

Robust Automated Music Transcription Systems

Presented at the 1996 International Computer Music Conference, Hong Kong

Andrew Sterian
asterian@umich.edu

Gregory H. Wakefield
ghw@eecs.umich.edu

University of Michigan
Dept. of Electrical Engineering and Computer Science
Ann Arbor, MI 48109-2122 U.S.A.

Abstract

We define an automated music transcription system as a sequence of processing modules, beginning with acoustic acquisition and ending with a pitch sequence. This modular approach allows the various system components to be studied and optimized separately. Our own single-voice transcription system is presented as a working example of such modular designs. To assess the performance of such transcription systems under real-world constraints, we propose a single numerical score based on matching known events in the source material to corresponding events in the transcription. We use this score to evaluate our transcription system under several conditions.

1. Introduction

We present our music transcription system as a precursor to a larger project concerning aspects of polyphonic transcription. Our goal is to maintain a modular approach, without strong feedback or feedforward data paths in the system. A modular design raises the issue of how well each module performs its task and how this performance, as well as the overall system performance, can be measured. As more commercial transcription systems become available, this issue of quantifying transcription performance becomes important for product evaluation.

2. Transcription Implementation

The first stage of our transcription algorithm forms a time-frequency representation of the audio waveform. The short-time Fourier transform, or spectrogram, is a commonly used time-frequency representation, but it suffers from a tradeoff in time and frequency resolution due to the need for data windowing [Cohen, 1989]. We use the modal distribution [Pielemeier and Wakefield, 1996] as our time-frequency representation. This distribution has been designed specifically for musical analysis. Compared to the spectrogram, the modal distribution offers both better time and frequency resolution. The modal distribution has also been shown to compare favorably to a time-frequency representation based upon wavelet analysis.

The next stage in transcription is peak picking and track formation. Peaks are recognized as local maxima that satisfy various threshold criteria. *Tracks* are formed by finding peaks closely related in time, frequency, and amplitude. Thresholds for maximum frequency deviation and maximum amplitude change are used as criteria for connecting two peaks that are adjacent in time. A set of connected peaks that is continuous in time is called a *track*. Each track is assigned a single frequency based upon the power-weighted average frequency of its peaks, and an amplitude measure that is simply the amplitude of the highest peak in the track. Tracks are filtered (i.e., split, connected, or removed) based upon properties such as duration, time gap between other similar-frequency tracks, and large amplitude changes.

In the next transcription stage, *track groups* are formed by finding sets of tracks that are harmonically related and that have similar onset times. To add a track to an existing track group, the track must have a frequency that is an integer multiple of the track group's lowest frequency (within a prescribed inharmonicity tolerance), and it must have an onset time that is close to the remaining tracks in the group (again, within some tolerance).

Finally, each track group is considered to be a potential single note from the source material. With the assumption that the source material contains no polyphony, track groups that occupy the same regions in time are resolved by eliminating the track group with the lesser power. Each track group that remains gives rise to a note event that is

characterized by onset time, offset time, frequency, and amplitude.

A final stage converts this note event information into a MIDI file. This stage exists simply for convenience and is not considered to be part of the transcription system.

3. Transcription Quality

The performance of a transcription system can be measured quantitatively by comparing features in the transcription with the source material. This comparison is especially simple when the source material is a parameterized musical description, such as a MIDI file.

We propose a numerical goodness-of-transcription score that is a single weighted sum of quantitative differences between features of the transcription and the source material. In our work, we consider only the most salient features of a note stream, the frequency of the notes and the times of onset and offset. We also penalize spurious notes (notes not in the source material) as well as missed notes. Weights that multiply observed differences are chosen arbitrarily but roughly match perceptual impressions. For example, the penalty for estimating the wrong note frequency is greater than the penalty for an onset time more than 50ms away from the true onset. An incorrect note in a musical passage is more objectionable than the correct note played slightly too early or too late.

Interest in quantifying transcription performance reaches beyond our own transcription system. Commercial transcription systems are now becoming available and we need tools for evaluating how well they perform. Given that any system will eventually fail when presented with input data of increasingly poor quality, we also need to know how these systems degrade in performance as the input audio waveform is distorted or corrupted in some fashion. Another issue, that of determining which audio waveforms are appropriate test cases for measuring transcription performance, is not addressed here.

3.1. Robustness Experiments

In the practical application of a transcription system, we expect certain types of constraints to impact on system performance based on the algorithms that make up the system. In the experiments presented below we consider two such challenges to the algorithms, reverberation and rapid tempos. We also address the real-time processing issue by measuring the impact of a lowered sampling rate on transcription performance (simulated by lowpass filtering).

