A Simple Music Transcription Method Using Comb Filter Techniques and Envelope Detection

Felipe Espic

University of Miami, Coral Gables, FL, 13343, USA

f.espic@umiami.edu

ABSTRACT

A polyphonic music transcription system is proposed, which is able to transcript music created with harmonic instruments into a MIDI file. Polyphonic pitch estimation is based on an iterative routine that chooses the most repeated estimated fundamental frequencies. Short Time Fourier Transform, Phase Vocoder and Comb Filtering based techniques are used to estimate fundamental frequencies. A method for onset detection based on envelope following is presented.

1. INTRODUCTION

Music transcription is one of the highly topical subjects for music technology scientists today. Resynthesizing music, either timbral features or rhythm structure is not a trivial task. There are several methods to address this task such as genetic algorithms [15], non-negative matrix factorization [3], generic templates [16], non-negative sparse coding [17], probabilistic models [18], harmonicity and spectral smoothness [8], and so on, which yield quite good results. The method proposed in this paper is broken up into two dependent sub-tasks; polyphonic pitch and music event detection.

There are many approaches for extracting the pitch information, such as Auditory Model-Based [4], Autocorrelation [7], Cepstrum Analysis [14], Comb Filtering [10], [11], [13], and so on. One of the most used techniques to estimate the signal pitch is comb filtering. This has several advantages; it involves low computational complexity avoiding high CPU load, which allows real time applications. On the contrary, many techniques based on Auditory Models or Independent Component Analysis are quite computationally complex. Moreover, this method yields accurate results obtained by using high sample rates or fractional delay techniques. Also, it updates its output every one sample, since it is applied in time domain, so that it presents excellent time resolution.

There are many techniques based on comb filtering to obtain multi-pitch data. The first approach was presented by Moorer who used an FIR filter bank [9]. Then, Miwa used this method for music transcription [20], and later Gainza modified the method using IIR filtering [11].

On the other hand, onset detection has been addressed with several methods including spectral difference [1], resonator time frequency image [2], auditory analysis [4], and linear prediction analysis [5]. An efficient and simple method is based on the *Envelope Follower* [6], [19], which shows robust performance. The method proposed uses an IIR-based envelope follower intended to be economic computationally.

The assumed constrains of the proposed system are:

- Up to three pitch frequencies simultaneously (triads).
- Intended for being used with synthesized harmonic instruments.
- Maximum interval between notes: one octave.

2. BACKGROUND

2.1. Polyphonic pitch detection using comb filtering

Comb Filters (CF) and Resonator Comb Filters (RCF) are able to select the harmonics of a specified fundamental frequency F0. Some typical representations of these filters are respectively:

$$H_{comb(z)} = 1 - \beta z^{-T} \quad (1)$$

$$H_{res\_comb(z)} = \frac{1-\propto}{1-\propto z^{-T}} \quad (2)$$

Where β is the magnitude scaling factor, T the time delay in samples, and $\propto$ the quality factor parameter. The latter must be less or equal to 1 to ensure

stability in the case of the resonant comb filter. Some examples of both filter types are shown in the Fig. 1. The resonances or nulls are located at frequencies:

$$f_n = n\frac{fs}{T}, \qquad n = 0,1,2... \tag{3}$$

Where *fs* is the sampling rate.

Then, for harmonics sounds, CF and RCF modify the signal severely when *F0* matches$\frac{fs}{T}$, since the resonances/nulls will coincide with the harmonics of the sound. In other words, there is a specific *T* with which the filter affects the signal the most.

