IMPROVEMENT OF AN ONLINE MUSIC ALIGNMENT BASED ON ONSET INFORMATION

Bochen Li
University of Rochester
Dept. of Electrical and Computer Engineering
bli23@ur.rochester.edu

ABSTRACT

Music alignment is the association of events in a musical score with points in the time frames of an audio signal. So it is also called audio-score alignment. Until now, most polyphonic audio-score alignment methods are offline algorithms, and alignment accuracy drops significantly when using online algorithms. But most applications based on alignment need real-time results, such as Automatically Accompaniment System or Music Tutor. So we aim at improving the alignment result of an online alignment algorithm.

In general online methods, a polyphonic music audio is segmented into time frames and they are fed to the score follower in sequence. Then the algorithm outputs a score position for each frame right after it is processed. In this paper, we find that not all audio frames give us the same level of reliability within the evolution of one note, especially in staccato performing or decay of piano sound when real sound will last shorter compared with our expectation from the score. Then the online follower in previous algorithm will get lost.

So in this paper, we propose to utilize note onsets information to find the "faithful" frames as observation to do the alignment. A weighting function is introduced to assign different weights to the frames as the measure of their reliability. The function value is dropping within evolution of one note and score position tends to move forward evenly excluding observations from audio, while when a new onset is detected and weight returns high, it will be dragged to right position quickly due to high trust of observation. Experiments show that proposed improvements can get better alignment results than previous online methods, especially in piano music.

1. INTRODUCTION

Generally, there are primarily two kinds of music data: sampled audio files, such as those found on compact discs, and symbolic music representations, which essentially specify notes with pitch, duration, tempo and so on. MIDI is widely used as symbolic music representations for computers since it concludes all the information on a sheet score with similar size as a text. So we can say MIDI is the score for computers. For human, there are thousands of different performances of one music piece based on the performers' styles, and musician can always find the position in score while music playing. To draw an analogy, music alignment is aim to give computer such an ability to find the position in MIDI from an audio recording. It is related to the problem of synchronization between performers and computers

There are two different versions of the problem, usually called "offline" and "online". Offline alignment uses the complete performance to estimate the correspondence between audio data and symbolic score. Thus, the offline problem allows one to "look into the future" while establishing the correspondence. Offline algorithms can be used to synchronize multiple modalities (video, audio, score, etc.) of music to build a digital library. But the application is limited.

Online alignment, sometimes called Score Following, process audio data in real-time as the signal is acquired, thus no "look ahead" is possible. The goal of score following is to identify the musical events described in the score with high accuracy and low latency. While offline algorithms can only be used in offline applications, online algorithms can be used in both offline and online scenarios, and even be made to work in real time if they are fast enough. For example, one can generate a flexible musical accompaniment that follows a live soloist. Or one can devise a music tutor system for beginning leaners. Other applications include real-time score-based audio enhancement e.g. pitch correction, and automatic page turners. [1]

Most existing polyphonic audio-score alignment methods use Dynamic Time Warping (DTW) [2]–[4], a HMM [5], [6], a hybrid graphical model [7] or a conditional random field model [8]. Although these techniques achieve good results, they are offline algorithms, that is, they need the whole audio performance to do the alignment.

Dannenberg [9] and Vercoe [10] propose the first two real-time score followers, but both work for MIDI performance instead of audio. There are some real-time or online audio-score alignment methods [11]–[14]. However, these methods are for monophonic (one note at a time) audio performances. Two of these systems ([13], [14]) also require training of the system on prior performances of each specific piece before alignment can be performed for a new performance of the piece. This limits the applicability of these approaches to pieces with preexisting aligned audio performances.

Cont [15] proposes a hierarchical HMM approach to follow piano performances, where the observation likelihood is calculated by comparing the pitches at the hypothesized score position and pitches transcribed by Nonnegative Matrix Factorization (NMF) with fixed

spectral bases. A spectral basis is learned for each pitch of the specific piano beforehand. This method might have difficulties in generalizing to multi-instrument polyphonic audio, as the timbre variation and tuning issues involved make it difficult to learn a general basis for each pitch.

[16] addressed the online audio-score alignment problem with a continuous state HMM process to model the audio performance, which allows an arbitrary precision of the alignment. And an observation model with multi-pitch information provides a more accurate connection between audio and score than traditional representations such as chroma features. This score follower performs better than other current online algorithms for most multi-instrument polyphonic music, but will get lost when the evolution of notes is not steady (decay of piano notes, staccato notes, echo, etc.).

