On Detecting Repeated Notes in Piano Music

Matija Marolt
University of Ljubljana
Faculty of Computer and Information Science
Trzaska 25
1000 Ljubljana
+386 1 4768483
matija.marolt@fri.uni-lj.si

Sasa Divjak
University of Ljubljana
Faculty of Computer and Information Science
Trzaska 25
1000 Ljubljana
+386 1 4768483
sasa.divjak@fri.uni-lj.si

ABSTRACT
One of the problems encountered in music transcription is to produce an algorithm that detects whether a note should be repeated, when a new onset is found during its duration, or not; with other words whether two or more shorter notes should be produced instead of a single longer note. The paper describes our approach to solving this problem, implemented within our system for transcription of piano music [4]. The approach is based on a multilayer perceptron neural network, trained to recognize repeated notes. We compare this method to a more naive method that tracks the amplitude of the first partial of each note and also present performance statistics of our system on transcriptions of several real piano recordings.

1. INTRODUCTION
Transcription of polyphonic music (polyphonic pitch recognition) can be defined as a process of converting an acoustical waveform into a parametric representation, where notes, their pitches, starting times and durations are extracted from the waveform. Transcription is a difficult problem and most research efforts in building transcription systems are directed into partial tracking and note recognition algorithms, which are the central part of all current transcription systems.

Even with a perfect note recognition score, one of the problems that each transcription system should handle in one way or another is detection of repeated notes. This can be a difficult problem, even if the played instrument has pronounced onsets (i.e. piano). An illustration of the problem is given in Figure 1.

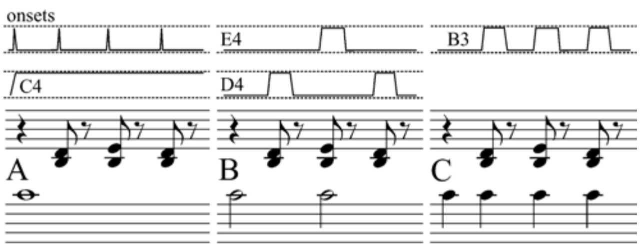

Figure 1. Different interpretations of note recognition output

The upper part of the figure shows hypothetical outputs of onset detection and note recognition algorithms on an unknown piece of music. Four onsets and four notes were found; note C4 lasts through the entire duration of the piece, while other notes appear for shorter periods of time. Four transcription examples show four possible interpretations of these outputs. Interpretations differ in the way note C4 is handled; it could be transcribed as one whole note, four quarter notes... Altogether eight combinations are possible, and all of them are consistent with outputs of onset detection and note recognition algorithms.

Although several transcription systems have recently been developed, few have tackled this problem in any way. Some authors have simply ignored errors related to repeated notes [1], some put constraints on the transcribed signal [2] (i.e. by restricting the minimal offset-onset distance), while some dealt with the problem implicitly by including instrument models into the transcription process. The paper presents our experiences in handling repeated notes within our system for transcription of piano music [4].

2. DETECTING REPEATED NOTES IN PIANO MUSIC
Problems related to repeated notes are quite common in piano music, especially in pedaled parts, where long sustained notes can become a source of many errors. To make matters worse, strong onsets can temporarily mask sustained notes, resulting in fragmented output of the note recognition module and consequently in many spurious repeated notes. An illustration of the problem is given in Figure 2: it shows hypothetical outputs of the onset detector and note recognition module on a piano piece shown in example A. Strong onsets of B3D4 and B3E4 chords temporarily mask the sustained C4 note, resulting in fragmented output of the note recognition module and consequently in three spurious notes.

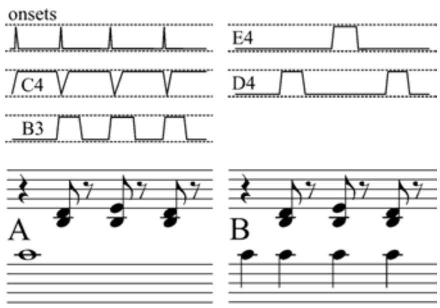

Figure 2. Masking of a sustained note

To solve the problem, we first devised a simple algorithm, which tracks amplitude envelopes of partials of found notes. At each onset, all notes found to be present in the signal up to 50 ms before the onset and 100 ms after the onset (depending on the pitch) are taken as candidate notes for repetition. The algorithm then compares amplitudes of the first four partials of candidate notes before and after the onset. All notes with an increase in amplitude that exceeds a certain threshold dependant on the pitch are repeated. The approach works reasonably well, but we wanted to improve it further by including additional features into the detection algorithm.