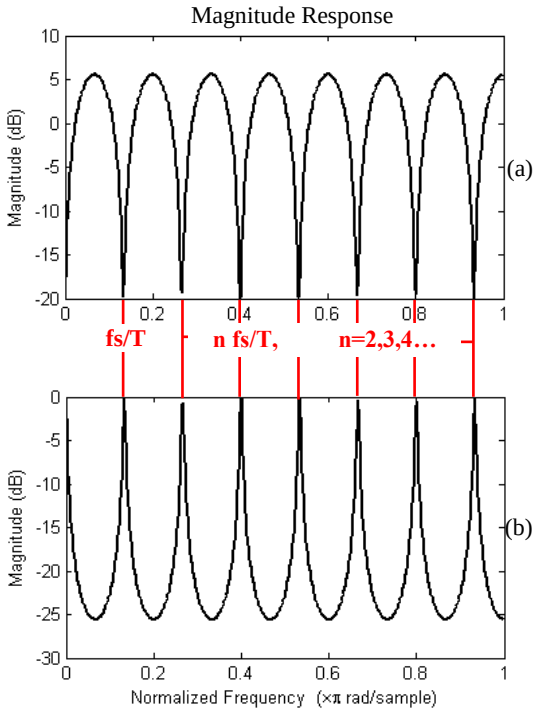

Fig.1. Examples of the magnitude of comb filters with *T*=15. (a) Comb filter with β=0.9. (b) Resonator comb filter with $\propto$=0.9.

Thus for polyphonic pitch detection, the methods proposed in [10], [11] and [13] are based on the idea of applying 12 comb filters in parallel, whose time delay are adjusted to match the frequencies of the 12 semitones of a selected octave. Then, the filters that affect the signal the most should correspond to the musical notes.

2.2. Frequency estimator using phase vocoder

Phase vocoder is a really versatile tool that allows performing many different processes, such as frequency estimation, pitch shifting and time stretching [12].

For frequency estimation, Fourier Transform gives us information about the spectral magnitude (energy per bin) and the spectral phase (phase per bin). Then, for instance, we could pick the frequency of the bin that has the highest energy. The center frequency *fc* of the bin *k* is given by:

$$fc = k\,\frac{fs}{N}, \quad k = 0,1,2,\dots,(N-1) \tag{4}$$

Assuming that there is only one sinusoidal partial within each bin; a good estimation of the actual frequency of that partial corresponds to the center frequency *fc*. However, as you can see in the expression above, the accuracy of the estimation is determined by the rate of *fs* and *N* in such way that the error estimation is less than $\frac{fs}{2N}$. Then, for example, a typical analysis with *N*=1024 and *fs*= 44.1 kHz yields an error estimation less than 21.5Hz, which is not enough for the purpose of this paper.

Phase vocoder allows improving the computation of the frequency estimation performed by other methods, such as Fourier Transform or a filter bank. Thus, the frequency estimate for the bin *k* (or channel in the case of filter bank) is given by [12]:

$$fe(k) = \frac{(angle(k,t_2) - angle(k,t_1) + 2\pi p)}{2\pi hop} \tag{5}$$

Where *angle* is the phase in the *k*th bin in time *t*; t_1 is the time of the first analysis and t_2 for the second analysis, *hop* is the time difference between two consecutive analyses $(t_2 - t_1)$; p is an integer that should be determined to unwrap the phase to obtain the correct value of *fe*(k). If p is not determined correctly, the value of *fe*(k) will be completely wrong. One constraint for this approach is that *hop* size must be less than a certain value that depends on the windows type [12]. For example, for Hanning and Hamming windows $hop \leq N/4$ and for Kaiser $hop \leq N/6$. If this constraint is met, p could take only one plausible value within the interval delimited by the *k*th bin. Thus, the computation of p is easy and safe.

2.3. Onset detection using the Envelope Follower method

The envelope follower (EF) method is summarized in the diagram shown in Fig. 2 [6]. A wide band signal or the channels from a filter bank feed the EF. It is known that using a filter bank at the input of the process yields better results, since the analysis is made with better frequency resolution.

The first step is a full-wave rectifier, which takes the input signal and makes it positive for every value, as shown in Fig. 4. Then, an FIR filter is applied to the signal by the convolver. The coeficients of the filter are nothing but the second half of a Hanning window function of 200ms length. This is to emulate the human auditory temporal response [19]. The output of this stage is actually the envelope of the signal.

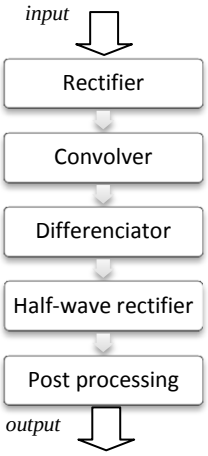

Figure 2 The Envelope follower method diagram.

