TEMPO MODIFICATION OF MUSIC SIGNAL USING SINUSOIDAL MODEL AND LPC-BASED RESIDUE MODEL

Akinori Ito, Yuki Igarashi
Graduate School of Engineering, Tohoku University

Masashi Ito
Faculty of Engineering, Tohoku Institute of Technology

Takashi Nose
Graduate School of Engineering, Tohoku University
aito@spcom.ecei.tohoku.ac.jp

Changing tempo of the music signal is one of the most basic signal processing applied to music signals. Traditional algorithms such as phase vocoder and PSOLA uniformly stretch and shrink the input signal. Therefore, those methods change not only the tempo but also the structure of the signal of the instrumental sound, such as attack and decay time, which changes the timbre of the instruments. To change the tempo of the music signal while keeping the Attack-Decay-Sustain-Release structure of the instruments, we need a non-linear modification of the time scale. To realize this, we propose a two-stage modeling of the music signal. The first stage models the music signal using the sinusoidal model that expresses the harmonic part of the signal using sum of sinusoids with temporally-variable amplitude and frequency. Because non-harmonic component of the signal cannot be modeled using the sinusoidal model, the residue of the sinusoidal model is analyzed using the linear-prediction coding (LPC) in the second stage, which expresses the reverberation of the impulsive sound. Then the residue of the LPC analysis is stretched or shrunk non-linearly according to the short-term power, where only the parts with small power are modified because the parts with larger power correspond to the attack parts. Finally, the modified residue is used to synthesis the modified signal using the LPC synthesis filter and sinusoidal synthesizer. We conducted experiments to modify the tempo of piano sounds, and compared the result with the conventional time stretch methods.

1. Introduction

Various sound signal analysis methods have been developed so far. The sinusoidal modeling is one of those methods. Sinusoidal modeling approximates the sound signal as the sum of sinusoidal components with time-varying amplitude and frequency [1]. The sinusoidal model using the local vector transform (LVT) [2] is one of the methods of sinusoidal modeling, which can express the signal with time-varying frequency and amplitude. The LVT is proved to have high accuracy when analyzing signals with quickly-changing pitch. As the sinusoidal model expresses an audio signal using a relatively small number of sinusoidal components, manipulations of the signal such as pitch shifting and speed change are relatively easy. Conventional sinusoidal analysis has been applied to
signals that have only one sound source, such as speech or monophonic instrumental sound. Here, we have studied a possibility to apply the sinusoidal model to polyphonic music signals \[3\], and obtained a result that the sinusoidal model can model the signal even when the signal contains two or more sound sources as long as the signal does not contain any percussive or noise-like sound.

In this paper, we developed a method to model the sinusoidal part and noise-like part separately, and the noise-like part is further modeled using the linear prediction analysis. Using the proposed method, we can express the signal with noise-like or percussive sound very accurately. Then we propose a method to manipulate the length of the signal without changing the pitch of the original signal, keeping the structure of the instrumental sound.

2. Sinusoidal modeling and LVT

The sinusoidal model approximates the speech and audio signal using an arbitrary number of sinusoidal components, which have time-varying amplitude and phase [1]:

\[
x(t) = \sum_{k=1}^{K} A_k(t) \cos(P_k(t))
\]  

where \(A_k(t)\) and \(P_k(t)\) are instantaneous amplitude and phase, respectively. The parameters \(A_k(t)\) and \(P_k(t)\) are firstly estimated from the spectrum obtained by the short-time Fourier transform (STFT) by picking the spectral peaks, and then chosen considering the temporal contiguity of the sinusoidal components.

Using this sinusoidal modeling, several applications such as speech synthesis and modification have been studied [4, 5]. However, a problem was pointed out that the estimation accuracy of the non-stationary part of the speech was not sufficient [6, 7, 8]. This problem is caused by the basic assumption of the algorithm, which assumes that the frequency and amplitude within a frame are stationary. However, in reality, the frequency and amplitude of the input signal vary continuously with time.

To overcome this problem, Ito and Yano proposed the sinusoidal model based on the local vector transform (LVT) [2], which assumes the sinusoidal components with time-varying amplitude and frequency within a frame. The LVT can model the voiced speech very accurately [9].