In our current approach, we use a multilayer perceptron (MLP) neural network to determine which notes should be repeated in the transcribed score. We chose the MLP network, because it is a well established method for solving classification problems, and we had good experiences in using these networks for transcription tasks [4]. The MLP is activated at each new onset for all potential repeated notes (sustained notes that are active before and after the onset) and decides which notes should be repeated. After many tests and experiments, we chose the following parameters to be included in the network's input vector:

- amplitude differences of the first four partials of the potential repeated note after and before the new (repeated) onset and after and before its original onset;
- weighted averages of the above amplitude differences. Weights represent amplitude ratios of the first four partials of a note, calculated from average spectral templates of piano notes;
- weighted averages as above, but only of partials with amplitude differences that fall within the standard deviation of all amplitude differences. These averages were included to eliminate outliers, which might occur because of new notes that share partials with the potential repeated note;
- amplitude differences and both weighted averages of a potential repeated note after average spectral templates of other notes starting at the new (repeated) onset have been subtracted from the frequency spectrum;
- time difference between the original and the new onset;
- differences of partial group strengths before and after the new onset. Partial group strengths are a product of the partial tracking module, described in [4].

Altogether, input vectors of the MLP consist of 26 parameters. MLP contains 5 neurons in the hidden layer and one output neuron, which indicates whether a note should be repeated. The network was trained on pairs of input/output patterns taken from our piano music database consisting of over 120 synthesized piano pieces of various styles, including classical from several periods, jazz, blues and pop.

3. RESULTS

We tested the performance the MLP by transcribing a set of 40 synthesized and real piano recordings (some examples can be found on *http://lgm.fri.uni-lj.si/SONIC*). We compared its performance to the ideal algorithm, where repeated notes were estimated from the transcription of the recording, to a system that never produced repeated notes and to the "amplitude envelopes" approach described in the previous section. A summary of results is presented in Table 1. The second column of the table shows the percentages of correctly found notes, while the third column lists percentages of spurious notes (notes not present in the input, but found by the system) in all pieces.

Table 1. A comparison of different approaches

	correct	spurious
transcription	90.4%	6.9%
no repeated notes	75.6%	6.9%
amplitude envelopes	89.1%	12.3%
MLP neural network	88.7%	9.0%

As expected, the ideal algorithm produces the best results. When no repeated notes are produced by the system, the percentage of correctly found notes diminishes substantially. The method that tracks amplitude envelopes of the first four partials of a note and the MLP network produce somewhat similar results; the amp. env. method produces more spurious notes and some more correct notes, while the MLP produces a sort of compromise between the number of spurious and correct notes. The improvement gained by using the MLP, however, is not as substantial as we had initially hoped.

Analysis of transcriptions shows that repeated notes still represent the second major cause of errors, octave errors being the most common one. Error analysis of transcriptions of three real recordings of piano music is presented in table 2. The pieces are: (1) J.S. Bach, English suite no. 5, BWV810, 1st mvm., performer Murray Perahia, Sony Classical SK 60277; (2) R. Schumman, Träumerei, performer Cyprien Katsaris, TELDEC 75863; (3) S. Joplin, The Entertainer, performer unknown, MCA 11836.

Table 2. Performance statistics on real recordings

	notes		total	octave	repeated
1	1351	missed	11.2%	39.1%	17.2%
		spurious	12.9%	72.5%	30.8%
2	458	missed	20.3%	55.9%	3.2%
		spurious	11.5%	78.6%	24.5%
3	1564	missed	11.3%	71.3%	12.9%
		spurious	14.2%	81.4%	11.2%

The second column of the table shows the total number of notes in each piece. Percentages of missed and spurious notes are given in column 4, while columns 5 and 6 show percentages of octave and repeated note errors with regard to all missed/spurious notes (both are frequently combined, so the sum can exceed 100%). Repeated note errors represent around 10% of all missed notes and approx. 21% of spurious notes. They are often combined with octave errors (a note is mistakenly repeated or a repetition missed because of a note an octave apart appears in the score), which is one of the reasons that they are so difficult to detect. The percentage of repeated note errors is especially high in quiet pedaled passages (Träumerei is a good example), where it sometimes exceeds 80%. We are working on a different set of features to include in MLP training that will hopefully improve current results.

4. CONCLUSION

The presented method of detecting repeated notes in piano music provides an improvement over the naive amplitude envelope tracking method, but still lacks accuracy that would be satisfying. We contribute most errors to the inadequacy of the feature set used for training MLP networks and are further exploring new features to improve the results. We first plan to experiment with using different features for notes in different registers (such as low, middle, high), as these have quite different amplitude envelopes, and should be treated differently by the detection algorithm.

5. REFERENCES

[1] L. Rossi, Identification de Sons Polyphoniques de Piano. Ph.D. Thesis, L'Universite de Corse, France, 1998.

[2] A.D. Sterian, Model-based Segmentation of Time-Frequency Images for Musical Transcription. Ph.D. Thesis, Univesity of Michigan, 1999.

[3] K. Kashino, K. Nakadai, T. Kinoshita, H. Tanaka, "Application of Bayesian Probability Network to Music Scene Analysis," in "Computational Auditory Scene Analysis", Lawrence Erlbaum Associates, 1998.

[4] M. Marolt, "SONIC : transcription of polyphonic piano music with neural networks," in Proceedings of Workshop on Current Research Directions in Computer Music, Barcelona, Pompeu Fabra University, 2001, pp. 217-224.