Later, the output of the differentiator could be understood as the value of the slopes for each imaginary straight line that connects two consecutive envelope values. It means that the faster the envelope changes, the higher is the differentiator output value. Therefore, the highest values will correspond to the sound onsets. This signal is called the '*difference of envelopes*'.

The post processing step depends on the objective of the analysis. For instance, if the goal is tempo tracking, firstly a half-wave rectify is applied to the difference of envelopes and then a periodicity estimation analysis is [19]. On the other hand, for onset detection, a peak picking method is applied [6].

3. PROPOSED METHOD

The proposed method is based on the concepts and processes described in the Background section. The algorithm should be able to create a MIDI file from a musical audio file. The overall diagram is shown in Fig. 3.

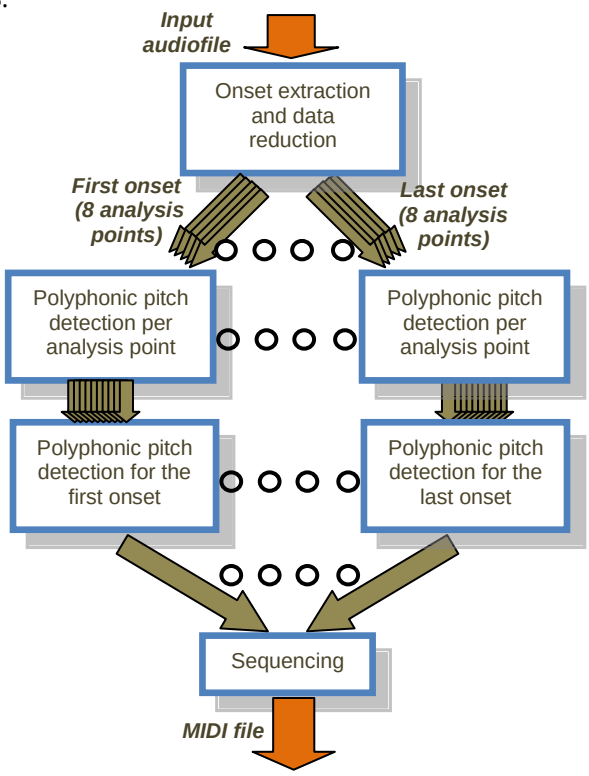

Figure 3 General diagram of the proposed method.

Each one of the steps of the algorithm is described in the following sections.

3.1. Onset detection and data reduction

3.1.1. Decimation

The application of decimation is intended to decrease computational complexity of the process. The decimation factor depends on the sample rate, since the Nyquist frequency has to be above 3kHz to accept frequencies up to the octave number 7. Decimation is optional and it also means a trade of with the FFT length for further analyses (see next subsection). One example of good performance is carried out by using fs=44.1kHz, decimation factor = 2 and *FFT* length = 8192 samples.

3.1.2. Onset Detection

The decimated input signal is processed by using a modified version of the method described in subsection 2.3:

Envelope Extraction

This new envelope extraction differs from the previously described, because this uses a two sections 2nd order peaking IIR filter instead of the 2nd half-Hanning window FIR filter explained on section 2.3, which is very expensive computationally. Each section is described by:

$$H_{env(z)} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 - a_1 z^{-1} - a_2 z^{-2}} \qquad (6)$$

Where $b_0 = 0.00041981$, $b_1 = 0$, $b_2 = -0.00041981$, $a_1 = 1.99916034$ and $a_2 = -0.99916036$. Figure 4 shows an example of the steps of the onset extraction algorithm. Plot (c) shows the envelope of the signal.

Difference of Envelopes Cleaning

After obtaining the *difference of envelopes* function (Fig. 4 (d)), this is cleaned using a threshold value to obtain only positive values at around onsets peaks. The signal values lower than the threshold will be set as zeros.

Binary Difference of Envelopes Cleaning

Finally, a binary signal is produced whose 1's correspond to areas around the onsets, and zero values set the most 'safe' areas to compute STFT's, which are performed by the pitch extraction process. It ensures that within the same safe area, the STFT's mostly analyze the same musical event, avoiding false pitch estimations. Moreover, it avoids the white-like spectral characteristics of the attack of the notes.