In the sinusoidal modeling based on the LVT, the input signal \(x(t)\) is approximated as Eq. (2) when \(t\) is nearly zero.

\[
x(t) = \sum_{k=1}^{K} \exp(a_k(t)) \cos(p_k(t))
\]  

Here, \(a_k(t)\) is the logarithmic instantaneous amplitude and \(p_k(t)\) is the instantaneous phase of the \(k\)-th sinusoidal component. Then we consider the function \(a_k(t)\),

\[
a_k(t) = a_k(t) + j p_k(t)
\]  

where \(j = \sqrt{-1}\), so that we can express the original signal as

\[
x(t) = \sum_{k=1}^{K} x_k(t) = \sum_{k=1}^{K} \Re[\exp(a_k(t))].
\]  

Then we approximate the \(a_k(t)\) as a quadratic function of \(t\), as

\[
a_k(t) = a_{k,0} + a_{k,1} t + a_{k,2} t^2 / 2.
\]
Because the above approximation holds when \( t \approx 0 \), we estimate \( \alpha_k(t) \) within a short frame. The LVT method estimates the parameters from the spectrum calculated by short-term Fourier transform (STFT). Then the entire signal is expressed as an overlap-add of the windowed sinusoidal components.

\[
x_k(t) = \sum_{n=1}^{N} \Re \{ w_\Delta(t - n\Delta) \exp(\alpha_k(n)(t - n\Delta)) \}
\]  

(6)

Here, \( N \) is the number of frames, \( \Delta \) is the frame shift and \( 0 \leq w_\Delta(t) \leq 1 \) is a window function that has width \( 2\Delta \) and satisfies

\[
w_\Delta(t) + w_\Delta(t - \Delta) = 1 \quad (0 \leq t \leq \Delta)
\]  

(7)

(such as the Hanning window). \( \alpha_k^{(n)}(t) \) is the \( k \)-th sinusoidal component of the \( n \)-th frame.

To calculate Eq (6), we need to choose an appropriate sinusoidal component for each frame that is a part of \( k \)-th sinusoidal component \( x_k(t) \), because the sinusoidal components of the frames are estimated independently. Considering that the same sinusoidal components of two contiguous frames have similar log amplitude and log frequency, we connect the sinusoidal components of contiguous frames using dynamic programming [15].

Let \( \alpha_0(n,m), \alpha_1(n,m) \) and \( \alpha_2(n,m) \) be the parameters having \( m \)-th lowest frequency at frame \( n \) (which does not necessarily correspond to the \( m \)-th sinusoidal component of the total signal). Then we determine indices of the sinusoidal components of all frames

\[
M_k = (m_1, m_2, \ldots, m_N)
\]  

(8)

that minimizes the following cost function \( \psi \).

\[
\psi(M_k) = \sum_{n=2}^{N} D(n, m_{n-1}, m_n)^2
\]  

(9)

\[
D(n, l, m) = \Im \left[ \alpha_1(n - 1, l) - \alpha_1(n, m) + \frac{\alpha_2(n - 1, l) - \alpha_2(n, m)}{2\Delta} \right]
\]  

(10)

This problem can be solved using dynamic programming using the following recurrence relation.

\[
\psi(n, m) = \min_l [\psi(n - 1, l) + D(n, l, m)^2]
\]  

(11)

Then

\[
\min \psi(M_k) = \psi(N, k).
\]  

(12)

\[
\tilde{M}_k = (\tilde{m}_1, \ldots, \tilde{m}_N) = \arg \min_{M_k} \psi(M_k)
\]  

(13)

The indices \( \tilde{M}_k \) can be decided by tracing back the minimization step (11). After determining the indices, we define

\[
\alpha_{k,i}^{(n)} \equiv \alpha_i(n, \tilde{m}_n).
\]  

(14)

### 3. Modeling the residue signal

To model the music signal better, Serra and Smith proposed an analysis method to split the music signal into the noise-like part and the harmonic part [11]. The harmonic part is represented using the sinusoidal model, and the noise-like part is treated as a stochastic signal. The stochastic part can be obtained by taking residue of the sinusoidal analysis. Ding and Qian [12] proposed a similar model, where the residue signal was modeled by the multi-pulse linear prediction (MPLP). Objective
In this paper, we employ a residue model similar to Ding and Qian for modeling polyphonic music signal. We used the linear predictive coding (LPC) to model the residue signal [3]. LPC-based modeling is advantageous for signal manipulation such as time stretch or compression.

Figure 1 shows an example of residue signal of the sinusoidal analysis and the reconstructed signal via LPC analysis/synthesis. The signal-to-noise ratio of the reconstructed signal with respect to the sinusoidal residue was 50.8 dB, which is high enough for further signal processing.

4. Tempo modification

4.1 Overview

Tempo modification of the audio signal is one of the basic problems of audio signal processing. Numerous methods have been proposed such as phase vocoder [16, 17] or the time domain harmonic scaling (TDHS) [18]. These methods basically stretch or compress the audio signal uniformly. However, sounds of a musical instrument with different length are not obtained by a simple signal stretch and compression. Figure 2 shows (a) an example of a signal of MIDI piano sound, (b) the sound of the same instrument with double value (twice as long as the previous note) and (c) the signal stretched by the TDHS. We can see that (a) and (b) have similar structure in the attack part, but (c) does not preserve that structure. If we want to manipulate a music sound as if the same music is played with a different tempo, we need to employ more sophisticated method for signal stretch/compression.

