BEAT-TRACKING USING A PROBABILISTIC FRAMEWORK AND LINEAR DISCRIMINANT ANALYSIS

Geoffroy Peeters

Ircam Sound Analysis-Synthesis Team - CNRS STMS

geoffroy.peeters@ircam.fr

ABSTRACT

This paper deals with the problem of beat-tracking in an audio-file. Considering time-variable tempo and meter estimation as input, we study two beat-tracking approaches. The first one is based on an adaptation of a method used in speech processing for locating the Glottal Closure Instants. The results obtained with this first approach allow us to derive a set of requirements for a robust approach. This second approach is based on a probabilistic framework. In this approach the beat-tracking problem is formulated as an "inverse" Viterbi decoding problem in which we decode times over beat-numbers according to observation and transition probabilities. A beat-template is used to derive the observation probabilities from the signal. For this task, we propose the use of a machine-learning method, the Linear Discriminant Analysis, to estimate the most discriminative beat-template. We finally propose a set of measures to evaluate the performances of a beat-tracking algorithm and perform a large-scale evaluation of the two approaches on four different test-sets.

1. INTRODUCTION

Beat-tracking, i.e. locating the times in an audio signal where beats are perceived or notated in the corresponding score, is one of the most challenging subject in the music-audio research community. This is due to the large use of the beat information in many applications: beat-synchronous analysis (such as for score alignment or for cover-version identification), beat-synchronous processing (time-stretching, beat-shuffling, beat-slicing ...), music analysis (beat taken as a prior for pitch estimation or onset detection) or visualization (time-grid in audio sequencers). This is also due to the complexity of the task. While tempo estimation is mainly a problem of periodicity detection (with the inherent octave ambiguities), beat-tracking is both a problem of periodicity detection and location of the periods inside a signal (with the inherent ambiguities of the rhythm itself).

Considering that the best results obtained in the last Audio Beat Tracking contest (MIREX-2006) are far from being perfect, this problem is far from being solved. If most beat-tracking algorithms achieve good results for most rock, pop or dance music track (except for highly compressed tracks), this is not the case when considering classical, jazz or world music. Moreover recent Western music styles such as Drum'n'Bass or R'n'B (which use more complex rhythms than pop, rock and dance) bring back the problem to the mainstream music.

Considering the numerous methods proposed for beat-tracking, it would be difficult to summarize them here. We therefore refer the reader to [1] and [2] for a good overview of the recent advances in this domain.

This work was partly supported by the "Quaero" Programme, funded by OSEO, French State agency for innovation.

In this paper we present two different approaches for locating the beat-markers. The first one is based on an algorithm developed in the framework of speech processing for locating the Glottal Closure Instants. We name it P-sola. We apply this method here to the problem of beat-tracking. The second one uses a probabilistic formulation of the problem of beat-tracking with observation and transition probabilities. Because of the use of a probabilistic framework, it shares some ideas with the methods based on Dynamic Programming [3] [4] [5] (as opposed to the ones based on Multiple-Agent [6] [7]), although the formulation of the probabilistic framework is different in our case and the input is a "continuous" onset function rather than "discrete" onsets.

Paper organization: In part 2, we give an overview of the system used to estimate the onset-energy-function, time-variable tempo and meter which are used as input variables in the remaining of the paper. In part 3, we propose a P-sola beat-tracking algorithm and highlight the drawbacks of it. In part 4, we propose a probabilistic model for beat-marking. In part 4.4.1, we propose a machine learning approach to estimate the best beat-template to compute the observation probabilities. Finally in part 5, we propose a set of evaluation measures and perform a large-scale evaluation of the two beat-tracking algorithms on four test-sets.