3.1.3. Data reduction

Polyphonic pitch analyses could be applied for every sample within the specified safe areas, which involves high computational cost. Then, for each event 8 equally spaced points in time are set. Thus, polyphonic pitch detection is performed only at these analysis points.

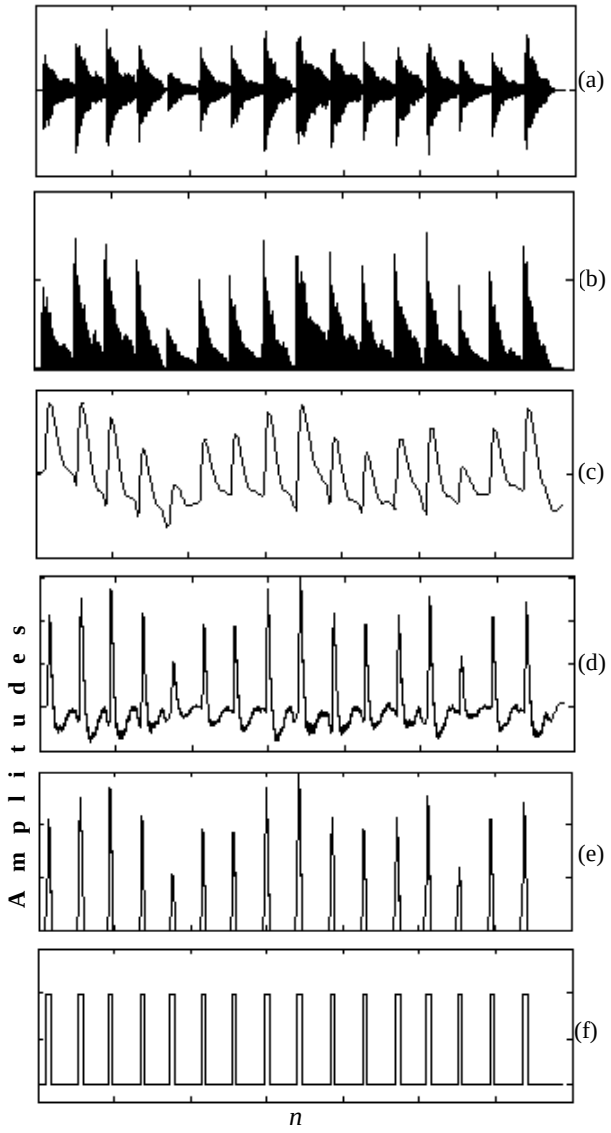

Figure 4 Time domain signals within the onset detection process. (a) Original input audio signal. (b) Full-wave rectified. (c) Envelope. (d) Difference of envelopes. (e) Cleaned difference of envelopes. (f) Binary difference of envelopes. The actual values of amplitudes and *n* are not considered, since they are not relevant for illustration purposes.

3.2. Polyphonic pitch detection

The next block in the general diagram (Fig. 3) is primarily applied for each analysis point turned out by the data reduction algorithm. This does not follow the typical approach of using 12 comb filters described in subsection 2.1 due to pitch of the instruments could be not perfected tuned to the expected Chroma frequencies. This is really an important factor, since the bandwidths of the comb filters must be quite narrow to avoid

affecting adjacent frequencies. It means, that even a slightly out of tune could make the algorithm fail.

3.2.1. Pitch detection

For this purpose, a STFT is applied at each analysis point, whose locations coincide to the center points of the analysis windows. It ensures a correct estimation of the spectrum around the points.

It is assumed that the musical instruments generate sounds that present their fundamental frequency *F0*. Then, according to the magnitude spectrum, the maximum is picked and passed throw a phase vocoder in order to make a more accurate estimate of its frequency. This is the first pitch candidate.

3.2.2. Comb filtering

The CF was selected as main pitch estimator engine. This is applied making its nulls to match the location of harmonics of the *F0* estimate. If this is well estimated, the comb filter will nearly erase all the energy of the corresponding musical note (Fig 5).

3.2.3. Iteration

The above algorithm produces two outputs: an *F0* estimate and the corresponding audio signal that lacks of the guessed musical note energy. Then, iteration is necessary to estimate the other simultaneous musical notes; the output signal is fed back into the pitch detection analysis, in order to estimate all the present musical notes. Sometimes, comb filters do not remove all the energy of a musical note, so the signal is filtered more than once by the same filter. Finally, if the energy of the input signal is lower than a specified threshold, the algorithm stops. This analysis made for one analysis point, turns out a list of pitch frequency candidates for the whole musical event.