In this paper, we will use the same model structure in [16] and utilize onset information to make the alignment more accurate and robust. We find that for most of music pieces, offset time of performed notes does not strictly follow the score. They are often earlier than their expected time. The reliability decreases when time goes further from onsets, due to the uncertainty of offset time. Since audio onset time usually strictly follows the score, the frames right after onsets are reliable. So we propose to impose a weighting function on audio frames to find out reliable audio frames by onset detection. More reliable frames will have stronger influence through the observation model so we will get a faithful observation model. Furthermore, the onset information can improve the alignment accuracy. Only using the multi-pitch information, the first audio frame in the sustain part of a note can be aligned to any position of the corresponding note with no difference. With onsets detected, we can require the first frame to be always aligned to the onset of a score note.

The remainder of this paper is arranged as follows: Section 2 reviews the ideas in [16], which is also used in this paper. Section 3 describes how to utilize onset information to improve the alignment. Experimental results are presented in Section 4 and the paper is concluded in Section 5.

2. MODEL STRUCTURE

In this paper we use the same model structure in [16]. A process model represents the score position and tempo. This model describes how the states transition and makes the position only move forward based on the position and tempo in last state. Then an observation model evaluates how likely the current audio frame is to contain the pitches at a hypothesized score position. The inference of the score position and tempo of the current frame is achieved by particle filtering. It is a way to do the inference in an online fashion.

Our model structure is illustrated in Figure 1. We process the audio frames in sequence. The *n*-th frame is associated with a 2-dimensional hidden state vector $\mathbf{s}_n = (x_n, v_n)^T$, where x_n is its score position (in beats), v_n is its tempo (in Beat Per Minute). Each audio frame is also associated with an observation, which is a vector of PCM encoded audio, $\mathbf{y}_n$. Our aim is to infer the current score position x_n from current and previous observations $\mathbf{y}_1, \cdots, \mathbf{y}_n$.

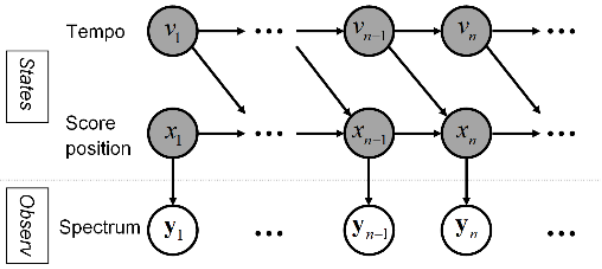

Figure 1. Illustration of the state space model for online audio-score alignment

2.1 Process Model

A process model defines the transition probability from the previous state to the current state, i.e. $p(\mathbf{s}_n \mid \mathbf{s}_{n-1})$. We use two dynamic equations to define this transition. To update the score position, we use

$$x_n = x_{n-1} + l \cdot v_{n-1} \tag{1}$$

where l is the audio frame hop size. Thus, score position of the current audio frame is determined by the score position of the previous frame and the current tempo. To update the tempo, we use

$$v_n = \begin{cases} v_{n-1} + n_v & \text{if } z_k \in [x_{n-1}, x_n] \text{ for some k} \\ v_{n-1} & \text{otherwise} \end{cases} \tag{2}$$

where $n_v \sim N(0, \sigma_v^2)$ is a Gaussian noise variable; z_k is the *k*-th note onset time in the score. This equation states that if the current score position has just passed a note onset, then the tempo makes a random walk around the previous tempo according to a Gaussian distribution; otherwise the tempo remains the same. Different from [17], we only consider the onset time and exclude the offset time when we segment the score into states, since offset parts will effect little due to the decay of weighting function. (It will be illustrated in Section 2.3.)

2.2 Observation Model

The observation model is to evaluate whether a hypothesized state can explain the observation, i.e. $p(\mathbf{y}_n \mid \mathbf{s}_n)$. Different representations of the audio frame can be used. Here we still use the multi-pitch analysis information in [16], since it is the most informative one to evaluate the hypothesized score position for most fully-scored musical works. So we use the multi-pitch observation likelihood as our observation model.