2. OVERALL PRESENTATION OF THE TEMPO/METER ESTIMATION ALGORITHM

This paper concerns the beat-tracking problem. For this, we consider an onset-energy-function, time-variable tempo and meter as input parameters of the algorithms. The system used for the estimation of these input parameters is the one described in [8]. Since the evaluation performed at the end of the paper will used these input parameters, we briefly summarized their estimation here. The **first stage** of the system described in [8] extracts an onset-energy-function. This function is a 172Hz function with high values at the onset positions and low values at the other positions. This function is obtained by computing a reassigned-spectral-energy-flux function from the signal (time and frequency reassignment of the spectrogram are used for better time and frequency resolution). Log-scaling, adaptive thresholding, low-pass, high-pass filtering, Half-Wave-Rectification and summation are then applied to obtain the function. In the remaining, we note this function $f(t)$. The **second stage** of the system measures the dominant periodicities over time of $f(t)$. The dominant periodicities are obtained by combining a Discrete Fourier Transform with a Frequency-Mapped Auto-Correlation Function [9]. The combination of both functions allows to better distinguish the dominant periodicities from the sub-harmonics and over-harmonics in $f(t)$. The **last stage** of

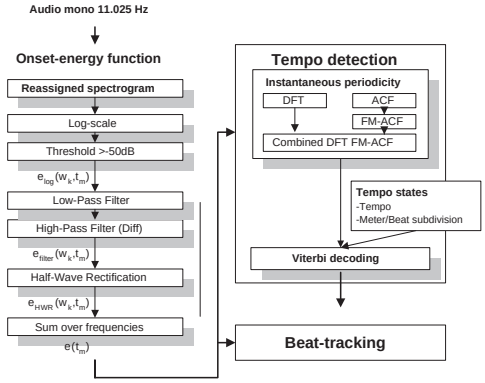

Figure 1: *Overall schema of tempo/ meter estimation system of [8].*

the system estimates the tempo and meter over time from the observed dominant periodicities. For this, we define a hidden Markov model which states are the specific combinations of possible tempi and meters (2/4, 3/4 or 6/8). The observation probabilities come from the comparison of state's templates to the observed dominant periodicities. The best path over time of tempo/ meter is obtained using a Viterbi decoding algorithm. In the remaining, we note $\mathrm{Tb}(t)$ the estimated temporal period between two beats ($\mathrm{Tb}(t) = 60/bpm(t)$). The overall schema of the system is represented in Figure 1.

3. BEAT-TRACKING USING P-SOLA BASED METHOD

P-sola (Pitch Synchronous Over-Lap Add) is a speech processing method allowing pitch-shifting and time-stretching of a speech signal. The first stage of the processing aims at locating the Glottal Closure Instants (GCIs) of the speech signal. The characteristics of these GCIs are (a) to be close to the local maxima of the energy-signal and (b) to have an inter-distance close to the local pitch-period $T_0(t)$ of the signal. The problem of locating the beat-markers is close to the one of locating the GCIs: (a) the positions of the beat-markers are often close to local maxima of the onset-energy function $f(t)$, (b) the inter-distance between successive beat-markers is close (equal) to the local tempo-period $\mathrm{Tb}(t)$. Because of this proximity, we propose here an adaptation of a method we have previously developed for GCIs location [10] to the case of beat-tracking . The method proceeds in two separated stages. The first one locates a set of local maxima of $f(t)$ with an inter-distance close to $\mathrm{Tb}(t)$. The second one performs a least-square optimization in order to satisfy simultaneously two constraints: (a) markers close to the local maxima of $f(t)$, (b) inter-distance between markers close to $\mathrm{Tb}(t)$.

3.1. Local maxima detection

We define a vector of times $\Theta = [\theta_0, \theta_1, ..., \theta_i, ...]$. The values of θ_i are recursively defined. For this, we define around the time θ_i, an interval $I_i = \left[\theta_i - \frac{\mathrm{Tb}_{i-1}}{\alpha}, \theta_i + \frac{\mathrm{Tb}_i}{\alpha}\right]$ in which Tb_i is the local tempo period around θ_i and α defines the relative length of the interval ($\alpha \in]2, \infty[$). Small values of α (large intervals) favor energy constraint, while large values of α (small intervals) favor periodicity constraint.

