment," in *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Utrecht, The Netherlands, 2010, pp. 417–422.
[38] R. B. Dannenberg, M. Sanchez, A. Joseph, P. Capell, R. Joseph, and R. Saul, "A computer-based multi-media tutor for beginning piano students," *Journal of New Music Research*, vol. 19, no. 2-3, pp. 155–173, 1990.
[39] J.-H. Wang, S.-A. Wang, W.-C. Chen, K.-N. Chang, and H.-Y. Chen, "Real-time pitch training system for violin learners," in *Proceedings of the IEEE International Conference on Multimedia and Expo Workshops (ICME)*, 2012, pp. 163–168.
[40] J. Woodruff, B. Pardo, and R. B. Dannenberg, "Remixing stereo music with score-informed source separation," in *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Victoria, Canada, 2006, pp. 314–319.
[41] K. Itoyama, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno, "Instrument equalizer for query-by-example retrieval: Improving sound source separation based on integrated harmonic and inharmonic models," in *Proceedings of the International Conference for Music Information Retrieval (ISMIR)*, Philadelphia, USA, 2008, pp. 133–138.
[42] T.-S. Chan, T.-C. Yeh, Z.-C. Fan, H.-W. Chen, L. Su, Y.-H. Yang, and R. Jang, "Vocal activity informed singing voice separation with the ikala dataset," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 718–722.
[43] S. Ewert, M. Müller, and P. Grosche, "High resolution audio synchronization using chroma onset features," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Taipei, Taiwan, 2009, pp. 1869–1872.
[44] A. Arzt, S. Böck, S. Flossmann, H. Frostel, M. Gasser, and G. Widmer, "The complete classical music companion v0.9," in *Proceedings of the AES International Conference on Semantic Audio*, London, UK, 18–20 2014, pp. 133–137.
[45] M. Müller and D. Appelt, "Path-constrained partial music synchronization," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Las Vegas, Nevada, USA, 2008, pp. 65–68.
[46] E. Nakamura, T. Nakamura, Y. Saito, N. Ono, and S. Sagayama, "Outer-product hidden markov model and polyphonic midi score following," *Journal of New Music Research*, vol. 43, no. 2, pp. 183–201, 2014.
[47] S. Wang, S. Ewert, and S. Dixon, "Compensating for asynchronies between musical voices in score-performance alignment," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2015, pp. 589–593.
[48] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems*, Denver, CO, USA, 2000, pp. 556–562.
[49] S. A. Raczynski, N. Ono, and S. Sagayama, "Multipitch analysis with harmonic nonnegative matrix approximation," in *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, 2007, pp. 381–386.
[50] P. Smaragdis, B. Raj, and M. Shashanka, "Sparse and shift-invariant feature extraction from non-negative data," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Las Vegas, Nevada, USA, 2008, pp. 2069–2072.
[51] J. Eggert, H. Wersing, and E. Korner, "Transformation-invariant representation and NMF," in *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, vol. 4, 2004, pp. 2535–2539.
[52] T. Chan, S. Esedoglu, F. Park, and A. Yip, "Recent developments in total variation image restoration," *Mathematical Models of Computer Vision*, vol. 17, 2005.
[53] T. Cheng, M. Mauch, E. Benetos, and S. Dixon, "An attack/decay model for piano transcription," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2016.
[54] J. S. Downie, "The music information retrieval evaluation exchange (2005–2007): A window into music information retrieval research," *Acoustical Science and Technology*, vol. 29, no. 4, pp. 247–255, 2008.

Siying Wang received the BSc(Eng) degree in telecommunication engineering from Beijing University of Posts and Telecommunications in 2009. She is currently pursuing her doctoral degree at the Centre for Digital Music, Queen Mary University of London (United Kingdom). Her research interests include audio signal processing, music information retrieval and musical performance study.

Sebastian Ewert received the M.Sc./Diplom and Ph.D. degrees (summa cum laude) in computer science from the University of Bonn (svd. Max-Planck-Institute for Informatics), Germany, in 2007 and 2012, respectively. After a postdoc at the Centre for Digital Music, Queen Mary University of London (United Kingdom), he became lecturer for signal processing in the centre in 2015. Currently, he is additionally holding a research position in the EPSRC programme Fusing Audio and Semantic Technologies (FAST) and is one of the leaders of the Machine Listening Lab.

Simon Dixon is a Reader (Assoc. Prof.), Director of Graduate Studies and Deputy Director of the Centre for Digital Music at Queen Mary University of London. He has a PhD in Computer Science (Sydney) and LMusA diploma in Classical Guitar. His research interests include high-level music signal analysis, computational modelling of musical knowledge, and the study of musical performance. Particular areas of focus include automatic music transcription, beat tracking, audio alignment and analysis of intonation and temperament. He was President (2014-15) of the International Society for Music Information Retrieval (ISMIR), is founding Editor of the Transactions of ISMIR, and member of the Editorial Board of the Journal of New Music Research (since 2011), and has published over 160 refereed papers in the area of music informatics.