A REAL-TIME SCORE FOLLOWER FOR MIREX 2015

Francisco J. Rodriguez-Serrano, Pedro Vera-Candeas
Telecommunication Engineering Department
University of Jaen
{fjrodrig,pvera}@ujaen.es

Julio J. Carabias-Orti
Music Technology Group
Universidad Pompeu Fabra
julio.carabias@upf.edu

ABSTRACT

This abstract describes a proposed score follower submitted to the MIREX 2015 Real-time Audio to Score Alignment (a.k.a. Score Following) evaluation task.

1. INTRODUCTION

A real-time score follower is an algorithm that synchronizes a performance with its corresponding score in real time. It should estimate the score position for each performance input time frame. This estimation is made in an online fashion (i.e without information from the future frames). In this MIREX task, the score information is given in MIDI format and the audio signals are given as WAV files.

In this work, we present a realtime score follower based on spectral factorization and online Dynamic Time Warping (DTW). The presented system has two separated stages, preprocessing and alignment. On the first one, we convert the score into a reference audio signal using a MIDI synthesizer software and we analyze the provided information in order to obtain the spectral patterns (i.e. basis functions) for each combination of the concurrent notes given at the score. These basis functions are learned from the synthetic MIDI signal using a method based on Alternated Non-Linear Least Squares (ANLS), where the gains are initialized with transcription obtained from the MIDI file. On the second stage, a realtime signal decomposition method with fixed basis functions per combination of notes is applied over the magnitude spectrogram of the input signal resulting in a distortion matrix that can be interpreted as the matching likelihood between each score-time frame and each real time frame. Finally the score alignment is obtained using an on-line Dynamic Time Warping (DTW) over the distortion matrix in order to find the path with the minimum cost and then determine the notes real duration.

2. SYSTEM DESCRIPTION

An overview of the evaluated score following system and an example of the alignment is shown in Figure 1.

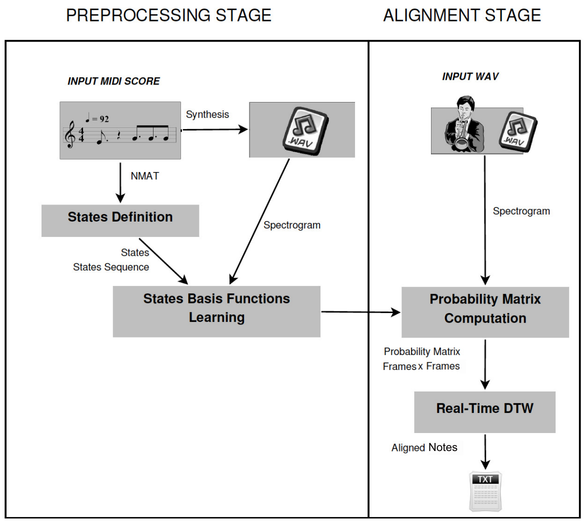

Figure 1. Proposed Real-Time Score Follower Block Diagram.

2.1 Preprocessing Stage

2.1.1 *States Definition*

The aim of this stage is to compute the states and states sequence from the MIDI data. A state is defined as a combination of notes that occurs simultaneously in the ground truth transcription obtained from the MIDI file. It can be formulated as

$$S_k = \{n_j^k, j = 1, \ldots, J, k = 1, \ldots, K\} \tag{1}$$

where n_j^k is the note played by instrument j, k is the state index, J is the total number of instruments and K the total number of states. The states sequence is a 1xM vector that provides information about the states transitions over the MIDI data. It is is defined as

$$\Psi = \{S_k^m, 1 \leq m \leq M\} \tag{2}$$

where S_k^m is the k-th state occurring at the m-th position in the state sequence vector and M is the total number of transitions between states.

2.1.2 *Basis Functions Learning*

Once the states and the states sequence have been defined, the basis functions associated to each state are learned. To this end, we use a supervised method based on Alternating Non- Negative Least Squares (ANLS) method [1].

First of all, the signal is defined as:

$$x(f,t) \approx \hat{x}(f,t) = \sum_{k=1}^{K} g_k(t), b_k(f) \tag{3}$$

where $x(f,t)$ is the magnitude spectrogram of the synthetic signal generated from the MIDI file with a sequencer, $\hat{x}(f,t)$ is the estimated signal spectrogram, $g_k(t)$ is the gain of the corresponding basis function for state k at frame t, and $b_k(f), k = 1, \ldots, K$ are the bases.

After that, the gains are initialized with the information from the MIDI file. At each time frame t, the corresponding note (k) gains are set to 1 while the other ones remains as zero. The basis are initialized as the mean of the magnitude spectrogram $x(f,t)$. Then an ANLS algorithm is run in order to obtain a set of descriptive basis for all the combinations of notes with only a few iterations.