The maximum of $f(t)$ in the interval I_i is noted τ_i. θ_{i+1} is given by $\theta_{i+1} = \tau_i + \mathrm{Tb}_i$. The process is repeated for several initialization time θ_0 of the vector Θ. The vector Θ with the initialization time leading to the maximum value of $\sum_i f(\tau_i)$ defines the best set of energy markers τ_i.

3.2. Least-square optimization of energy and tempo constraints

We define m_i as the beat-markers to estimate. These markers must satisfy simultaneously the two constraints: (a) markers m_i must be close to the local maxima τ_i of the onset-energy function $f(t)$, (b) the inter-distance between two successive markers m_i must be equal to the local tempo-period Tb_i. These constraints can be expressed mathematically as:

$$\begin{cases} (a) : \mathrm{m}_i = \tau_i \\ (b) : \mathrm{m}_i - \mathrm{m}_{i-1} = \mathrm{Tb}_{i-1} \\ (b) : \mathrm{m}_{i+1} - \mathrm{m}_i = \mathrm{Tb}_i \end{cases} \quad (1)$$

Given that modifying one m_i has consequences on the left and right periods (the same for m_{i-1}, m_{i+1}), we need to solve the above equations for all m_i simultaneously. This leads to the minimization over m_i of the following sum of the square error ϵ:

$$\epsilon = \sum_{i \in I} \left[((\mathrm{m}_i - \mathrm{m}_{i-1}) - \mathrm{Tb}_{i-1})^2 + \beta(\mathrm{m}_i - \tau_i)^2 \right] \quad (2)$$

where β is a weights: $\beta > 1$ favors the energy constrains (a), $\beta < 1$ favors the periodicity constrains (b). The solution to this problem is the following. If we note $\mathbf{m} = [\mathrm{m}_0 \mathrm{m}_1 ... \mathrm{m}_i ... \mathrm{m}_I]$ the vector of beat markers to estimate, their optimal positions are given by

$$\mathbf{m} = \mathbf{M^{-1}} \cdot \begin{pmatrix} 0 & -\mathrm{Tb}_0 & +\beta\tau_0 \\ \mathrm{Tb}_0 & -\mathrm{Tb}_1 & +\beta\tau_1 \\ \vdots & \vdots & \vdots \\ \mathrm{Tb}_{i-1} & -\mathrm{Tb}_i & +\beta\tau_i \\ \vdots & \vdots & \vdots \\ \mathrm{Tb}_{I-2} & -\mathrm{Tb}_{N-1} & +\beta\tau_{I-1} \\ \mathrm{Tb}_{I-1} & 0 & +\beta\tau_I \end{pmatrix} \quad (3)$$

where $\mathbf{M}$ is defined as

$$\mathbf{M} = \begin{pmatrix} 1+\beta & -1 & 0 & \dots & & \\ -1 & 2+\beta & -1 & 0 & \dots & \\ 0 & -1 & 2+\beta & -1 & 0 & \dots \\ & \ddots & \ddots & \ddots & \ddots & \ddots \\ & & 0 & -1 & 2+\beta & -1 \\ & & \dots & 0 & -1 & 1+\beta \end{pmatrix} \quad (4)$$

For the evaluation presented in part 5, we will use $\alpha = 8$ and $\beta = 1$.

4. BEAT-TRACKING USING INVERSE VITERBI FORMULATION

4.1. Motivations for a probabilistic model

When experimenting with the method presented in part 3, a set of marking problems were observed that we highlight here.