In the proposed framework, the input signal is divided into the sinusoidal part and the residual part, and then the residual part is analyzed by LPC to generate the LPC coefficients and the LPC residue signal. Then sinusoidal components and the LPC residue signal are stretched or compressed independently. Finally, the modified signal is generated by the LPC synthesis and the sinusoidal

![Figure 1](image1.png)

**Figure 1.** An example of residue of the sinusoidal analysis and the reconstructed signal by the LPC analysis/synthesis.

![Figure 2](image2.png)

**Figure 2.** The signal of the piano and the stretched signal.
Figure 3. A block diagram of time stretch/compress using LPC-based residue modeling

Figure 4. A block diagram of stretching/compressing LPC residue

synthesis. Figure 3 shows a block diagram of time stretch/compression using sinusoidal modeling and residue modeling using LPC.

4.2 Modification of residue signal

When stretching/compressing the residue signal, it is our goal to realize the method to keep the structure of the sound, where the attack and decay parts are kept unchanged, while the sustain and release parts are stretched or compressed. To this end, we need to decide which part can be stretched or compressed and which part cannot. In this work, we consider that the peaks of the LPC residue signal are related to the attack part. Therefore, we searched samples of the LPC residue that have small absolute value. Then values of those samples are substituted with zero, and only the zero-valued samples are stretched or compressed. Figure 4 shows the overview of the residue stretch and compression.

Here, we need to determine the “stretchable part” in Figure 4 to avoid modifying the attack and decay part. This should be performed automatically; however, in the following experiment, we manually specified the stretchable part. We need to develop a method to determine them automatically (using a method such as beat tracking) in the future work.

4.3 Modification of the sinusoidal part

Tempo modification is basically done by stretching or compressing the length of a frame. When the relative speed is $\beta > 0$, the length of the frame changes from $\Delta$ to $\Delta/\beta$. Here, we use the instantaneous phase $\tilde{p}_k(n)(t)$ instead of $p_k^{(n)}(t) = \Im[a_k^{(n)}(t)]$, as follows.

$$\tilde{p}_k(n)(t) = C_k^{(n)} + p_k^{(n)}(t) + \beta p_k^{(n)}(t)^2/2$$ (15)
When modifying the frame length, we need to adjust phases of the overlapping frames so that the instantaneous phases of two contiguous frames are similar at the boundary of the frames [14]. So we set the initial phase $C_k^{(n)}$ so that the next equation holds.

$$\tilde{p}_k^{(n)}(0) = \tilde{p}_k^{(n)}(-\Delta/\beta)$$

(16)

Therefore,

$$C_k^{(n)} = C_k^{(n-1)} + p_k^{(n)} \Delta \beta - p_k^{(n)} \frac{\Delta^2}{2\beta}$$

(17)

And finally, we obtain the tempo-changed signal by overlapping the stretched or compressed frames as follows,

$$\tilde{x}_k(t) = \sum_{n=1}^{N} \Re \left[w_{\Delta/\beta}(t - n\Delta/\beta) \exp(\tilde{a}_k^{(n)}(t - n\Delta/\beta))\right]$$

(18)

where

$$\tilde{a}_k(t) = a_k(t) + j\tilde{p}_k(t).$$

(19)

When actually performing signal stretch and compression, we need a nonlinear stretch/compression just as that of the LPC residue. Therefore, in the following experiment, we modified tempo of the “stretchable part” of the signal so that the total length of the signal becomes the desired length.

5. Experiment

We conducted an experiment to evaluate the tempo change using the proposed method. Frame length for sinusoidal analysis was 5 ms. We used SPTK [13] for the LPC analysis. The window length, frame shift and order of the LPC analysis was 80 ms, 40 ms and 40, respectively, which were determined according to a preliminary experiment. We used the Hanning window as the window function.

We stretched the MIDI piano signal for $\beta = 0.5$ using the proposed method and TDHS. Figure 5 shows the waveforms of the results. We can see that the proposed method preserves the structure of the attack and decay part compared with the TDHS. We measured the signal-to-noise ratio between the MIDI piano signal of the double-valued note (Figure 5 (b)) and the stretched signal. As a result, SNR of the proposed method was 11.0 dB, while that of the TDHS was $-2.6$ dB. This result proved that our method outperformed the conventional method.

6. Conclusion

In this paper, we proposed a method for stretching/compressing audio signals based on sinusoidal model and residue model. First, we model the harmonic part of the input signal using the sinusoidal model, and the residue is modeled using the LPC analysis. Then the sinusoidal part and
the residue are stretched or compressed independently, and finally they are combined to generate the manipulated signal. The evaluation experiment proved that the proposed method could achieve signal manipulation that keeps the structure of the instrumental sound. The remaining problem is to determine the “stretchable part” automatically; that could be done by detecting the beat of the input signal [19] and setting the inter-beat part as “stretchable.” Another future work is to evaluate the manipulation of more complicated signal such as music signal with multiple instruments.

Acknowledgment

Part of this work was supported by JSPS A3 foresight program “Ultra-realistic acoustic interactive communication on next-generation Internet.”

REFERENCES


