PEAK EXTRACTION AND PARTIAL TRACKING OF MUSIC SIGNALS USING KALMAN FILTERING

Hamid Satar-Boroujeni
Northeastern University
Department of Electrical and Computer Engineering

Bahram Shafai
Northeastern University
Department of Electrical and Computer Engineering

ABSTRACT

In this paper we propose a partial tracking method for music signals based on Kalman filtering. We first introduce a novel technique for detection of peaks in spectral representations of music signals. We also introduce different evolution models for our Kalman filter based on the shape of frequency and power partials in different classes of melodic instruments. Parameters of these models are estimated using a large database of music signals. We analyze the performance of our tracker through a comparison with another method and also by observing its effectiveness in the presence of crossing partials and vibrato. The problem of missing peaks and the contribution of a backward tracker are also discussed.

1. INTRODUCTION

Tracking of partials plays an important role in the areas of music signal analysis where the focus is on estimating pseudo-stationary properties of these signals such as frequency and amplitude [1]-[2]. These properties can be extracted from spectral representations of these signals. However, this can be done only for discrete frames of the temporal data as small as it can be assumed to be stationary. Peaks or local maxima within these frames are indications of partials and for reconstructing the temporal evolution of these partials, data association techniques are used.

There exist various methodologies for tracking of partials, all of which are based on a model of time-varying sinusoidal component plus noise [1]. Partial tracking was first used for analysis and synthesis of music signal by [1], where it was based on a heuristic approach. In a more recent approach [3], linear prediction was used to enhance the tracking of frequency components in music signals. In these approaches peaks from successive frames are connected to each other based on their proximity in frequency, and the behaviour of peaks' amplitude is not taken into account, while performing the tracking. Another approach [2] takes the advantage of Kalman filter by constructing a state-space model for behaviour of peaks' power (i.e. amplitude in dB scale) and frequency. In this approach peaks are not matched based on how close they look like in frequency, rather they are matched based on the future behaviour of the frequency and power of a peak.

This paper will proceed with our peak detection method in section 2. In section 3 we discuss the problem of modelling and introduce our set of evolution models needed for Kalman filter. The formulation of our Kalman tracker and its properties is in section 4, and results of tracking as well as a comparison with the method of [2] will follow in section 5.

2. PEAK DETECTION

Detection of peaks in the spectral representation is an important task. Shortcoming of peak detection strategy in collecting as many valuable peaks as possible and rejecting spurious peaks, can result in discontinuity in partials or formation of false partial tracks. This motivated us to look for an efficient peak tracking strategy for music signals which is detailed in [4].

Our proposed algorithm consists of two steps which are shown in figure 1. In the first step we use a 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. In the next step we use 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.

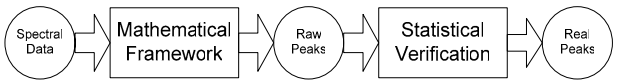

Figure 1. Peak detection algorithm: two steps with their resulting peaks

3. MODELING

3.1. Sum-of-Sinusoidal Model

A well-known approach to modelling of music signals for the purpose of statistical analysis/synthesis assumes a model of additive sinusoidal plus residuals that can be formulated as [1]

$$y(t) = s(t) + e(t) \tag{1}$$

$$s(t) = \sum_{n=1}^{N} A_n(t) \cos\left(\omega_n(t) + \phi_n(t)\right) \tag{2}$$

Here, $s(t)$ reflects the pure musical part of the signal and $e(t)$ can be modelled as a stationary autoregressive process. In the musical portion, $A_n(t)$ and $\omega_n(t)$ are representatives of time-varying amplitude and frequency of partials, and N is the number of partials. Quantity

$\phi_n(t)$ can represent timbral variations and performance effects. It should be noted that timbral variations are also scattered among the amplitude changes, the instantaneous frequency changes and also in the power and spectral envelope changes of $e(t)$. However, these are all treated as noise processes in our modelling.

3.2. Partial Evolution Model

The next step in our music signal modeling is estimation of $A_n(t)$ and $\omega_n(t)$ using available observations from the peak detection step. What we have is discrete sets of peaks from successive time frames. $A_n(t)$ and $\omega_n(t)$ can be estimated by making connections between those peaks from adjacent frames that look like being the continuation of the same partial.