At the first stage of the P-sola algorithm, a binary decision is taken: a time is a local maximum of $f(t)$ or not. Also only one local maximum per period Tb is estimated. The consequences of this

are: - If the estimated local maximum is not the one corresponding to the beat positions, the marking will be incorrect because the second stage of the algorithm will also fail. - If there is no local maximum in the signal (for example a part of a track without any onset such as a beat in the middle of a silence part), the algorithm also fails. A solution to these problems would be to have several candidates for the local maxima and associated probabilities.

At the second stage, there is no adaptive weighting between the constraints (a) "close-to–local-maxima" and (b) "inter-distance close to local period". The two constraints are taken into account with a constant weight β over time. Ideally, if a part of a track has no clear onsets, the periodicity constraint should be favored.

For all these reasons, we formulate the beat-marking problem in a probabilistic framework, with probabilities associated to the times and to the transitions between times. The formulation proposed in the following allows applying a Viterbi decoding algorithm [11] but requires inverting the x and y axis of the usual formulation. Hence we call it "inverse" Viterbi decoding. We first present this inversion of the axis.

4.2. Viterbi and inverse Viterbi decoding

Viterbi decoding: We take here as example the formulation of the Viterbi decoding as used for the tempo/ meter tracking in [8]. In this formulation, a hidden state s_{ij} is defined as a specific combination of a tempo i and a meter j. We estimate the best succession of states s_{ij} over time given the probability to observe a given state s_{ij} at a given time t_k: $p_{obs}(o(t_k)|s_{ij})$, and given the probability to transit from a state s_{ij} to a state $s_{i'j'}$: $p_{trans}(s_{i'j'}|s_{ij})$ (the transition probability aims to ensure tempo and meter continuity). We decode states over times. This is illustrated in the left part of Figure 2.

Inverse Viterbi decoding: We want to formulate the Viterbi algorithm in order to decode the beat-marker positions over time. This raises the problem that beat-marker positions are "times" that we want to decode over "time". In order to solve this problem, we inverse the x and y axis as follows: we decode times over beat-numbers b_k (b_k is a monolithically increasing function). For this, we define the states s_i as the various times t_i of the time axis of the track: s_i is defined as "time t_i is a beat". We then look for the best succession of states s_i (or times t_i) that explain the beat-number succession b_k. We define - an initial probability $p_{init}(s_i)$ which represents the probability to be in hidden state s_i ("t_i is a beat") at the beginning of the decoding, - an emission probability $p_{obs}(\underline{o}(t)|s_i)$ which is the probability to observe $\underline{o}(t)$ given a specific state s_i (given that "t_i is a beat"), - a transition probability $p_{trans}(s_{i'}|s_i)$ which represents the probability to transit from state s_i (or "t_i is beat") to state s'_i (or "t'_i is the next beat"). We compare the Viterbi formulation to the inverse Viterbi formulation in Figure 2.

4.3. Initial probability $p_{init}(s_i)$

$p_{init}(s_i)$ represents the probability to be in hidden state s_i ("t_i is a beat") at the beginning of the decoding. We favor t_i to be a time close to the beginning of the track. For this, we use a gaussian function with $\mu = 0$ and $\sigma = 0.5$ evaluated on the t_i of all the states s_i.

