Score-Informed Identification of Missing and Extra Notes in Piano Recordings
Ewert, S; Wang, S; Müller, M; Sandler, M

© Sebastian Ewert, Siying Wang, Meinard Muller and Mark Sandler. Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0)

For additional information about this publication click this link.
http://qmro.qmul.ac.uk/xmlui/handle/123456789/15688

Information about this research object was correct at the time of download; we occasionally make corrections to records, please therefore check the published record when citing. For more information contact scholarlycommunications@qmul.ac.uk
A main goal in music tuition is to enable a student to play a score without mistakes, where common mistakes include missing notes or playing additional extra ones. To automatically detect these mistakes, a first idea is to use a music transcription method to detect notes played in an audio recording and to compare the results with a corresponding score. However, as the number of transcription errors produced by standard methods is often considerably higher than the number of actual mistakes, the results are often of limited use. In contrast, our method exploits that the score already provides rough information about what we seek to detect in the audio, which allows us to construct a tailored transcription method. In particular, we employ score-informed source separation techniques to learn for each score pitch a set of templates capturing the spectral properties of that pitch. After extrapolating the resulting template dictionary to pitches not in the score, we estimate the activity of each MIDI pitch over time. Finally, making again use of the score, we choose for each pitch an individualized threshold to differentiate note onsets from spurious activity in an optimized way. We indicate the accuracy of our approach on a dataset of piano pieces commonly used in education.

1. INTRODUCTION

Automatic music transcription (AMT) has a long history in music signal processing, with early approaches dating back to the 1970s [1]. Despite the considerable interest in the topic, the challenges inherent to the task are still to overcome by state-of-the-art methods, with error rates for note detection typically between 20 and 40 percent, or even above, for polyphonic music [2–8]. While these error rates can drop considerably if rich prior knowledge can be provided [9, 10], the accuracy achievable in the more general case still prevents the use of AMT technologies in many useful applications.

This paper is motivated by a music tuition application, where a central learning outcome is to enable the student to read and reproduce (simple) musical scores using an instrument. In this scenario, a natural use of AMT technologies could be to detect which notes have been played by the student and to compare the results against a reference score – this way one could give feedback, highlighting where notes in the score have not been played (missed notes) and where notes have been played that cannot be found in the score.

Figure 1. Given (a) an audio recording and (b) a score (e.g. as a MIDI file) for a piece of music, our method (c) estimates which notes have been played correctly (green/light crosses), have been missed (red/dark crosses for pitch 55) or have been added (blue/dark crosses for pitch 59) in the recording compared to the score.
wavetable method, and then transcribing both the real and (under the same recording conditions) – a requirement not

To the best of the authors’ knowledge, the method presented valuable source of prior knowledge: the score. Therefore, the method is shown in Fig. 1, where correctly played notes transcription is compared to the given score, which enables the given score as closely as possible. Finally, the resulting real notes such that the resulting note onsets correspond to pitch, a threshold used to differentiate between noise and errors in the detection stage. An example output of our score to analyze the resulting activities: we set, for each note in the score, the corresponding or expected position in pitch manifests in a time-frequency representation of the audio. Next, we use this information to learn how each note in the score, the corresponding or expected position in

2. PROPOSED METHOD

2.1 Step 1: Score-Audio Alignment

As a first step in our proposed method, we align a score (given as a MIDI file) to an audio recording of a student playing the corresponding piece. For this purpose, we employ the method proposed in [13], which combines chroma with onset indicator features to increase the temporal accuracy of the resulting alignments. Since we expect differences on the note level between the score and the audio recording related to the playing mistakes, we manually checked the temporal accuracy of the method but found the alignments to be robust in this scenario. It should be noted, however, that the method is not designed to cope with structural differences (e.g. the student adding repetitions of some segments in the score, or leaving out certain parts) – if such differences are to be expected, partial alignment techniques should be used instead [14, 15].

2.2 Step 2: Score-Informed Adaptive Dictionary Learning

As a result of the alignment, we now roughly know for each note in the score, the corresponding or expected position in the audio. Next, we use this information to learn how each pitch manifests in a time-frequency representation of the audio recording, employing techniques similarly used in score-informed source separation (SISS). There are various SISS approaches to choose from: Early methods essentially integrated the score information into existing signal models, which already drastically boosted the stability of the methods. These signal models, however, were designed for blind source separation and thus have the trade-off between the capacity to model details (variance) and the robustness in the parameter estimation (bias) heavily leaned towards the bias. For example, various approaches make specific assumptions to keep the parameter space small, such as that partials of a harmonic sound behave like a Gaussian in frequency direction [16], are highly stationary in a single frame [17] or occur as part of predefined clusters of harmonics [6]. However, with score information providing extremely rich prior knowledge, later approaches found that the variance-bias trade-off can be shifted considerably towards variance.