The multi-pitch observation model is adapted from multi-pitch estimation in [17]. It is a maximum likelihood-based method which finds the set of pitches that

maximizes the likelihood of the power spectrum. In [17], each significant peak of the power spectrum is detected and represented as a frequency-amplitude pair (f_i, a_i). Non-peak regions of the power spectrum are also extracted. The likelihood of the power spectrum given a set of hypothesized pitches θ is defined in the peak region and non-peak region respectively.

$$\mathcal{L}(\boldsymbol{\theta}) = \mathcal{L}_{\text{peak region}}(\boldsymbol{\theta}) \cdot \mathcal{L}_{\text{non-peak region}}(\boldsymbol{\theta}) \quad (3)$$

More details of this approach are described in [16] and [17].

2.3 Inference

Given the process model and the observation model, we want to infer the state of the current frame from current and past observations. From a Bayesian point of view, this means we first estimate the posterior probability $p(\mathbf{s}_n \mid \mathbf{Y}_{1:n})$, then decide its value using some criterion like maximum a posterior (MAP) or minimum mean square error (MMSE). Here, $\mathbf{Y}_{1:n} = (\mathbf{y}_1, \cdots, \mathbf{y}_n)$ is a matrix whose each column denotes the observation in one frame. By Bayes' rule, we have Equation (4).

$$p(s_n \mid \mathrm{Y}_{1:n}) = \frac{p(\mathrm{Y}_{1:n} \mid s_n) \cdot p(s_n)}{p(\mathrm{Y}_{1:n})} \quad (4)$$
$$= C_n \cdot p(y_n \mid s_n) \cdot \int p(s_{n-1} \mid \mathrm{Y}_{1:n-1}) \cdot p(s_n \mid s_{n-1}) ds_{n-1}$$

Where $\mathbf{y}_n$, $\mathbf{Y}_{1:n}$, $\mathbf{s}_n$ and $\mathbf{s}_{n-1}$ are all random variables; $\mathbf{s}_{n-1}$ is integrated over the whole state space; C_n is the normalization factor. $p(\mathbf{s}_n \mid \mathbf{s}_{n-1})$ is the process model and $p(\mathbf{y}_n \mid \mathbf{s}_n)$ is the observation model.

Note that Equation (4) is a recursive equation of the posterior probability $p(s_n \mid \mathrm{Y}_{1:n})$. Then we use particle filtering to implement the online process, as Figure.2 shows. The parameters are the same as [16].

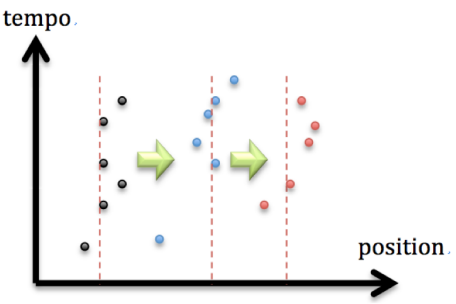

Figure 2. Particle filtering

3. NOTE ONSET INFORMATION

Until now, we have reviewed the model structure proposed in [16]. But with more experiments, the preliminary online audio-score alignment algorithm proposed in [16] was found to fail for some musical pieces, especially when there are many staccato notes. The reason is that the multi-pitch-based observation model is not robust enough in these cases. In this section, we propose to improve this observation model using onset information.

In theory, for faithful audio performances, an audio frame is expected to contain all score-indicated pitches at its corresponding score position. In practice, however, a frame may miss some expected pitches mainly because these notes are shorter than their expected lengths. Take Figure.2 as an example, the first red note is expected to be one beat long as in the score (from beat 1 to 2), but is only played for 0.7 beats in the audio performance. This can be due to the performing style (e.g. staccato), physical properties of the instrument (e.g. piano note decay) or taking a breath. Then the audio frames corresponding to the last 0.3 beats of the score note would not contain the expected pitch. In this case, the multi-pitch observation model would fail. In general, these frames often appear at the later phase of notes (close to offsets) as the gray area in Figure.3, since note offset times of a faithful audio performance do not strictly follow the score. Concurrent notes whose offset times are expected to be at the same time may also be at different times.

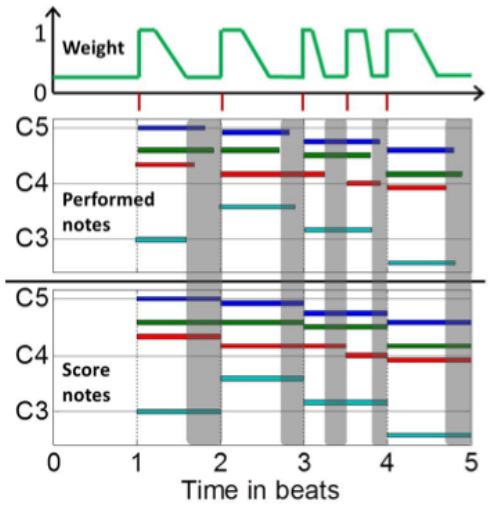

Figure 3. Note segments show with faithful frames (white) and unreliable frames (gray).