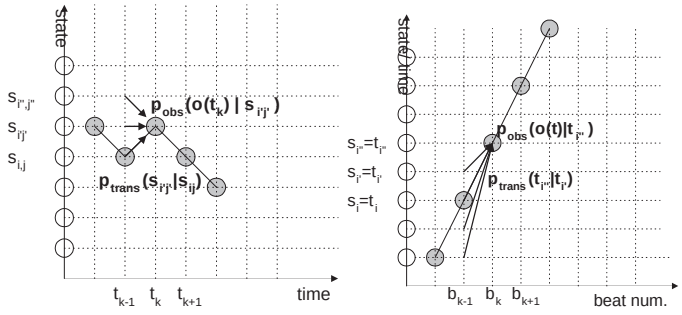

Figure 2: *[Left:] Viterbi decoding: we decode the states s_{ij} over time t_k given the probability to observe a state $s_{i'j'}$ at time t_k ($p_{obs}(\underline{o}(t_k)|s_{i'j'})$) and given the probability to transit from state s_{ij} to state $s_{i'j'}$ ($p_{trans}(s_{i'j'}|s_{ij})$). [Right:] Inverse Viterbi decoding: we decode the states s_i (or times t_i) over the beat-number b_k given the probability to observe a state $s_{i''}$ (or time $t_{i''}$) at beat number b_k ($p_{obs}(\underline{o}(t)|s_{i''})$) and given the probability to transit from state $s_{i'}$ (or time $t_{i'}$) to state $s_{i''}$ (or time $t_{i''}$) ($p_{trans}(t_{i''}|t_{i'})$).*

4.4. Observation probabilities $p_{obs}(\underline{o}(t)|s_i)$

The states s_i are defined as the various times t_i of the time axis of the track. s_i is defined as "time t_i is a beat". We associate to each state s_i an emission probability, which represent the probability of observing $\underline{o}(t)$ given that we are in state s_i, i.e. given that t_i is a beat. In practice, we estimate this probability using $p_{obs}(s_i|\underline{o}(t)) = p_{obs}(t = t_i) \cdot p_{obs}(s_i|\underline{o}(t))$. The hidden state s_i has a non-nul emission probability only when $t = t_i$ in $\underline{o}(t)$. We associate to each state s_i an observation probability, which represent the likelihood that this state s_i is a beat. $p_{obs}(s_i|\underline{o}(t))$ is estimated by computing the likelihood that a beat-template $g_{\mathrm{Tb}}(t)$ starting at time t_i and corresponding to the local tempo $\mathrm{Tb}(t_i)$ explains the content of the onset-energy-function $\underline{o}(t) = f(t, t \in [t_i, t_i + 4\mathrm{Tb}])$. This likelihood is estimated using correlation. The beat-template can be a simple function with values of 1 at the expected beat-position and 0 otherwise (as in [3]). We propose here a method that allows finding by machine-learning the beat-template that maximizes the discrimination between the correlation values obtained when t_i is a beat-position and when t_i is a non-beat position..

4.4.1. Learning the best beat-template by Linear Discriminant Analysis

The beat-template must be chosen such as (a) to have the maximum correlation with the local signal when t_i is a beat-position, (b) to provide the largest discrimination between the correlation values when t_i is a beat-position and a non-beat position. The condition (b) is needed in our case since the values of correlation will be used as observation probability in our framework. In the following, we only discuss the case of a "binary subdivision of the beat" and "binary grouping of the beat into bar". Extension to other meters is straightforward.

Using a discrete notation, we note $g(1)...g(N)$ the discrete sequence of values of the beat-template representing a one-bar duration beat-pattern. Considering a 4/4 measure, $g(1)$ represents the value of the beat-template at the downbeat position, $g(1 + jN/4)$

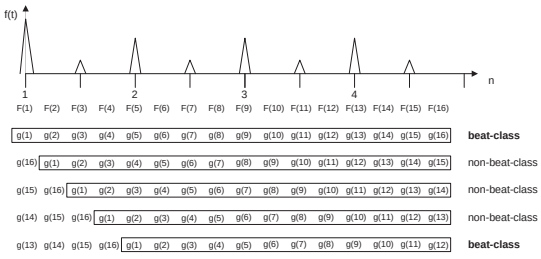

Figure 3: *Correlation computation as a multiplication of signal $F(n)$ by weights $g(n)$: time-norm. and sampled observation function $F(n)$ and time-norm. and sampled beat-template $g(n)$.*

with $j \in [1, 2, 3]$ the values at the other beat positions. We define $F(n)$ as the function obtained by sampling the local values of $f(t, t \in [t_i, t_i + 4\text{Tb}])$ by N value: $F(1) = f(t_i)...F(N) = f(t_i{+}4\text{Tb})$. We look for the beat-template (the values of $g(n), n \in [1, N]$) which maximize the correlation with $F(n)$ when t_i is a beat-position and minimize it when t_i is not a beat-position. If t_i is a beat-position (hence $F(1 + jN/4)$ with $j \in [0, 1, 2, 3]$ are also beat positions), this can be expressed as