3.3. Selection of Candidates

As mentioned, each musical event is analyzed at 8 equally spaced points in time. Thus, for one musical event, we obtain 8 lists of pitch candidates. The 3 most repeated ones are the final candidates. Finally, the candidate that is present on 4 of the 8 lists is chosen as one of the definitive estimate pitch frequencies.

3.4. Sequencing

After obtaining the pitch frequencies of each musical event, a MIDI file is generated with NOTE ON messages located at onsets time locations. NOTE OFF messages are triggered just before a new onset, since the proposed method does not involve a *release* detection process.

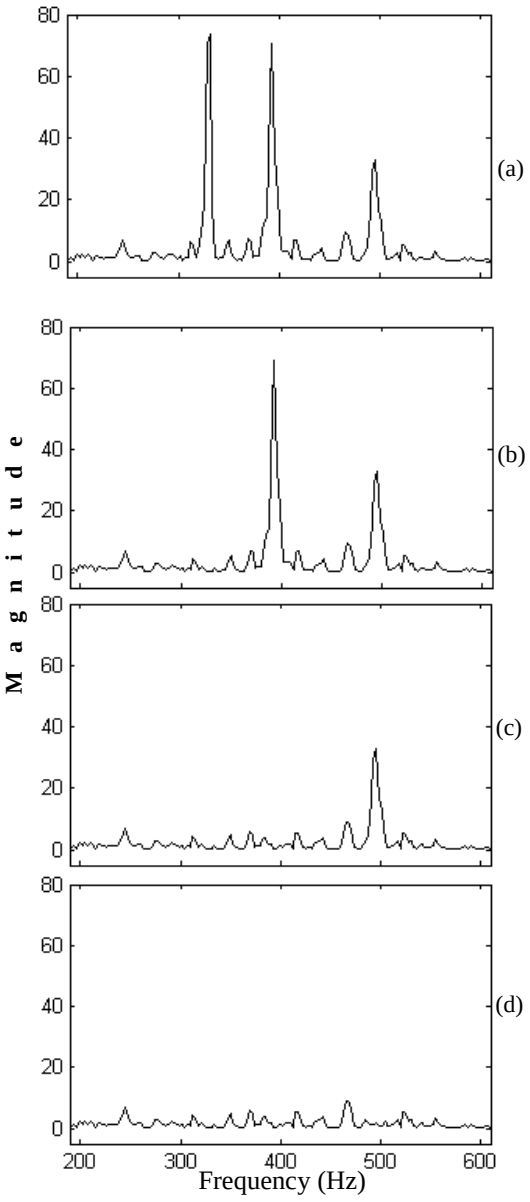

Figure 5 spectral decomposition of the chord EM4 carried out by the looping process. (a) Original signal. (b)Signal after iteration 1. (c) Signal after iteration 2. (d)Signal after iteration 3.

4. RESULTS AND CONCLUSIONS

The algorithm has been tested with several types of audio files yielding that is quite reliable and robust for the basic types of signals for what was designed. However, it is necessary to adjust the *target octave*, and the *repetition threshold* in the case of more complex music. The test signals include piano sequence chords, piano melodies, piano interval sequences, synthetic sound chord sequences. Despite the results, the system has some drawbacks: it works properly for one preselected octave, harmonic instruments and for a maximum of three notes simultaneously.

Features to be implemented on next realizations are NOTE OFF detection, support for more than triads and intervals of notes greater than one octave.