For our method, we adapt an approach that makes fewer assumptions about how partials manifests and rather learns these properties from data. The basic idea is to constrain a (shift-invariant) non-negative matrix factorization (NMF) based model using the score, making only use of rough information and allowing the learning process to identify
the details, see also [12]. Since we focus on piano recordings where tuning shifts in a single recording or vibrato do not occur, we do not make use of shift invariance. In the following, we assume general familiarity with NMF and refer to [18] for further details. Let \( V \in \mathbb{R}^{M \times N} \) be a magnitude spectrogram of our audio recording, with logarithmic spacing for the frequency axis. We approximate \( V \) as a product of two non-negative matrices \( W \in \mathbb{R}^{M \times K} \) and \( H \in \mathbb{R}^{K \times N} \), where the columns of \( W \) are called (spectral) templates and the rows in \( H \) the corresponding activities. We start by allocating two NMF templates to each pitch in the score – one for the attack and one for the sustain part. The sustain part of a piano is harmonic in nature and thus we do not expect significant energy in frequencies that lie between its partials. We implement this constraint as in [12] by initializing for each sustain template only those entries with positive values that are close to a harmonic of the pitch associated with the template, i.e. entries between partials are set to zero, compare Fig. 2a. This constraint will remain intact throughout the NMF learning process as we will use multiplicative update rules and thus setting entries to zero is a straightforward way to efficiently implement certain constraints in NMF, while letting some room for the NMF process to learn where exactly each partial is and how it spectrally manifests. The attack templates are initialized with a uniform energy distribution to account for their broadband properties. Constraints on the activations are implemented in a similar way: activations are set to zero if a pitch is known to be inactive in a time segment, with a tolerance used to account for alignment inaccuracies, compare Fig. 2b. To counter the lack of constraints for attack templates, the corresponding activations are subject to stricter rules: attack templates are only allowed to be used in a close vicinity around expected onset positions. After these initializations, the method presented in [12] employs the commonly used Lee-Seung NMF update rules [18] to minimize a generalized Kullback-Leibler divergence between \( V \) and \( W H \). This way, the NMF learning process refines the information within the unconstrained areas on \( W \) and \( H \).

However, we propose a modified learning process that enhances the broadband properties for the attack templates. More precisely, we include attack templates to bind the broadband energy related to onsets and thus reduce the number of spurious note detections. We observed, however, that depending on the piece, the attack templates would capture too much of the harmonic energy, which interfered with the note detection later on. Since harmonic energy manifest as peaks along the frequency axis, we discourage such peaks for attack templates and favour smoothness using an additional spectral continuity constraint in the objective function:

\[
f(W, H) := \sum_{m,n} V_{m,n} \log \left( \frac{V_{m,n}}{(WH)_{m,n}} \right) - V_{m,n} + (WH)_{m,n} + \sigma \sum_{k \in A} \sum_{m,n} (W_{m,k} - W_{m-1,k})^2
\]

where the first sum is the generalized Kullback-Leibler divergence and the second sum is a total variation term in frequency direction, with \( A \subset \{1, \ldots, K\} \) denoting the index set of attack templates and \( \sigma \) controlling the relative importance of the two terms. Note that \( W_{m,k} - W_{m-1,k} = (F \ast W_{:,k})(m) \), where \( W_{:,k} \) denotes the \( k \)-th column of \( W \) and \( F = (-1, 1) \) is a high-pass filter. To find a local minimum for this bi-convex problem, we propose the following iterative update rules alternating between \( W \) and \( H \) (we omit the derivation for a lack of space but followed similar strategies as used for example in [19]):

\[
W_{m,k} \leftarrow \frac{\sum_{n} H_{k,n} \frac{V_{m,n}}{WH_{m,n}} + I_A(k) 2\sigma(W_{m+1,k} + W_{m-1,k})}{\sum_{n} H_{k,n} + I_A(k) 4\sigma W_{m,k}}
\]

\[
W_{m,k} \leftarrow \frac{\sum_{n} W_{m,k} \frac{V_{m,n}}{WH_{m,n}}}{\sum_{m} W_{m,k}}
\]