Algorithm 1 Alternated Non-Negative Least Squares Algorithm for gains estimation

```
Initialize $g_k(f)$ with the MIDI file information.
Initialize $b_k(f)$ with the mean of the magnitude spectrogram $x(f,t)$.
for i=1:5 do
    Update the basis using eq. (5).
    Update the gains using eq. (4).
end for
```

$$g_{k,i}(t) = \frac{\sum_f x(f,t) b_{k,i-1}(f)^{\beta-1}}{\sum_f b_{k,i-1}(f)^{\beta}} \tag{4}$$

$$b_{k,i}(f) = \frac{\sum_t x(f,t) g_{k,i}(t)^{\beta-1}}{\sum_t g_{k,i}(t)^{\beta}} \tag{5}$$

2.2 Alignment Stage

2.2.1 Probability Matrix Computation

As explained in section 2.1.2, the basis functions $b_k(f)$ for each state are trained in advance using the MIDI data and kept fixed. Each basis function models the spectrum of an unique state. Now, the aim is to compute the gain matrix $g_k(t)$ and the final cost matrix $D\beta(t,k)$ (see eq.(6)) that measures the likelihood between the estimated and the real spectrogram. The process is detailed in Algorithm 2.

Algorithm 2 Probability Matrix Computation

```
1 Initialize $b_k(f)$ with the values learned in section 2.1.2 and $g_k(t)$ with random positive values.
2 Update the gains using eq.(4).
3 Repeat step 2 until the algorithm converges (or maximum number of iterations is reached).
4 Compute the distortion matrix $D\beta(x|\hat{x})$ using eq.(6).
```

$$D\beta(x|\hat{x}) = \begin{cases} \frac{1}{\beta(\beta-1)}\left(x^{\beta} + (\beta-1)\hat{x}^{\beta} - \beta x \hat{x}^{\beta-1}\right) & \beta \in (0,1) \cup (1,2] \\ x \log \frac{x}{\hat{x}} - x + \hat{x} & \beta = 1 \\ \frac{x}{\hat{x}} + \log \frac{x}{\hat{x}} - 1 & \beta = 0 \end{cases} \tag{6}$$

As can be seen, the distortion matrix $D\beta(x|\hat{x})$ provides the system information about the similitude of each state k basis function with the real signal spectrum at each frame t. Using this information, we can directly compute the probability matrix for the state sequence as

$$D(t_r, t_m) = \{|D\beta_k^m(t)|\, 1 \leq t_m \leq T_m\} \tag{7}$$

were t_r is the real time frame index,t_m is the MIDI time frame index, T_m is the synthetic duration of the MIDI file and m is the index of current state over the state sequence.

Therefore, we should find the minimum cost path in order to determine the duration in the real performance of each state in the sequence. To this end, we have applied a real-time DTW as explained in the following section.

2.2.2 Real State Sequence Estimation by DTW

We used the following constrained DTW path:

$$D(t_r, t_m) = \min \left\{ \begin{array}{l} D(t_r - 1, t_m) + d(t_r, m) \\ D(t_r - 1, t_m - 1) + 1.2 d(t_r, m) \end{array} \right\} \tag{8}$$

$d(t_r, m)$ is value of the distortion computed with the β-divergence function for the t_r-th frame and the state in the m-th position in the sequence and $D(t_r, t_m)$ is the accumulated cost value at the t_r-th frame and the t_m-th state at the sequence. Note that this constrained path inhibits occurrence of vertical steps since only one state can be active at each frame.

2.2.3 Real-Time DTW

Standard DTW assumes off-line search and the estimated path is obtained by backtracing of whole the signal. To extend DTW for the on-line search without backtracing, we simply select the reference state which has the smallest accumulated distance with the current performance frame t.

Sometimes the distortion value of $D\beta$ computed for consecutive states and the same frame is very similar. It occurs when the basis of both states are similar (i.e. when the combination of notes of two consecutive states are almost the same). This situations could generate some mistakes at the DTW path. In order to avoid this, a new path is obtained. It becomes by taking into account only those state-transition points from the DTW path where the correlation between the basis of the consecutive states are different enough. These trustworthy points are named moorings points. Then, the note alignment interpolation is made by using this new path which goes over the moorings points of the original DTW path.

We have sent three versions of this algorithm. One of them (RVC6, SF_BACK) does not report any note onset until the the next mooring point is reached. Other submitted version (RVC5, SF_FWD) uses the information from the two previous moorings points to estimate the onset time of all the notes before the next mooring point. So that, RVC6 will always have a positive delay when reporting each note, while RVC5 will always have a negative delay, it report each note before its time is reached. Finally, RVC4 version of the algorithm implements a global tempo estimation of the last 10 mooring points. This tempo information is used for estimating the onset time of the following notes, tempo estimation is updated when a mooring point is found. This is a more realistic scene for an automatic accompaniment system.

3. EVALUATION

Results are published in the MIREX 2015 results website.

4. REFERENCES

[1] Berry M.W., Browne M., Langville A, Pauca V., Plemmons R., “Algorithms and applications for aproximate nonnegative matrix factorization,” *Comput. Statist. Data Anal.*, 52, pp. 155173 (2007).