5. REFERENCES

[1] Bor-Shen Lin; Hsin-Jung Huang; , "Reliable onset detection scheme for singing voices based on enhanced difference filtering and combined features," *Wireless Communications & Signal Processing*, 2009. WCSP 2009. International Conference on , vol., no., pp.1-5, 13-15 Nov. 2009

[2] Ruohua Zhou; Mattavelli, M.; Zoia, G.; , "Music Onset Detection Based on Resonator Time Frequency Image," *Audio, Speech, and Language Processing, IEEE Transactions on* , vol.16, no.8, pp.1685-1695, Nov. 2008

[3] Smaragdis, P.; Brown, J.C., "Non-negative matrix factorization for polyphonic music transcription," Applications *of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on.* , vol., no., pp.177,180, 19-22 Oct. 2003

[4] Benetos, E.; Stylianou, Y. , "Auditory Spectrum-Based Pitched Instrument Onset Detection," Audio, *Speech, and Language Processing, IEEE Transactions on* , vol.18, no.8, pp.1968-1977, Nov. 2010

[5] Lee, W.-C.; Kuo, C.-C.J.; , "Musical Onset Detection Based on Adaptive Linear Prediction," *Multimedia and Expo, 2006 IEEE International Conference on* , vol., no., pp.957-960, 9-12 July 2006

[6] Bello, J.P.; Daudet, L.; Abdallah, S.; Duxbury, C.; Davies, M.; Sandler, M.B.; , "A Tutorial on Onset Detection in Music Signals," *Speech and Audio Processing, IEEE Transactions on* , vol.13, no.5, pp. 1035- 1047, Sept. 2005

[7] Hat Yai, Songkhla; , "Pitch Detection Algorithm: Autocorrelation Method And AMDF", Department of Computer Engineering, Faculty of Engineering, Prince of Songkhla University, Thailand, 90112

[8] Klapuri, A.P., "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness," *Speech and Audio Processing, IEEE Transactions on* , vol.11, no.6, pp.804,816, Nov. 2003

[9] Moorer, J.A., "The optimum comb method of pitch period analysis of continuous digitized speech," *Acoustics, Speech and Signal Processing, IEEE Transactions on* , vol.22, no.5, pp.330,338, Oct 1974

[10] Tadokoro, Y., Morita, T. and Yamaguchi, M. (2003); , "Pitch detection of musical sounds noticing minimum output of parallel connected comb filters", *Conference on Convergent Technologies for Asia-Pacific Region (TENCON)*, Bangalore, India

[11] Gainza, Mikel and Lawlor, Robert and Coyle, Eugene, ; "Multi pitch estimation by using IIR comb filters," 47th. *International Symposium focused on Multimedia Systems and Applications (ELMAR)*, Zadar, 2005

[12] Puckette, Miller, ; "Phase-locked vocoder," *IEEE ASSP Workshop on app. of sig. proc. to audio and acous.*, 1995

[13] Tadokoro, Y. and Yamaguchi, M, ; "Pitch estimation of polyphony based on controlling delays of comb filters for transcription," *Proc. 11th DSP Workshop*, pp.371-375, Taos Ski Valley, New Mexico, USA, Aug. 2004

[14] Noll, A.M.; "Short—Time Spectrum and Cepstrum Techniques for Vocal—Pitch Detection, " *JASA*, Vol. 36, 296—302. 1969

[15] Reis, G.; Fonseca, N.; Fernandez, F.; Ferreira, A.; , "A Genetic Algorithm Approach with Harmonic Structure Evolution for Polyphonic Music Transcription," *Signal Processing and Information Technology, 2008. ISSPIT 2008. IEEE*

International Symposium on , vol., no., pp.491-496, 16-19 Dec. 2008

[16] Faruqe, M.O.; Hasan, M.A.-M.; Ahmad, S.; Bhuiyan, F.H.; , "Template music transcription for different types of musical instruments," *Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference on*, vol.5, no., pp.737-742, 26-28 Feb. 2010

[17] Abdallah, S., and Plumbley, M.; , "Polyphonic music transcription by non-negative sparse coding of power spectra," in *Proc. 5th Int. Conf.Music Inf. Retrieval (ISMIR'04),* Barcelona, Spain, Oct. 10–14, 2004

[18] Mauch, M., and Dixon, S., ; "Approximate Note Transcription for the Improved Identification of Difficult Chords," to appear in the *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR 2010),* 2010

[19] Scheirer, E.,"Tempo and Beat Analysis of Acoustic Musical Signals," *JASA*, 103, 1998

[20] Miwa, T., Y. Tadakoro, and T. Saito, *The Problems of Transcription using Comb Filters for Musical Instrument Sounds and Their Solutions*. 2000. Technical Report of IEICE.