PEAK TRACKING AND PARTIAL FORMATION OF MUSIC SIGNALS

Hamid Satar-Boroujeni and Bahram Shafai

ECE Department, Northeastern University
Boston, MA 02115, USA
phone: + 1(617) 373-2984, fax: + 1(617) 373-4189
email: {hsattar & shafai}@ece.neu.edu

ABSTRACT

In this paper we provide a method for detection of peaks in spectral representations of music signals for the purpose of partial tracking. The basic idea is to detect local maxima in any digitized signal and use statistical techniques for rejecting spurious peaks. Detected peaks are then connected to each other to form partial tracks. The performance of our algorithm is investigated in two levels. First the output of peak detection algorithm is compared with another method. Second, these two sets of peaks are fed into our partial tracking system and the results are compared, which confirms the superiority of our strategy. This superiority is in consistency of our method in detecting the valid tracks and preventing spurious peaks from forming false tracks.

1. INTRODUCTION

In signal processing domain detection of peaks in spectral representations can serve as a front-end step in tracking periodicities and partials within audio signals. This is frequently used in research areas such as music analysis/synthesis [1], audio restoration [2], automatic music transcription [3], and speech analysis [4].

There exist a great number of methods for detection of peaks in different areas of signal processing domain, many of which are developed based on the characteristic of analyzed signal. A common approach includes defining a threshold level, solid or adjustable, and collecting maxima beyond the threshold ([3], [5]). In another approach, which was used for physiological signal, a mathematical framework for behaviour of picks is developed, and peaks (as well as troughs) in temporal signals are detected without using any threshold level [6].

This paper will proceed with a brief discussion on the existing peak detection strategies. In section 3 we present detailed discussion on our proposed algorithm. This is followed in section 4 by a discussion on estimation of the parameters introduced. The performance of our technique is examined and compared with others in section 5. Since our intention for peak detection is tracking of partials in music signals, the quality of detected peaks is examined by feeding them into our partial tracking system. This is also done for detected peaks using method of [3]. The resulting partial tracks are compared at the end.

2. BACKGROUND

Two main properties of a good peak detection algorithm are inclusion of all the genuine peaks and exclusion of all those peaks related to noise or imperfections in estimating the spectrum. As it is shown in figure 1, having only the numbered peaks as the valuable ones, the solid threshold techniques fails to include #4 and #5 while capturing three spurious peaks. Adjustable threshold, which adjusts itself to the overall shape of spectrum, works well by detecting four peaks, but still #4 is missing. The mathematical framework detects as many as nineteen peaks which are too many but comprise all the real peaks.

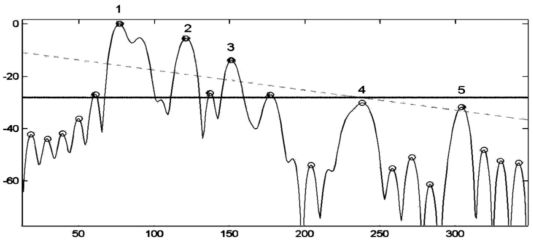

Figure 1: Three peak detection techniques: solid threshold (solid line), adjustable threshold (dashed line), and mathematical framework (circles)

Our proposed algorithm consists of two steps which are shown in figure 2**.** In the first step which implements the property of inclusion, we use the mathematical framework to collect all the peaks that fit into the very definition of a peak as a local maximum. These are referred to as raw peaks. The implementation of the exclusion part is through the process of using statistical properties of a relative number of data points surrounding each raw peak to examine the concreteness of detected peak and rule out any incompetent maxima.

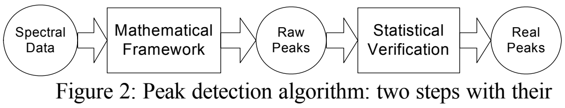

Figure 2: Peak detection algorithm: two steps with their resulting peaks

3. DETECTION ALGORITHM

3.1 Spectral Estimation

We used Correlogram Spectral Estimation (CSE) technique [7] for estimating the spectrum of our music signal. Here the only assumption is that the signal under study is stationary. For this, we slice our signal into frames small enough to be considered as stationary. Since music signal can be approximated as stationary for about 25 milliseconds, we selected frames of that length using a hamming window.

If we have N samples $\{y(1), y(2), \ldots, y(N)\}$, then

$$\hat{\phi}_c(\omega) = \sum_{k=-(N-1)}^{N-1} \hat{r}(k) e^{-j\omega k} \qquad (1)$$

where $\hat{\phi}_c(\omega)$ is the estimated power spectral density and $\hat{r}(k)$ denotes an estimate of the covariance lag which is obtained through

$$\hat{r}(k) = \frac{1}{N-k} \sum_{t=k+1}^{N} y(t) y^*(t-k) \qquad (2)$$

To gain a better frequency resolution we used a highly zero-padded CSE with 32768 points. After estimating the spectrum using (1) and (2), we took it into logarithmic scale.