- $F(1+j)g(1)+F(2+j)g(2)+...+F(N+j)g(N)$ must have a maximum value for $j \in [0, N/4, 2N/4, 3N/4]$,
- $F(1+j)g(1)+F(2+j)g(2)+...+F(N+j)g(N)$ must have a minimum value for all the other values of j.

We illustrate this in Figure 3 for the case $N = 16$.

According to the equations above, the problem of finding the best values of $g(n)$ is close to the problem of finding the best weights to apply to the dimensions of multi-dimensional observations in order to maximize class separation. This problem can be solved using Linear Discriminant Analysis (LDA) [12]. In our case the weights are the $g(n)$, the dimensions of the feature vectors are the successive values of $F(n)$ and the classes are "beat" and "non-beat". We therefore apply a two-class Linear Discriminant Analysis to our problem.

Creating observations for the two-classes LDA problem: Linear Discriminant Analysis necessitates observations to learn from. We therefore create observations for the two classes "beat" and "non-beat". These observations are coming from a test-set annotated into beat and down-beat positions. Knowing the down-beat locations, we create for each track l of the test-set and for each annotated bar m inside a track, the corresponding $F_{l,m}(n)$. For a specific track, we compute the vector $F_l(n)$ by averaging the values of $F_{l,m}(n)$ over all the bars of the track. In Figure 4, we illustrate this for the RWC-Popular-Music test-set [13]. The upper part represents the vectors $F_l(n)$ for the 100 tracks of the test-set for the case $N = 64$. The lower part represents the average (over the 100 tracks) vector $F(n)$.

From the observed $F_l(n)$, we then create two sets of observations corresponding to the two classes "beat" and "non-beat". This is obtained simply by shifting (circular permutation is assumed in the following) $F_l(n)$ as follows:

- "beat" class: the four patterns $F_l^b(n) = F_l(n + j)$ with $j \in [0, N/4, 2N/4, 3N/4]$,
- "non-beat" class: all the remaining patterns $F_l^{nb}(n) = F_l(n+j)$ with $j \in [1, N]$ $j \neq 0, N/4, 2N/4, 3N/4$

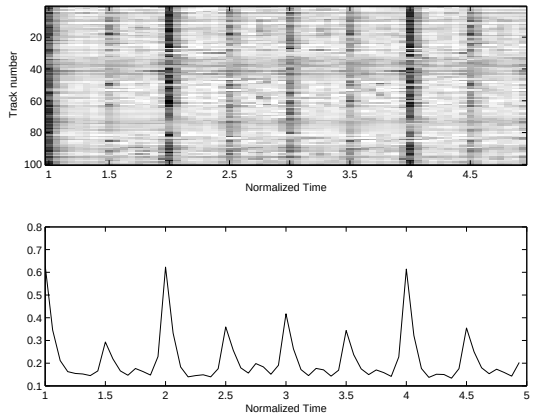

Figure 4: *[Top part] Time-norm. and sampled observation function $F_l(n)$ for each of the 100 tracks of the RWC Popular-Music test-set [13] [Bottom part] Average (over all tracks) value $F(n)$.*

From the set of L observations, we therefore create $4L$ observations representing the "beat" class (the sequences starting in one of the 4 beat positions), and $(N-4)*L$ observations representing the "non-beat" class (all the other sequences).