Kalman filtering, in fact, takes the noisy observations and based on a model for evolution of certain states finds the optimal estimate of the process behaviour. Here the noise corrupted observations are the identified peaks and system model is a state-space model for evolution of frequency and power. This model can be represented as

$$\begin{aligned} x(k+1) &= Ax(k) + Bv(k) \\ y(k) &= Cx(k) + w(k) \end{aligned} \tag{3}$$

where

$$\begin{aligned} x(k) &= \left[f(k) \quad p(k) \quad n_1(k) \quad \cdots \quad n_m(k) \right]^T \\ v(k) &= \left[u_1(k) \quad \cdots \quad u_m(k) \right]^T \\ y(k) &= \left[f(k) \quad p(k) \right] \end{aligned} \tag{4}$$

Here, $f(k)$ and $p(k)$ are frequency and power for a detected peak respectively. $v(k)$ and $w(k)$ are process noise and observation noise, and $n_i(k), i = 1, ..., m$ are states for as many shaping filters for which the uncorrelated noise processes $u_i(k), i = 1, ..., m$ are white. The matrix A is the transition matrix, the matrix B describes coupling of the process noise $v(k)$ into the system states, and C is the observation matrix. In this model, $v(k)$ and $w(k)$ are zero-mean and jointly uncorrelated Gaussian processes with covariance matrices Q and R, respectively.

3.3. Instrument-Specific Models

We need prior information about power and frequency partials to specify our model by a piecewise-linear fit to $p(t) = 20 \log A_n(t)$ and $f(t) = \omega(t) / 2\pi$.

Melodic instruments can be classified into two groups based on the way their source for sound production behaves. If during production of sound, source continues to inject energy, the overall shape of the steady-state part of the amplitude track will be non-decaying. We consider these instruments in the class of Continued Energy Injection (CEI). Examples of this group of instruments are woodwind, brass, and violins from the family of string instruments. An example of the fundamental of chamber note played on a clarinet is shown in the upper part of figure 2.

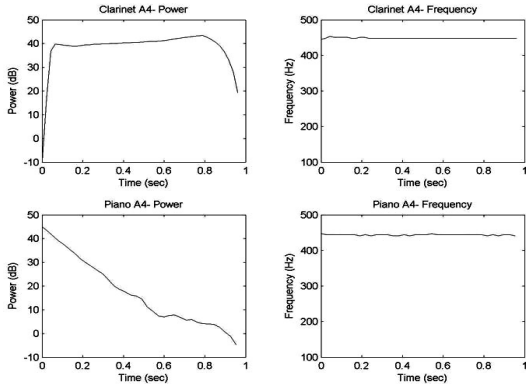

Figure 2. Shape of power and frequency partials for the fundamental of a chamber note played on the clarinet and the piano.

The second group includes those instruments for which the injection of energy is discontinued and amplitude partial represents an exponentially decaying shape. In this case, the power partial will have a linear decay since it is in logarithmic scale. These instruments are considered in the class of Discontinued Energy Injection (DEI). Members of this group are hammered and plucked instruments such as piano and guitar. One example of the shape of fundamental frequency for the chamber note played on the piano is shown in the lower part of figure 2.

In a polyphonic setting, there are three possible scenarios when we do not consider non-melodic instruments such as percussions. A piece of music can consist of instruments solely form the CEI, or only from DEI, or a combination of both.

For the first scenario, where both frequency and power are nearly constant we have

$$\begin{aligned} f(k+1) &= f(k) + n_1(k) \\ n_1(k+1) &= a_1 n_1(k) + b_1 u_1(k) \\ p(k+1) &= p(k) + n_2(k) \\ n_2(k+1) &= a_2 n_2(k) + b_2 u_2(k) \\ x(k) &= \left[f(k) \quad p(k) \quad n_1(k) \quad n_2(k) \right]^T \\ v(k) &= \left[u_1(k) \quad u_2(k) \right]^T \\ y(k) &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} x(k) + w(k) \end{aligned} \tag{5}$$