3.2 Mathematical Framework

Here a peak is defined as a maximal element that locally dominates its surrounding points in the magnitude spectrum with some positive, predefined threshold. Using this basic definition we try to collect all peaks without missing any peak that can be considered as a real peak in the next step.

If we consider our spectral data as the sequence $\{p_1, p_2, \cdots, p_N\}$, then the i$^{\text{th}}$ point p_i is a peak if [6]

$$\begin{cases} p_i \geq p_j + \delta \text{ and } p_j \leq p_l \leq p_i \\ \quad \text{for } l = j+1, j+2, \ldots, i-1 \\ p_i \geq p_k + \delta \text{ and } p_k \leq p_m \leq p_i \\ \quad \text{for } m = i+1, i+2, \ldots, k-1 \end{cases}, 1 \leq j < i < k \leq N \qquad (3)$$

where δ is the pre-defined threshold and N is the number of spectral points. As it can be seen, the first and last points can not be considered as peaks since for verifying a peak we need at least one predecessor and one successor elements.

If two or more successive points with equal value satisfy the requirements, then we can consider the last point as a peak and discard all the preceding ones. This is a rather arbitrary decision, but to be more precise, we considered the mid-point between the first and the last element.

After a point-to-point scan on our spectral data, the indices of all the qualified peaks are collected and stored to be processed through the next step.

3.3 Statistical Analysis and Verification

The first requirement is to set our threshold δ in (3) to a value that guarantees capturing all the valuable peaks into the set of raw peaks. The second requirement is exclusiveness in terms of spurious peaks which is approached statistically.

Our best guide for evaluating the quality of a peak is its highness among a relative number of its surrounding data points. Here peak p_i is defined to be high enough if

$$p_i \geq m_{[p_i]} + d\sigma_{[p_i]} \qquad (4)$$

where $m_{[p_i]}$ and $\sigma_{[p_i]}$ refer to the mean and standard deviation of the elements surrounding p_i respectively, and d is a tuning parameter. We use mean and standard deviation to make sure that all the spurious peaks which appear as side peaks or simply as small variations as high as δ are excluded and parameter d gives us one degree of freedom for tuning purposes.

For verifying a peak we must consider as many surrounding points related to that peak as possible. The number of neighbouring points is related to the index of p_i or the frequency of the peak. This can be shown as follows. The fundamental frequency of standard notes in Western music notation, also called pitch, is given by

$$f_k = 440 * 2^{\frac{k}{12}} \; [Hz], \quad k \in Z \qquad (5)$$

in which *k=0* refers to so-called chamber note (A_4) and *k* ranges from -48 to 39 for a standard piano. For a note with fundamental frequency at f_k we must consider all the neighbouring data points whose frequencies lie between $f_{k-1} + \frac{1}{2}(f_k - f_{k-1})$ and $f_{k+1} - \frac{1}{2}(f_{k+1} - f_k)$. This requirement is to ensure that in the case where three succeeding notes are present, we only consider those points associated mostly with the peak at f_k. So, the range of neighbouring points with respect to f_k is

$$R = \frac{1}{2f_k}\left(440 \times 2^{\frac{k+1}{12}} - 440 \times 2^{\frac{k-1}{12}}\right) = 0.0578 \qquad (6)$$

The above range is calculated for the fundamental frequency f_k. However, since other harmonics of each note are integer-multiples of f_k, we can apply this range factor to any point in the spectrum.

Something to be noted here is that although the distance between harmonics of the same level grows as we move to higher frequencies, they might interfere with harmonics of different notes from different levels. Theoretically, this can happen when

$$\left| n_1 f_{k_1} - n_2 f_{k_2} \right| < R f_{k_1} \qquad (7)$$

where n_1 and n_2 are harmonic numbers of note with fundamental frequency f_{k_1} and f_{k_2} respectively. This will result in

$$\frac{n_1(1-R)}{n_2} < \frac{f_{k_2}}{f_{k_1}} < \frac{n_1}{n_2} \; OR \; \frac{n_1}{n_2} < \frac{f_{k_2}}{f_{k_1}} < \frac{n_1(1+R)}{n_2} \qquad (8)$$

Either one of this inequalities can lead to (7). Since these are probable to happen both in theory and real world, some valuable peaks may be rejected if they get close to each other. However, the algorithm proved to work distinctively better than the case with a constant number of surrounding points [3].

4. PARAMETER ESTIMATION

Throughout our evaluations for accuracy of our algorithm, the detection accuracy turned out to be most sensitive to the value of thresholding factor *d*. At first, this factor was set heuristically but after further investigations it turned out to be frequency-dependant. This motivated us to study its sensitivity to frequency and estimate its value for different frequencies by analyzing a large database of music sounds with known identities.

