Computationally Inexpensive and Effective Scheme for Automatic Transcription of Polyphonic Music

Weilun Lao[1], Ek Tsoon Tan[2], Alvin H. Kam[1]
[1]*Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore*
[2]*Department of Electrical &Computer Engineering, National University of Singapore*

Abstract

We describe a computationally inexpensive but effective scheme for automatic transcription of polyphonic music. Based on a two-step strategy: tracks creation and tracks grouping, the scheme utilizes innovative comb-filtering and 'sharpening' steps to produce desired transcription output in the form of discrete notes with temporal, pitch and amplitude attributes. Recall and precision measurements are used to analyze and quantify transcription accuracy. Promising results are achieved for the automatic transcription of both synthetic and acoustic piano music.

1. Introduction

Automatic music transcription aims to convert audio signals into musical scores with limited human interaction. Systems for automatic music transcription have been studied since 1975. As work on transcription of monophonic signals is relatively mature, current research focuses mainly on polyphonic signals [1][2][3][4].

The most basic musical symbol is a note, and note detection consists of two non-trivial tasks: onset detection and pitch identification. Polyphonic note detection in addition requires dealing with the coincidence of frequencies (harmonics) belonging to different notes. As a musical score could be very long, computationally efficient schemes for automatic music transcription are important.

In this paper, we describe a computationally inexpensive but effective scheme that transcribes monophonic and polyphonic music produced from a single instrument. The problem is divided into two steps: tracks (group of notes) creation and tracks grouping, with the final output containing the temporal, pitch and amplitude information of discrete notes.

Onset detection is performed using constant time windows without making any assumptions on the instrument amplitude profile. Despite not taking phase information into account for simplicity, our system is able to handle both digital (synthetic) and acoustic signals well.

To achieve our goal, we make use of a comb-filter without training on the musical instrument and pitch frequency involved. Comb-filtering proves to be a fast and effective method for detecting semi-tones, assuming that the instrument's pitch is well calibrated.

'Sharpening' used in the tracks creation process is another contribution of this paper. It generally results in significant reduction of noise about "actual" notes and harmonics. Sharpening also reduces the possibility of erroneous lengthening of notes that are a semi-tone apart. The effects of sharpening are better with lower frequencies, because of the narrower frequency intervals.

A constant spectral profile is applied for the detection of harmonics for all notes with promising results. This scheme however seems to work better for synthetic compared with acoustic music as acoustic harmonics tend to vary for different notes.

The methodology used in the system is explained in detail in section 2. The system performs parts of the blackboard architecture [1] using a power-spectrogram for analysis, as well as an extension to Dixon's scoring system [2] for performance measurements. Complementing our experimental results are critical analysis and discussion followed by concluding remarks.

2. Methodology

The transcription process is divided into two parts – tracks (consisting of a group of notes) creation and tracks grouping. As shown in Figure 1, tracks creation consists of three main steps: pre-processing, frequency detection and post-processing. For the pre-processing step, the input monophonic or polyphonic music is first formatted into a wave file. A high sampling rate of 22.05 kHz at 16-bit is used in capturing 4 octaves of notes from A1 to A5 (110 Hz to 1760 Hz). This is followed by low-pass filtering using a 9th-order Butterworth low-pass filter with a cut-off

0-7803-8603-5/04/$20.00 ©2004 IEEE

frequency slightly higher than A7 (7040 Hz) (to facilitate harmonic detection up to at least the 3rd harmonic for the track grouping process later on).

The first step for the subsequent frequency detection is to calibrate the pitch of the entire dynamic range (based on the assumption that the instrument has been accurately calibrated). This is done by obtaining maximum-frequency samples from the input. Discrete Fourier Transform (implemented using FFT) is then performed using half-overlapping constant time-windows.