\[
H_{k,n} \leftarrow \frac{\sum_{m} W_{m,k} \frac{V_{m,n}}{WH_{m,n}}}{\sum_{m} W_{m,k}}
\]

where \( I_A \) is the indicator function for \( A \). The result of this update process is shown in Fig. 2c and d. It is clearly visible how the learning process refined the unconstrained areas in \( W \) and \( H \), closely reflecting the acoustical properties in the recording. Further, the total variation term led to attack templates with broadband characteristics for all pitches, while still capturing the non-uniform, pitch dependent energy distribution typical for piano attacks.
2.3 Step 3: Dictionary Extrapolation and Residual Modelling

All notes not reflected by the score naturally lead to a difference or residual between $V$ and $WH$ as observed also in [20]. To model this residual, the next step in our proposed method is to extrapolate our learnt dictionary of spectral templates to the complete MIDI range, which enables us to transcribe pitches not used in the score. Since we use a time-frequency representation with a logarithmic frequency scale, we can implement this step by a simple shift operation: for each MIDI pitch not in the score, we find the closest pitch in the score and shift the two associated templates by the number of frequency bins corresponding to the difference between the two pitches. After this operation we can use our recording-specific full-range dictionary to compute activities for all MIDI pitches. To this end, we add an activity row to $H$ for each extrapolated template and reset any zero constraints in $H$ by adding a small value to all entries. Then, without updating $W$, we re-estimate this full-range $H$ using the same update rules as given above.

2.4 Step 4: Onset Detection Using Score-Informed Adaptive Thresholding

After convergence, we next analyze $H$ to detect note onsets. A straightforward solution would be to add, for each pitch and in each time frame, the activity for the two templates associated with that pitch and detecting peaks afterwards in time direction. This approach, however, leads to several problems. To illustrate these, we look again at Fig. 2c, and compare the different attack templates learnt by our procedure. As we can see, the individual attack templates do differ for different pitches, yet their energy distribution is quite broadband leading to considerable overlap or similarity between some attack templates. Therefore, when we compute $H$ there is often very little difference with respect to the objective function if we activate the attack template associated with the correct pitch, or an attack template for a neighboring pitch (from an optimization point of view, these similarities lead to relatively wide plateaus in the objective function, where all solutions are almost equally good). The activity in these neighboring pitches led to wrong note detections.

As one solution, inspired by the methods presented in [21, 22], we initially incorporated a Markov process into the learning process described above. Such a process can be employed to model that if a certain template (e.g. for the attack part) is being used in one frame, another template (e.g. for the sustain part) has to be used in the next frame. This extension often solved the problem described above as attack templates cannot be used without their sustain parts anymore. Unfortunately, the dictionary learning process with this extension is not (bi-)convex anymore and in practice we found the learning process to regularly get stuck in poor local minima leading to less accurate transcription results.

A much simpler solution, however, solved the above problems in our experiments similar to the Markov process, without the numerical issues associated with it: we simply ignore activities for attack templates. Here, the idea is that as long as the broadband onset energy is meaningfully captured by some templates, we do not need to care about spurious note detections caused by this energy and can focus entirely on detecting peaks in the cleaner, more discriminative sustain part to detect the notes (compare also Fig. 2d). Since this simpler solution turned out to be more robust, efficient and accurate overall, we use this approach in the following. The result of using only the sustain activities is shown in the background of Fig. 1. Comparing these results to standard NMF-based transcription methods, these activities are much cleaner and easier to interpret – a result of using learnt, recording-specific templates.

As a next step, we need to differentiate real onsets from spurious activity. A common technique in the AMT literature is to simply use a global threshold to identify peaks in the activity. As another approach often used for sustained instruments like the violin or the flute, hidden Markov models (HMMs) implement a similar idea but add capabilities to smooth over local activity fluctuations, which might otherwise be detected as onsets [2]. We tried both approaches for our method but given the distinctive, fast energy decay for piano notes, we could not identify significant benefits for the somewhat more complex HMM solution and thus only report on our thresholding based results. A main difference in our approach to standard AMT methods, however, is the use of pitch-dependent thresholds, which we optimize again using the score information. The main reason why this pitch dependency is useful is that loudness perception in the human auditory system non-linearly depends on the frequency and is highly complex for non-sinusoidal sounds. Therefore, to reach a specific loudness for a given pitch, a pianist might strike the corresponding key with different intensity compared to another pitch, which can lead to considerable differences in measured energy.