For the second and third scenario we have the same model, but their parameters are different since we estimate these parameters by considering different databases of sound for each scenario. This model can be shown as

$$
\begin{aligned}
&f(k+1) = f(k) + n_1(k) \\
&n_1(k+1) = a_1 n_1(k) + b_1 u_1(k) \\
&p(k+1) = p(k) + v_p(k) \\
&v_p(k+1) = v_p(k) + n_2(k) \\
&n_2(k+1) = a_2 n_2(k) + b_2 u_2(k) \\
&x(k) = \begin{bmatrix} f(k) & p(k) & v_p(k) & n_1(k) & n_2(k) \end{bmatrix}^T \\
&v(k) = \begin{bmatrix} u_1(k) & u_2(k) \end{bmatrix}^T \\
&y(k) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} x(k) + w(k)
\end{aligned}
\tag{6}
$$

As noted above, we estimate the parameters of each model, e. g. a_1, b_1, a_2, b_2, by performing a statistical analysis on a large number of musical sounds with known identities in a forward-problem setting. The details of this procedure are presented in [5].

4. PARTIAL TRACKING

Here, we process discrete segments of music signal, and after extracting useful information related to the behavior of partials, we intend to put this information together and represent the shape of these partials in time. The extracted information or noisy observations are frequency and power of peaks from different time frames and we use Kalman filter to estimate noise free outputs and from there, find the set of frequency and power data in the adjacent frame that are most related to them.

4.1. Kalman Tracker

Our Kalman tracker is initiated with peak data from the first time frame. Depending on the nature of our music signal and the class of instrument, and based on the frequency of the initial peak, a class of models is selected as the evolution model. Kalman tracker then estimates the noise-free values for power and frequency in the following frame. If the following frame contains a peak that is close enough to the estimated values, that peak is added to the track and is used to update the tracker. This process is continued through successive frames until there is no peak close enough to the last estimated peak. Here, the track is terminated or considered as *dead*, and a new track is initiated in the following frame. The process starts with all peaks in the first frame and also with all peaks from other frames that have not been used in any track.

4.2. Acceptance Gate

After estimating noise-free values for frequency and power of a peak in the i^{th} track at the time frame k, we compare them to all existing peaks in the k^{th} frame. We then update our tracker with the peaks that are close enough to these estimations, or in another term, fall into the acceptance gate of the tracker. We define a distance function as a function of both frequency and power of the estimated value for a track. A peak falls into the acceptance gate of an estimated peak if the value of its distance function is less that the gate value. If more than one peak fall into the acceptance gate, the one with less distance is selected.

4.3. Crossing Partials

Although power and frequency partials evolve independently, considering a function of both power and frequency for the distance function is especially rewarding when we are dealing with crossing partials. In partial tracking techniques, where power and frequency partials are tracked separately (e.g. in[1] and [3]), the problem of crossing partials needs considerable attention and requires additional adjustments to the original tracker. However, in our algorithm the contribution of constant and distinct frequency partials in the distance function helps the tracker to distinguish between the corresponding power partials in the crossing area, and it does not need additional adjustments.

4.4. Missing Peaks

Due to imperfections in estimating the spectrum and also because partials with low power can get buried in noise, we might face the problem of missing peaks. This can result in discontinuities in parts of a partial. To overcome this problem, it is proposed in [1] to add "zombie" states to the end of a track where we cannot find any peak within the acceptance gate. In our algorithm we update a track with estimated states in such situation, and continue this process for a maximum of three frames. If during these attempts no peak falls into the acceptance gate, we consider that track as dead and extract the fake updates from the track. If we find a peak during this process, the track is updated with this peak and we keep the fake updates.

4.5. Backward Tracking

To add to the accuracy of our algorithm we can perform a backward tracking at the end of each track. When a track is terminated, we can initiate a backward tracker with the last updated states and error covariance matrix. This process is identical to the forward tracking but in the reverse direction. This can be helpful because the forward tracker is loosely initiated with the noisy observations for power and frequency and zero values for other states, while our backward tracker is initiated more accurately. On the other hand, the backward tracker is capable of recovering discontinuity in the forward tracking results, since it has the support of a more accurate initiation and a longer history of observation updates.

5. RESULTS

We examined the accuracy of our algorithm by performing the proposed partial tracking on a wide range of instrumental sounds from different classes of melodic instruments. The tracking results were compared with that of the method proposed in [2]. This comparison was done by first defining some accuracy factors. These factors are

$$R_{dt} = \frac{n_{dt}}{n_{et}} \times 100, \qquad R_{ft} = \frac{n_{ft}}{n_{et}} \times 100 \tag{7}$$

where R_{dt} is the detection rate, R_{ft} is the false rate, n_{dt} is the number of detected tracks, n_{ft} is the number of false tracks, and n_{et} is the number of expected tracks. We computed these factors for 32 musical notes from all classes of melodic instruments. The same sets of peaks from our peak detection process were fed into the tracker of [2]. Table 1 contains accuracy factors for these two trackers.