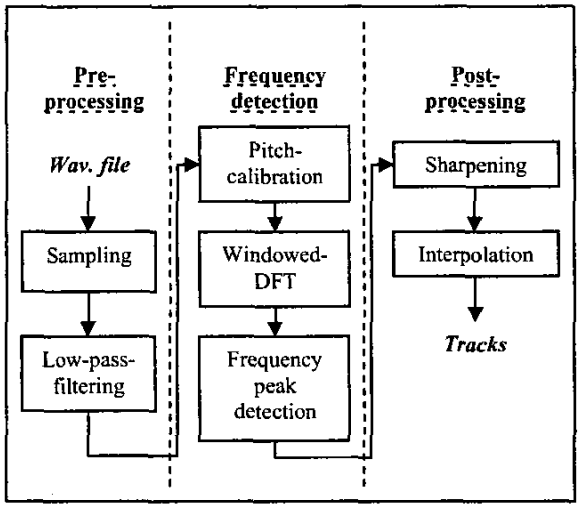

Figure1. Tracks creation process

For peak detection, a unique tool, in addition to second-order discrete derivatives and amplitude constraints, is used: a discrete variable-window comb filter (Figure 3). This comb filter incorporates the pitch offset from the pitch calibration step together with increased detection intervals for higher frequencies. Moreover, it enables the convenient access to accurate correspondence between frequency and note. For instance, the detected frequency 440Hz corresponds to the note A3 through the comb filter. The output of the next peak detection step is an effective amplitude profile of the input, which could be illustrated on a log-linearized frequency axis within a power spectrogram (Figure 4a). This is the so-called 'output track'. Finally, the post-processing step consists of sharpening and interpolation. A sharpening filter is implemented to make detected frequency peaks more outstanding by adjusting their neighboring amplitudes. Meanwhile, it succeeds in attenuating noisy detection of semi-tones. Then the interpolation of the tracks is performed in time domain for smoothening (to compensate the effects of using a constant time-window for the prior FFT implementation).

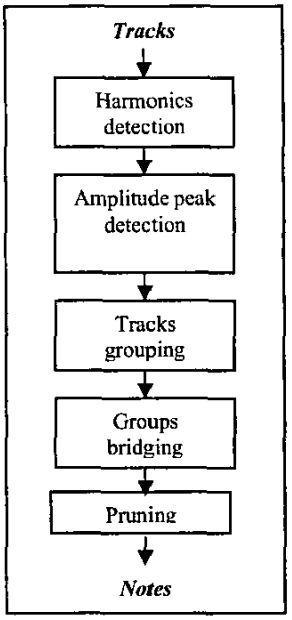

Figure 2. Tracks grouping process

For tracks grouping as shown in Figure 2, the harmonics of the tracks are first detected for every non-zero frequency. Next, a second-order derivative peak detection scheme is applied to the amplitude of all notes within the detection range. This is done by reasonably assuming that all notes should produce an amplitude peak in the fundamental frequency. By training the amplitude ratio of the harmonics to the fundamental frequencies [6], notes coinciding with harmonics (especially octaves) could be determined.

Grouping uses every detected peak to determine the duration of each peak. A combination of grouping rules on i) the number of harmonics, and ii) the amplitude of the