Linear Discriminant Analysis: We then apply Linear Discriminant Analysis considering the two new set of observations ($F_l^b(n)$ and $F_l^{nb}(n)$) and their associated classes "beat" and "non-beat". We compute the matrix $\underline{\underline{U}}$ such that after transformation of the features by this matrix, the ratio of the Between-Class-Inertia and the Total-Inertia is maximized. If we note $\underline{u}$ the column vectors of $\underline{\underline{U}}$, this maximization leads to the condition $\underline{\underline{T}}^{-1}\underline{\underline{B}}\underline{u} = \lambda\underline{u}$ where $\underline{\underline{T}}$ is the Total-Inertia matrix and $\underline{\underline{B}}$ the Between-Class-Inertia matrix. The column vectors of $\underline{\underline{U}}$ are then given by the eigen vectors of the matrix $\underline{\underline{T}}^{-1}\underline{\underline{B}}$ associated to the eigen values λ. Since the problem is a two-classes problem, only one column remains in $\underline{\underline{U}}$. This column gives us the weights to apply to $F(n)$ in order to obtain the best separation between the classes "beat" and "non-beat". It therefore defines the best beat-template $g(n)$.

In Figure 5, we illustrate this for the RWC-Popular-Music test-set [13]. We represent (in thin line) the average (over the 100 tracks) vector $F(n)$. We represent (in thick line) the values of $g(n)$ obtained by Linear Discriminant Analysis. As one can see, the LDA-trained beat-template assigns - large positive weights at the beat-positions (1, 2, 3, 4) and - negative weights at the counter-beat positions (1.5, 2.5, ...) and at the just-before/ just-after beat positions. The use of negative weights is a major difference with the weights used in usual beat-templates (as in [3]) which only use positive or zero weights. The specific locations of the negative weights allow reducing the common counter-beat detection error (negative weights at the counter-beat positions) and the precision of the beat location (negative weights at the just-before/ just-after beat positions). This wouldn't be achieved by using a model where all the positions outside the main beats are set to a constant negative number.

Use of the LDA-trained beat-templates: In the beat-tracking process, the LDA-trained beat-templates $g(n)$ are used to create the beat-template which corresponds to the local tempo $\text{Tb}(t_i)$. For this, $g(n)$ is considered as representing the interval $[0, 4\text{Tb}(t_i)]$ and is interpolated to provide the values corresponding to the sampling rate of $f(t)$: 172 Hz. In order to save computation time, the

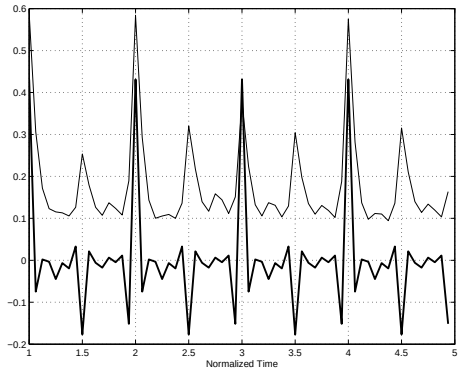

Figure 5: *[Thin line] Average value $F(n)$ for the RWC-Popular-Music test-set [Thick line] LDA-trained beat-template $g(n)$.*

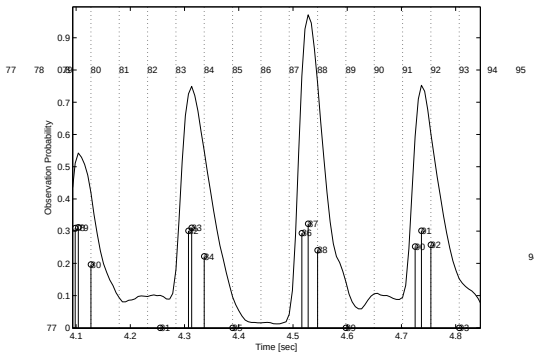

Figure 6: *Observation probability for state s_i: onset-energy-function (continuous line), initial positions of states s_i (dotted vertical lines), final positions of states s_i (cont. vert. linez) and associated observation probability (cont. vert. lines' height).*

values of $g_{\text{Tb}}(t)$ for all possible tempo Tb can be stored in a table. In the evaluation presented in part 5, we will compare various sampling and interpolation method of the LDA-trained beat-templates.