To find pitch-wise thresholds, our method first generates $C \in \mathbb{N}$ threshold candidates, which are uniformly distributed between 0 and $\max_{k,n} H_{k,n}$. Next, we use each candidate to find note onsets in each activity row in $H$ that is associated with a pitch in the score. Then, we evaluate how many of the detected onsets correspond to notes in the aligned score, how many are extra and how many are missing – expressed as a precision, recall and F-measure value for each candidate and pitch. To increase the robustness of this step, in particular for pitches with only few notes, we compute these candidate ratings not only using the notes for a single pitch but include the notes and onsets for the $N$ closest neighbouring pitches. For example, to rate threshold candidates for MIDI pitch $P$, we compute the F-measure using all onsets and notes corresponding to, for example, MIDI pitch $P - 1$ to $P + 1$. The result of this step is a curve for each pitch showing the F-measure for each candidate, from which we choose the lowest threshold maximizing the F-measure, compare Fig. 3. This way, we can choose a threshold that generates the least amount of extra and missing notes, or alternatively, a threshold that maximizes the match between the detected onsets and the given score. Thresholds for pitches not used in the score are
We indicate the performance of our proposed method using a dataset originally compiled in [11]. The dataset comprises seven pieces shown in Table 1 that were taken from the syllabus used by the Associated Board of the Royal Schools of Music for grades 1 and 2 in the 2011/2012 period. Making various intentional mistakes, a pianist played these pieces on a Yamaha U3 Disklavier, an acoustic upright piano capable of returning MIDI events encoding the keys being pressed. The dataset includes for each piece an audio recording, a MIDI file encoding the reference score, as well as three annotation MIDI files encoding the extra, missing and correctly played notes, respectively.

In initial tests using this dataset, we observed that the annotations were created in a quite rigid way. In particular, several note events in the score were associated with one missing and one extra note, which were in close vicinity of each other. Listening to the corresponding audio recording, we found that these events were seemingly played correctly. This could indicate that the annotation process was potentially a bit too strict in terms of temporal tolerance. Therefore, we modified the three annotation files in some cases. Other corrections included the case that a single score note was played more than once and we re-assigned in some cases which of the repeated notes should be considered as extra notes and which as the correctly played note, taking the timing of other notes into account. Further, some notes in the score were not played but were not found in the corresponding annotation of missing notes. We make these slightly modified annotation files available online.

### 3.2 Metrics

Our method yields a transcription along with a classification into correct, extra and missing notes. Using the available ground truth annotations, we can evaluate each class individually. In each class, we can identify up to a small temporal tolerance the number of true positives (TP), false positives (FP) and false negatives (FN). From these, we can derive the Precision $P = \frac{TP}{TP+FP}$, the Recall $R = \frac{TP}{TP+FN}$, the F-measure $2PR/(P+R)$ and the Accuracy $A = \frac{TP}{TP+FP+FN}$. We use a temporal tolerance of $\pm 250$ms to account for the inherent difficulties aligning different versions of a piece with local differences, i.e. playing errors can lead to local uncertainties which position in the one version corresponds to which position in the other.

### 3.3 Results

The results for our method are shown in Table 2 for each class and piece separately. As we can see for the ‘correct’ class, with an F-measure of more than 99% the results are beyond the limits of standard transcription methods. However, this is expected as we can use prior knowledge provided by the score to tune our method to detect exactly these events. More interestingly are the results for the events we do not expect. With an F-measure of 94.5%, the results for the ‘missing’ class are almost on the same level as for the ‘correct’ class. The F-measure for the ‘extra’ class is 77.2%, which would be a good result for a standard AMT method but it is well below the results for the other two classes.

Let us investigate the reasons. A good starting point is piece number 6 where the results for the ‘extra’ class are well below average. In this recording, MIDI notes in the score with a pitch of 54 and 66 are consistently replaced in the recording with notes of MIDI pitch 53 and 65. In particular, pitches 54 and 66 are never actually played in the recording. Therefore, the dictionary learning process