3.1 Onset Detection

In the case of multiple instruments playing at the same time, we use the spectral-based onset detection. We convert the signal into frequency domain and then capture spectral changes in frequency content.

For consideration of implementation, the frame window and hop size is the same as the observation model in Section 2.2. We use the log-magnitude Y (Figure.4.b) to calculate spectral flux $\Delta spectral$ (Figure.4.d, blue)

$$\Delta spectral(n) = \sum_{k=0}^{K} |Y(n,k) - Y(n-1,k)|_{\geq 0} \quad (5)$$

where n is the current frame, k is the frequency bins, and $|x|_{\geq 0} = (x + |x|)/2$, which means zero for negative arguments. This rectification has the effect of counting only those frequencies where there is an increase in energy, and is intended to emphasize onsets rather than offsets. Then we introduce a local average function μ (Figure.4.b, red) by setting:

$$\mu(n) = \frac{1}{W+1} \sum_{m=-W}^{0} \Delta_{spectral}(n+m) \quad (6)$$

where W determines the size of an averaging window. Then an enhanced spectral flux $\tilde{\Delta}spectral$ is obtained by subtracting the local average from $\Delta spectral$ and by only keeping the positive part:

$$\tilde{\Delta}spectral(n) = |\Delta spectral(n) - \mu(n)|_{\geq 0} \quad (7)$$

Then we set a threshold for the $\tilde{\Delta}spectral$ (Figure.4.c). If it is larger than the threshold, we know that there may exist note onset in the following frames. And then if

$$\tilde{\Delta}spectral(n) < \tilde{\Delta}spectral(n-1)$$

we consider that frame $n-1$ is an onset frame.

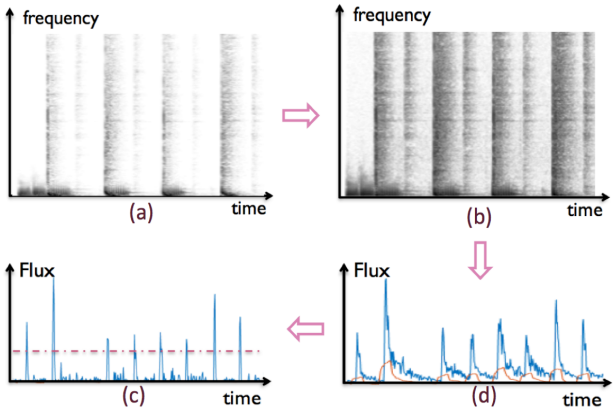

Figure 4. Flowchart of onset detection by spectral flux with (a) linear magnitude spectrogram, (b) log-magnitude spectrogram, (d) $\Delta spectral(n)$ and (c) $\tilde{\Delta}spectral(n)$.

Notice that we only use the past frames to calculate frequency flux. So the system is still online when we add onset detection.

3.2 Weighting Function

The frames where the multi-pitch information is more reliable correspond to the transient part of a note. The transient part starts right after the onset of the note, as Figure.5 (b) shows. While offset times are ambiguous and hard to detect in nature as described before, onsets are clearly defined. So we introduce the weighting function in Equation (8):

$$W = 1 - \frac{C}{1 + \exp\{-a \cdot (N - N_{onset}) + b\}} \quad (8)$$

where N is the current frame, N_{onset} is the onset frame it just pass by and a, b, C are parameters to shape the weighting function. Since we do not know exactly when the transient part will end, the reliability of the frames that are further away from onsets will be smaller. As the Figure.5 shows:

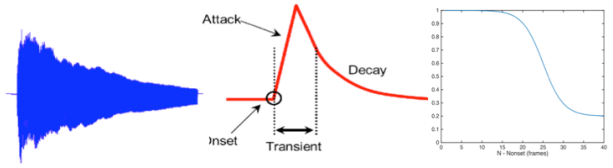

Figure 5. Waveform of one piano note (a), envelope (b) and one possible function simulation (c).

However, there is no universal shape of the weighting function to simulate every circumstance well. So we only simulate a piano note decay as our weighting function.

By imposing a weighting function on audio frames so that more reliable frames will have larger weights. The weighting function will be multiplied with the multi-pitch observation model $p(\mathbf{y}_n | \mathbf{s}_n)$ in Equation (4). More reliable frames will have stronger influence through the observation model to the alignment, and the robustness of the score follower will be improved.