For the purpose of this study, we conducted a statistical analysis on 57 notes played individually on different woodwind instruments. The aim of this study was to find a first and a second candidate for every harmonic of each note and find bounds of the thresholding factor in such a way that the first candidate is picked up and the second one is ignored, i.e.

$$T_2(t,f) < m_{[p_i]}(t,f) + d\sigma_{[p_i]}(t,f) < T_1(t,f) \qquad (9)$$

in which $T_1(t,f)$ and $T_2(t,f)$ are tracks formed by the first and second candidates respectively. These are functions of both time and frequency since they are formed by connecting similar peaks within the same frequency bin and through successive time frames. Using (9), our factor is bounded by

$$\frac{T_2(t,f) - m_{[p_i]}(t,f)}{\sigma_{[p_i]}(t,f)} < d < \frac{T_1(t,f) - m_{[p_i]}(t,f)}{\sigma_{[p_i]}(t,f)} \qquad (10)$$

The process of finding tracks of first and second candidates was done as follows. The waveform of each individual note was windowed into frames of 25 milliseconds length, and the spectrum for each frame was estimated using method of section 3.1. Each spectrum was partitioned into 40 bins of length equal to the known fundamental frequency and centred at the harmonics of the note. Within each frequency bin, the maximum value and all those peaks that were less than 5 dB away from the maximum were collected and all peaks below -80 dB were rejected.

After finding all such peaks for all frequency bins and all time frames, tracks of first and second candidates were formed by searching within the same frequency bin and through successive frames based on the relevance of peaks to the expected harmonic in that bin. This process was started with the peak with maximum power in the same bin of all time frames and continued in forward and backward directions. Starting with this global maximum if the following frame contained more that one peak, the one which was closer in frequency to, and less than $0.0578 f_k$ away from the selected peak was considered as the continuation of the track for the first candidates. The maximum value among all remaining peaks in each frequency bin was added to the track of second candidates. If there were no qualified peak for either of the tracks in any time frame, the track was terminated, and a new track was initiated in the following frame by choosing a peak with maximum power.

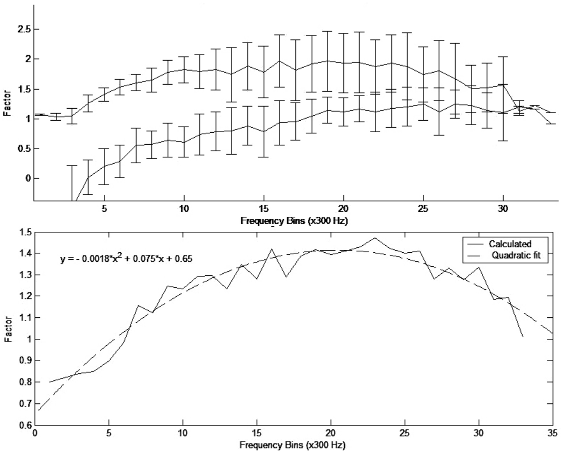

Figure 3: Upper and lower bounds of thresholding factor (up), and its best value along with the best quadratic fit (down)

After all the tracks had been acquired (a total number of 916 tracks) they were organized into 80 different groups based on their average frequency. Since the sampling frequency is 48000 Hz, then the i^{th} group contains tracks with average frequency between 300(i-1) and 300i. For each track and its corresponding second candidate the upper and lower bounds for the thresholding factor were calculated using (10). The upper and lower bound values for each track were averaged to yield a single value for each track. The mean and standard deviation of upper and lower bound of all the tracks within each group were then calculated and plotted, which is shown in figure 3. It should be noted that only the 33 first groups contained more than one track and result is shown for up to 9900 Hz.

We can also attain a single graph as a representative for the best values of the factor for different frequencies, which is shown in figure 3.

5. RESULTS

5.1 Peak Detection

We tested our algorithm using artificial and real data, and detection result for a short B3 played on the oboe is shown in figure 4. Although in this example peaks are hardly recognizable after 6.8 kHz, 37 out of 40 possible peaks are detected and only three peaks are missed, two of which are almost buried in the noise surface of higher frequencies. Here, the temporal signal was sampled at 48000 points per second and our correlogram spectral estimator used 70000 points which gives rise to a reasonably detailed representation of about 0.7 point per Hz. The threshold for the raw peak detection step was set to 10 dB for which in all tests performed on artificial and real data we were able to include all the real peaks, and it produced the least computational load for the next step.