---

1 available online: http://c4dm.eecs.qmul.ac.uk/rdr/

2 http://www.eecs.qmul.ac.uk/~ewerts/
Table 2. Evaluation results for our proposed method in percent.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>C</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>100.0</td>
<td>71.4</td>
<td>83.3</td>
<td>71.4</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
</tr>
<tr>
<td>2</td>
<td>C</td>
<td>100.0</td>
<td>99.7</td>
<td>99.8</td>
<td>99.7</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>90.0</td>
<td>81.8</td>
<td>85.7</td>
<td>75.0</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>92.3</td>
<td>100.0</td>
<td>96.0</td>
<td>92.3</td>
</tr>
<tr>
<td>3</td>
<td>C</td>
<td>99.2</td>
<td>99.2</td>
<td>99.2</td>
<td>98.4</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>100.0</td>
<td>66.7</td>
<td>80.0</td>
<td>66.7</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
</tr>
<tr>
<td>4</td>
<td>C</td>
<td>99.3</td>
<td>97.1</td>
<td>98.5</td>
<td>98.7</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>75.0</td>
<td>80.0</td>
<td>77.4</td>
<td>63.2</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>100.0</td>
<td>100.0</td>
<td>86.5</td>
<td>76.2</td>
</tr>
<tr>
<td>5</td>
<td>C</td>
<td>99.5</td>
<td>99.1</td>
<td>99.3</td>
<td>98.6</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>50.0</td>
<td>52.9</td>
<td>51.4</td>
<td>34.6</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>93.3</td>
<td>93.3</td>
<td>93.3</td>
<td>87.5</td>
</tr>
<tr>
<td>6</td>
<td>C</td>
<td>99.5</td>
<td>97.1</td>
<td>98.3</td>
<td>96.7</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>75.0</td>
<td>80.0</td>
<td>77.4</td>
<td>63.2</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>76.2</td>
<td>100.0</td>
<td>86.5</td>
<td>76.2</td>
</tr>
<tr>
<td>7</td>
<td>C</td>
<td>99.4</td>
<td>99.1</td>
<td>99.3</td>
<td>98.6</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>81.4</td>
<td>75.0</td>
<td>77.2</td>
<td>64.0</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>92.8</td>
<td>97.0</td>
<td>94.5</td>
<td>89.9</td>
</tr>
</tbody>
</table>

Avg. C 99.4 99.1 99.3 98.6

Table 3. Results reported for the method proposed in [11]. Remark: Values are not directly comparable with the results shown in Table 2 due to using different ground truth annotations in the evaluation.

<table>
<thead>
<tr>
<th>Class</th>
<th>C</th>
<th>E</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accur.</td>
<td>93.2</td>
<td>60.5</td>
<td>49.2</td>
</tr>
</tbody>
</table>

Finally, we reproduce the evaluation results reported for the method proposed in [11] in Table 3. It should be noted, however, that the results are not directly comparable with the results in Table 2 as we modified the underlying ground truth annotations. However, some general observations might be possible. In particular, since the class of ‘correct’ notes is the biggest in numbers, the results for this class are roughly comparable. In terms of accuracy, the number of errors in this class is five times higher in [11] (6.8 errors vs 1.4 errors per 100 notes). In this context, we want to remark that the method presented in [11] relied on the availability of recordings of single notes for the instrument in use, in contrast to ours. The underlying reason for the difference in accuracy between the two methods could be that instead of post-processing a standard AMT method, our approach yields a transcription method optimized in each step using score information. This involves a different signal model using several templates with dedicated meaning per pitch, the use of score information to optimize the onset detection and the use of pitch-dependent detection thresholds. Since the number of notes in the ‘extra’ and ‘missing’ classes are lower, it might not be valid to draw conclusions here.

4. CONCLUSIONS

We presented a novel method for detecting deviations from a given score in the form of missing and extra notes in corresponding audio recordings. In contrast to previous methods, our approach employs the information provided by the score to adapt the transcription process from the start, yielding a method specialized in transcribing a specific recording and corresponding piece. Our method is inspired by techniques commonly used in score-informed source separation that learn a highly optimized dictionary of spectral templates to model the given recording. Our evaluation results showed a high F-measure for notes in the classes ‘correct’ and ‘missing’, and a good F-measure for the ‘extra’ class. Our error analysis for the latter indicated possible directions for improvements, in particular for the dictionary extrapolation step. Further it would be highly valuable to create new datasets to better understand the behaviour of score-informed transcription methods under more varying recording conditions and numbers of mistakes made.

Acknowledgements: This work was partly funded by EPSRC grant EP/L019981/1. The International Audio Laboratories Erlangen are a joint institution of the Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) and Fraunhofer Institut für Integrierte Schaltungen IIS. Sandler acknowledges the support of the Royal Society as a recipient of a Wolfson Research Merit Award.
5. REFERENCES