	R_{dt}	R_{ft}
Our Method	98.2	18.2
Method of [2]	84.7	27.4

Table 1. Accuracy rates

To test the performance of our tracking system in the presence of crossing partials, we used fictitious sound signals containing two music notes; one with constant and the other with linearly decaying power partials. Tracking result for these crossing power partials is shown in figure 3.

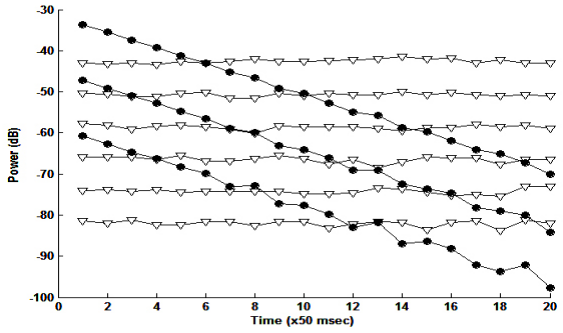

Figure 3. Crossing power partials

We also tested our tracker in the presence of vibrato. Tracking result for five frequency partials of a signal with vibrato is presented in figure 4. In addition, performance of the backward tracker is shown in figure 5. The forward track is discontinued from frame 27 to 31, but our backward tracker, which is initiated with estimated states at the end of forward track (frame 55), is able to recover missing point of the partial.

6. CONCLUSION

In this paper we proposed improved techniques for detection of peaks in spectral representations of music signals. We proposed instruments-specific models for evolution of partials, which was used in our Kalman tracker. We also investigated the performance of our tracker in critical situations.

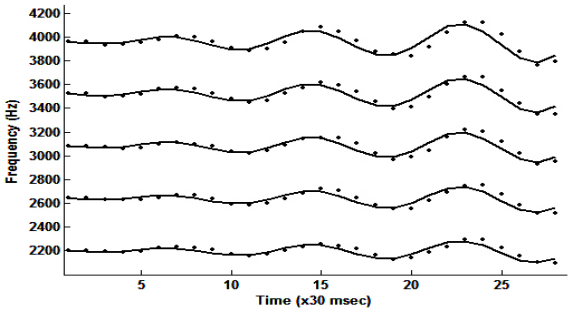

Figure 4. Tracking results for frequency partials containing vibrato (solid) along with estimated values (dots).

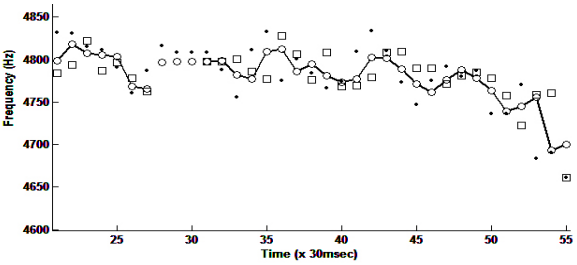

Figure 5. Forward and backward tracking: discontinued partial in the forward tracking (solid line) and its estimate (squares), along with the backward track (circles) and its estimates (dots).

7. REFERENCES

[1] X. Serra, “Musical Sound Modelling with Sinusoids plus Noise,” in *Musical Signal Processing*, Exton, PA: Swets & Zeitlinger, 1997, pp. 91-122.

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

[3] M. Lagrange, S. Marchand, and J. B. Rault, "Using Linear Prediction to Enhance the Tracking of Partials," *Proc. of IEEE-ICASSP '04*, Toronto, Canada, 2004.

[4] H. Satar-Boroujeni and B. Shafai, "An Adaptive Peak Detector with Parameter Estimation for Music Signals," submitted to IEEE-WASPAA, New Paltz, NY, 2005

[5] H. Satar-Boroujeni and B. Shafai, "State-Space Modeling and Analysis for Partial Tracking of Music Signals," *Proc. of the 24th IASTED-MIC*, Innsbruck, Austria, 2005.