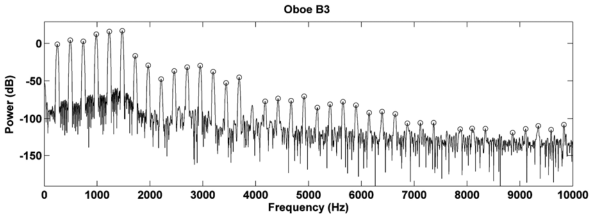

Figure 4: Detected peaks (circled) for a B3 note played on the Oboe

5.2 Detection Comparison

For the purpose of comparing our algorithm with existing ones, we define the accuracy rate as follows

$$R_a = \frac{n_d - 0.5n_f}{n_e} \times 100 \quad (11)$$

in which n_d is the number of detected peaks, n_f is the number of false peaks and n_e is the number of expected peaks. Since the effect of false peaks can be reduced during the later process of track formation in our developing project of partial tracking, we degraded its role in computing the accuracy rate of our algorithm. Other useful factors are

$$R_d = \frac{n_d}{n_e} \times 100, \qquad R_f = \frac{n_f}{n_e} \times 100 \quad (12)$$

Here, R_d is the detection rate and R_f is the false rate. Table 1 contains a comparison between our algorithm and the one introduced in [3]. This result is computed for a range of 32 notes and the averaged percentages are shown here. It should be noted that unlike our strategy, the adjustable threshold in [3] is highly dependent upon the overall shape of the spectrum. Hence, for our comparison, it was tuned to yield the best results.

	R_d	R_f	R_a
Our Method	98.3	23.4	86.6
Method of [3]	80.4	35.2	62.8

Table 1: Accuracy comparison for peak detection

5.3 Tracking Comparison

Since we consider discrete frames of temporal data in estimation of spectrum, for tracking the evolution of frequency in time and creating partial tracks we need to make connection between peaks from adjacent time frames using data association techniques. We use Kalman filtering to track frequency and power of partials. The state-space evolution model for the Kalman tracker is introduced in [8]. We fed the detected peaks in both methods from the previous step into the Kalman tracker and the results were compared against each other. The factors used for this comparison are

$$R_{dt} = \frac{n_{dt}}{n_{et}} \times 100, \qquad R_{ft} = \frac{n_{ft}}{n_{et}} \times 100 \quad (13)$$

in which, R_{dt} is the rate of detected tracks and R_{ft} is the rate of false tracks. n_{et} is the number of expected tracks, and n_{dt} and n_{ft} are the number of detected tracks and false tracks respectively. Table 2 contains the comparison results.

As we can see, the rate of detected tracks is very close to the rate of detected peaks for our method, but this is not the case for [3]. This means that, compared to [3], our algorithm is more consistent in detecting peaks pertaining to the same tracks in adjacent time frames and more number of detected peaks take part in formation of correct tracks. On the other hand, false peaks in our system are less likely to form false partial tracks.

	R_{dt}	R_{ft}
Our Method	98.2	18.2
Method of [3]	70.6	34.8

Table 2: Accuracy comparison after partial tracking

6. CONCLUSION

In this paper we presented a peak detection approach with a novel strategy in which we first collect all possible peaks and then examine their validity by using prior knowledge from music signals. This showed to give more accurate results than where a pre-thresholding is used. Since these peaks are used for partial tracking, we tested our result through our partial tracking system, in which the accuracy of our detection algorithm was confirmed.

REFERENCES

[1] X. Serra, "Musical Sound Modeling with Sinusoids plus Noise" Musical Signal Processing, Studies on New Music Research. Swets & Zeitlinger, Lisse, the Netherlands, pp. 91–122, 1997.

[2] S. J. Godsill, and P. J. W. Rayner, *Digital audio restoration: a statistical model based approach*. London; New York: Springer, 1998.

[3] A. Sterian, "Model-Based Segmentation of Time-Frequency Images for Musical Transcription," Ph.D. Dissertation. Ann Arbor: University of Michigan, 1999.

[4] R. J. McAulay and T. F. Quatieri, "Speech Analysis/ Synthesis Based on a Sinusoidal Representation," *IEEE ICASSP*, vol. 34, no. 4, pp. 744–754, 1986.

[5] S. W. Hainsworth and P. J. Wolfe. "Time-frequency reassignment for musical analysis," In *Proceedings of the International Computer Music Conference*, pp 14-17, 2001.

[6] B. S. Todd and D. C. Andrews, "The Identification of Peaks in Physiological Signals," *Computers and Biomedical Research*, vol. 32, pp. 322-335, 1999.

[7] P. Stoica and R. L. Moses*, Introduction to spectral analysis*, Upper Saddle River, N.J.: Prentice Hall, 1997.

[8] H. Satar-Boroujeni and B. Shafai, "State-Space Modeling and Analysis for Partial Tracking of Music Signals," presented at the 24th IASTED International Conference on Modelling, Identification, and Control, Innsbruck, Austria, 2005.