Then we apply the weighting function value to the observation model. By this way, we add more reliability to the frames right after an onset time than frames in the later phase of the note.

4. EXPERIMENTAL RESULTS

Experimental material contains some of Bach's Chorales and RWC Classical Music Dataset. Since there is no ground-truth alignment in most music database, we can only measure the result by hearing the warped audio or MIDI. The alignment results are improved when dealing with piano music. But the current method still has some limitations:

1. It can only work well for piano music with obvious onsets and it will get lost when it's hard to detect onset such as strings music.
2. Even in piano music, the alignment result will drop when the volume distribution of different pitch is different. Similar circumstance happens when the sustain pedal of piano is often used. The echo will make the observation model fail.

5. CONCLUSION

In this paper, we utilize the note onset information to improve an online music alignment algorithm and get better result in piano music. In terms of the limitations, there are some possible improvements in the future work.

The first limitation is about the robust of onset detection. It's easy to detect piano note onsets while soft onset from strings is difficult to detect. The spectral flux is one onset detection method. It gets high accuracy on piano music with low latency, so we can still get an online system after applying this method. But a more advanced online onset detection method is necessary.

For the second limitation, since the current method highly relies on the multi-pitch likelihood model, it's sensitive to inconformity of the spectra. Perhaps we need to

train more models based on volume distribution of each pitch to consider the echo from last musical notes.

REFERENCES

[1] Roger B. Dannenberg, Christopher Raphael, "Music Score Alignment and Computer Accompaniment".

[2] N. Orio and D. Schwarz, "Alignment of monophonic and polyphonic music to a score," in *Proc. International Computer Music Conference (ICMC)*, 2001.

[3] N. Hu, R.B. Dannenberg and G. Tzanetakis, "Polyphonic audio matching and alignment for music retrieval," in *Proc. 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, New York, USA, 2003, pp. 185-188.

[4] S. Ewert, M. Muller, and P. Grosche, "High resolution audio synchronization using chroma onset features," in *Proc. IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, pp. 1869-1872, 2009.

[5] P. Cano, A. Loscos, and J. Bonada, "Score-performance matching using HMMs," in *Proc. International Computer Music Conference (ICMC)*, 1999.

[6] C. Raphael, "Automatic segmentation of acoustic musical signals using hidden markov models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 4, pp. 360-370, 1999.

[7] C. Raphael, "Aligning music audio with symbolic scores using a hybrid graphical model," *Machine Learning*, vol. 65, pp. 389-409, 2006.

[8] C. Joder, S. Essid and G. Richard, "A conditional random field frame- work for robust and scalable audio-to-score matching," *IEEE Trans. Speech, Audio and Lang. Process*, in press.

[9] R.B. Dannenberg, "An on-line algorithm for real-time accompaniment," in *Proc. International Computer Music Conference (ICMC)*, pp. 193-198, 1984.

[10] B. Vercoe, "The synthetic performer in the context of live performance," in *Proc. International Computer Music Conference (ICMC)*, pp. 199-200, 1984.

[11] M. Puckette, "Score following using the sung voice," in *International Computer Music Conference (ICMC)*, 1995.

[12] L. Grubb and R.B. Dannenberg, "A stochastic method of tracking a vocal performer," in *Proc. International Computer Music Conference (ICMC)*, 1997.

[13] N. Orio and F. Dechelle, "Score following using spectral analysis and hidden markov models," in *Proc. International Computer Music Conference (ICMC)*, 2001.

[14] C. Raphael, "A Bayesian network for real-time musical accompaniment," in *Proc. Advances in Neural Information Processing Systems (NIPS)*, 2001.

[15] A. Cont, "Realtime audio to score alignment for polyphonic music instruments using sparse non-negative constraints and hierarchical HMMs," in *Proc. IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, 2006.

[16] Zhiyao Duan, Bryan Pardo, "Soundprism: An Online System for Score-informed Source Separation of Music Audio", *IEEE Journal of Selected Topics in Signal Process*, vol. 5, no. 6, pp. 1205-1215, 2011.

[17] Zhiyao Duan, Bryan Pardo, and Changshui Zhang, "Multiple fundamental frequency estimation by modeling spectral peaks and non-peak regions", *IEEE Trans. Audio Speech Language Process*, vol. 18, no. 8, pp. 2121-2133, 2010.