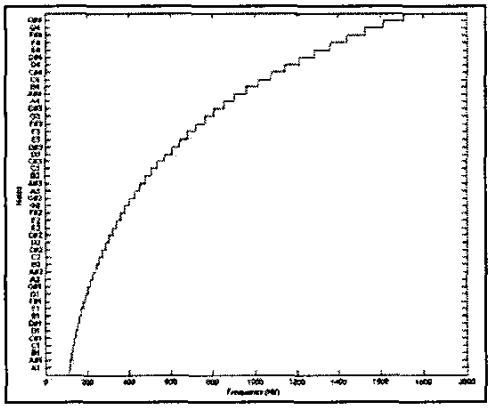

Figure 3. Discrete variable-window comb-filter

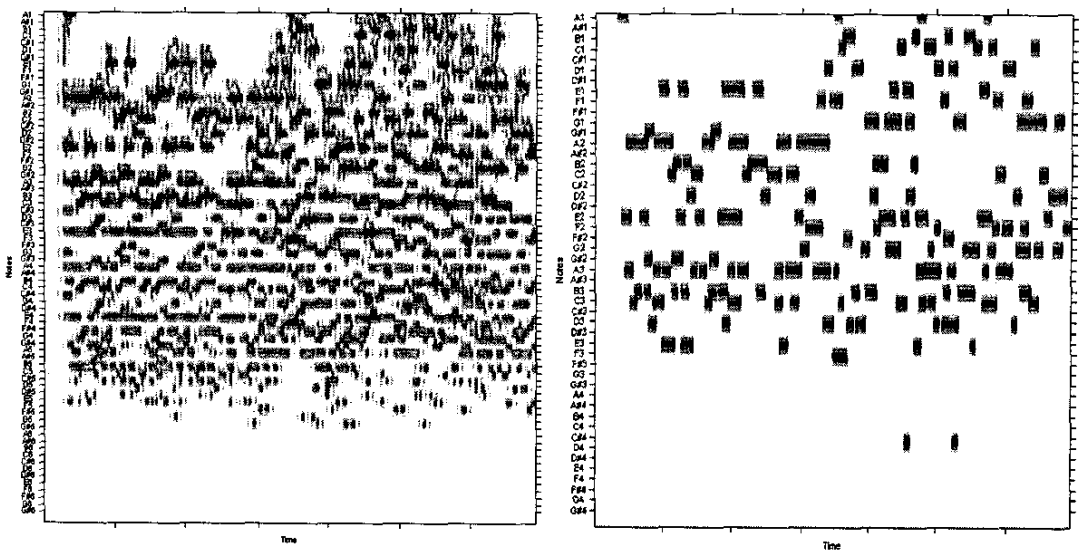

Figure 4. Power spectrogram (notes against time) with darkness indicating log-amplitude of notes for a piece of synthetic polyphonic music: a) Tracks prior to grouping (A1 to A7) [left]; b) Tracks after grouping (A1 to A5) [right]

peaks – are used to increase robustness. Neighboring and overlapping groups with the same notes are then grouped together. Finally, fuzzy and hierarchical schemes operating on tones' maximum amplitudes and durations are employed to prune remaining errors from the final output. Note parameters of the transcripted music include the: i) onset time, ii) duration, iii) pitch, and iv) amplitude – all log-normalized and scaled to the original input sample. A power spectrogram of the detected notes could also be used as an alternative illustration to standard musical notation (Figure 4b).

3. Experiments and Results

Different instruments have different harmonics and temporal characteristics. For our experiments, piano is the chosen instrument due to its sharp attack, symmetric waveforms (when played without right sustain pedal and without excessive holding), and its relatively low harmonic content.

Synthetic and acoustic piano pieces constituted our dataset. While synthetic pieces were digitally recorded, acoustic pieces were recorded using a corded microphone. The pieces and the subsequent analysis were conducted by professional musicians. In general, synthetically produced music has much better calibrated frequencies and harmonics while acoustic music is susceptible to various noises from the environment. The algorithms were first tested on synthetic music prior to acoustic music. Monophonic and polyphonic (up to four member chord) were used. As the focus was to achieve music transcription of acoustic music, more acoustic data were used.

Recall and precision of note detection were used to measure the quality of the transcription. Recall is defined as:

$$\mathrm{Re}call = \frac{\text{the number of correct notes detected}}{\text{the actual number of notes}} \tag{1}$$

while precision is defined as:

$$\mathrm{Pr}ecision = \frac{\text{the number of correct notes detected}}{\text{the number of all notes detected}} \tag{2}$$

Notes that were considered falsely detected included: i) octaves of correct notes, ii) repeated correct notes, iii) non-octave harmonics of correct notes, iv) others.

Detection of synthetic music was generally more accurate compared to acoustic music as shown in Tables I and II. The mean recall and precision rates were 98% and 92% for synthetic music, and 89% and 76% for acoustic music respectively. False notes were subdivided into octaves of correct notes (34%), repeated correct notes (43%), non-octave harmonics (22%) and others (1%). With the exception to errors classified as "others", the false notes could easily be rectified by fine-tuning the tracks-grouping procedure.