For the reverberation and lowpass filtering tests, the source material consisted of a 2.3s 17-note representative passage from a MIDI file. This MIDI file was rendered to an audio waveform using an FM synthesis voice with a harpsichord quality. The lowest note in the passage was C3 and the highest was G4.

In the reverberation test, the audio waveform is made reverberant by adding time-shifted and amplitude-scaled copies of the waveform to itself at regular intervals (the inter-echo spacing). The leftmost graph in the figure shows the performance of the transcription system as the inter-echo spacing is increased from 5ms to 100ms.

In the same-note repetition test, a single 100ms note is extracted from the audio waveform. This note is then used to create a new audio waveform by repeating the note at regular intervals with a prescribed inter-note interval (onset-to-onset time). The middle graph in the figure shows the transcription system performance as the inter-note interval is decreased from 200ms to 10ms.

In the lowpass filtering test, the same audio waveform is first passed through a lowpass filter prior to transcription. The rightmost graph in the figure shows the transcription system performance as the lowpass cutoff frequency is decreased from 1000 Hz to 110 Hz. In all three graphs, the vertical axis includes the maximum achievable score for each individual test

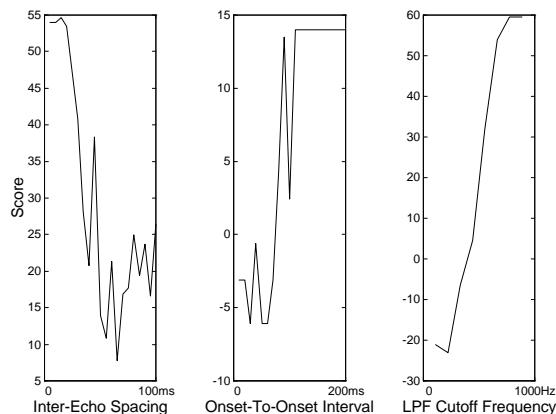


Figure 1: Transcription performance under degraded conditions

3.2. Experiment Results

The results of the reverberation experiment reveal that the transcription algorithm is very sensitive to this effect, even for reverberation delays that are commonly found in recorded music. It is not surprising that echoes in the signal can greatly confuse a transcription algorithm, since scaled and

time-shifted replicas of the notes are likely to be picked up as separate notes. Nevertheless, this experiment reveals a need to improve the algorithm's robustness against reverberation.

The same-note repetition experiment yields much better results. The transcription algorithm does not start to break down until there is no gap between the offset of one note and the onset of the next (100ms onset-to-onset interval). This means that the algorithm is capable of transcribing at least 9 individual note events per second, possibly more if each note event was shorter than 100ms.

The lowpass filtering experiment suggests good transcription performance even when little of a note's spectral content is available. Since the highest note in the passage was G3, a 1000Hz spectral cutoff eliminated all but the fundamental and a single harmonic for this note.

4. Polyphonic Transcription

Our transcription algorithm is currently being extended to tackle the problem of multi-voice, or polyphonic transcription. With the modular structure that we have in place, a simple first-try at this extension is to simply identify all track groups as potential notes and not apply the monophonic constraint for rejecting track groups. Although this approach gives rise to some practical problems (such as note segmentation and spurious tracks due to the time-frequency representation), there are other higher-level issues to consider.

For example, two instruments playing in unison poses an interesting challenge. There are two voices but only one track group represents each set of simultaneous notes. This can be seen as a limiting

case of the reverberation experiment we have described above. Should this be transcribed as a single voice or as two voices? And if the answer is the latter, do we have the tools for resolving two track groups from separate instruments that are coincident in time and frequency?

With the polyphonic transcription problem in mind, the usage of the modal time-frequency distribution as the front-end processing stage becomes ever more important due to its high resolution properties. An added benefit of this distribution is its separable kernel, giving it an advantage in computational cost as compared to other time-frequency techniques.

References

- [Cohen, 1989] Leon Cohen. Time-Frequency Distributions - A Review. *Proc. of the IEEE*, **77**(7): pp. 941-980, July 1989.
- [Fernández-Cid and Casajús-Quirós, 1994] Pablo Fernández-Cid and Francisco J. Casajús-Quirós. DSP Based Reliable Pitch-to-MIDI Converter by Harmonic Matching. *Proc. Intl. Computer Music Conf.*, 1994, pp. 307-310.
- [Fitz *et al.*, 1995] Kelly Fitz, Lippold Haken, and Bryan Holloway. Lemur - A Tool for Timbre Manipulation. *Proc. Intl. Computer Music Conf.*, 1995, pp. 154-157.
- [Pielemeier and Wakefield, 1996] William J. Pielemeier and Gregory H. Wakefield. A high-resolution time-frequency representation for musical instrument signals. *J. Acoust. Soc. Am.*, **99**(4): pp. 2382-2396, 1996.