The contribution of our innovation in using comb-filtering

and sharpening in the tracks creation process were also evaluated. When either was removed, the number of false notes detected more than doubled.

Table I: Recall and Precision of note detection on synthetically produced piano music

Monophonic/ Polyphonic	Tune	Notes Detected	Recall	Precision
Monophonic	Piano1	58	1.00	1.00
Monophonic	Piano2	63	0.92	0.97
Polyphonic	Piano3	151	1.00	0.87

Table II: Recall and Precision of note detection on acoustically produced piano music

Monophonic/ Polyphonic	Tune	Notes Detected	Recall	Precision
Monophonic	Scale1	16	1.00	0.94
Monophonic	Scale2	101	0.86	0.69
Monophonic	Green	22	0.94	0.77
Polyphonic	Sonata1	85	0.78	0.74
Polyphonic	Chord1	76	0.98	0.82
Polyphonic	Chord2	54	0.85	0.72
Polyphonic	Chord3	82	0.90	0.90
Polyphonic	Chord4	70	1.00	0.80
Polyphonic	Air1	71	0.83	0.61

4. Analysis and Discussion

Along with a reduced set of assumptions, onset detection of notes for automatic music transcription is significantly simplified. For example, the DFT window size is assumed to be constant. In general however, the tempo of music could vary throughout, requiring varying window sizes for optimum detection [7]. Although this problem is compensated using temporal interpolation for smoothening, occasional spurious notes could still be produced.

The comb-filter proves to be a convenient and fast method for detecting semi-tones. This is based on the assumption that the instrument was well-calibrated; this might not always be the case in practice, especially for instruments which rely on non-discrete enunciation of notes.

Although the accuracy of the scheme is commendable, most detection errors could be attributed to the simplification of not taking phase information into account. Performance could generally be improved using computationally more sophisticated onset detection algorithms [8].

5. Conclusion

A computationally simple yet effective scheme for automatic transcription of polyphonic music is described. The scheme basically operates using a two-step approach: tracks creation and tracks grouping. A discrete variable window-size comb-filter without required training on the musical instrument and pitch frequency is used to good effect. Additionally, the innovative use of a sharpening filter to attenuate noisy detection of semi-tones is found to be effective.

Overall results indicate high recall and precision rates, illustrating that accurate note detection can be achieved for synthetic and acoustic piano music containing up to four member chord. Although our scheme is much simpler, results are comparable to previous studies [2][3] using more sophisticated methods.

Acknowledgement

We would like to thank Dr Terence Sim from the School of Computing, National University of Singapore, for his kind advice.

References

[1] K.D. Martin. A Blackboard System for Automatic Transcription of Simple Polyphonic Music. M.I.T Media Laboratory Perceptual Computing Section Technical Report No.385, MIT, 1996

[2] S. Dixon. On the Computer Recognition of Solo Piano Music. In Australian Computer Music Conference, Brisbane, Australia, 2000, pp31-37.

[3] G. Monti, M. Sandler. Automatic Polyphonic Piano Note Extraction Using Fuzzy Logic in a Blackboard System. In Proc. 5th Int. Conf. on Digital Audio Effects (DAFx-02), Hamburg, Germany, September 26-28, 2002.

[4] J.P.Bello. Towards the Automated Analysis of Simple Polyphonic Music: A knowledge-based Approach, Queen Mary, University of London, Ph.D thesis, 2003

[5] M. Marolt. SONIC: Transcription of Polyphonic Piano Music with Neural Networks. In Proceedings of Workshop on Current Research Directions in Computer Music, Barcelona, November 15-17, 2001.

[6] S. W. Foo, E. W. T. Lee. Transcription of Polyphonic Signals Using Fast Filter Bank, IEEE ISCAS 2002, vol. III pp.241-244

[7] E. D. Scheirer. Tempo and Beat Analysis of Acoustic Musical Signals. J. Acoust. Soc. Am., 103(1): pp.588-601, Jan 1998.

[8] P.M.Martins, J.S. Ferreira. PCM to MIDI Transposition, Audio Engineering Society 112th convention, Munich, Germany, May 2002