4.4.2. Optimization considerations:

In order to reduce the number of states s_i[1], we apply a discretization of the time axis of the track. A sampling rate of 20Hz (hop size of 50ms) is used for the creation of the states (20 states/ second). The observation probability (obtained using the beat-template) is then computed for each of the discrete states s_i. Because of this discretization, we reassign the time t_i of the state s_i to the position around t_i which leads to the maximum correlation between the local signal $f(t, t \in [t_i, t_i + 4\text{Tb}])$ and the one-bar beat-template $g_{\text{Tb}}(t)$. The horizon on which the maximum correlation is searched for is proportional to the local tempo $\text{Tb}(t_i)$ and defined by $L = \text{Tb}(t_i)/\tau$. We illustrate this process in Figure 6,. In the evaluation presented in part 5, we will compare the two following values of τ: 32 and 8[2].

4.5. Transition probabilities $p_{trans}(s'_i|s_i)$

Since the states s_i represent the times t_i of the successive beat numbers b_k, the distance between successive states must be close to the local tempo period $\text{Tb}(t_i)$. The transition probability models the tolerated departure of the distance between successive beat-markers from the local tempo. The model used for the probability to transit from state s_i to state s'_i is a Gaussian function with $\mu = \text{Tb}(t_i)$ and $\sigma = 0.02$s or 0.05s evaluated at $\Delta = t'_i - t_i$. Also, considering that the states are ordered in increasing time, it is not possible to transit from a state s_i to a state s'_i with $i' \leq i$. This makes our model a Left-Right HMM.

4.6. Decoding

The decoding then consists in finding the best succession of states s_i over beat-numbers b_k given $p_{init}(s_i)$, $p_{obs}(o(t)|s_i)$ and $p_{trans}(s'_i|s_i)$.

We note τ_k the time t_i associated to the most-likely ending state s_i for a forward path going until step b_k. We stop the forward algorithm when τ_k reaches the end of the music track. In the usual Viterbi algorithm, the decoding occurs over the time axis of the signal, which length is known before the decoding. Hence the various possible decoding paths over the states have all the same predetermined length. The final path is found by using the backward algorithm starting from the most-likely ending state.

Modified backward algorithm: In our reverse Viterbi decoding formulation, the last decoded hidden states (at the end of the music track) can be a time t_i in a silent part (the end of the files can be a silence period after the music) which is not a beat. In other words, we do not know which the best ending state is. We therefore modified the backtracking algorithm as follows. Instead of computing a single backward path, we compute all the backward paths for all the b_k with τ_k close to the end of the track. Since these various paths can have different (but close) lengths, we normalize the log-likelihood of each path by its length before comparing them. We finally choose the path with has the highest normalized log-likelihood. This path attributes to each beat number b_k the best state s_i, hence the best time t_i, hence the best beat locations. We illustrate the decoding algorithm in Figure 7.

Memory consideration: Given that a 4 minutes track leads to the definition of over 4800 states, hence a 4800*4800 transition matrix, memory consideration has to be taken into account when implementing the above mentioned algorithm. Because of the Left-Right nature of the HMM and because of the definition of the states (states are times), most transitions are equal (or close to zero) in the transition matrix. Therefore, the whole transition matrix does not need to be stored. One can use for example sparse matrices. Another optimization concerns the number of comparisons for the forward algorithm. In order to reduce the computation time, we only consider the states in a time-corridor around the current state.

5. EVALUATION

5.1. Evaluation measures

The evaluation performed here only concerns the quality of the estimation of the beat-tracking algorithms. However, because the time-variable tempo and meter used here, are estimations coming

[1]Defining a state s_i for each value of the onset-energy-function $f(t)$ (sampling rate of 172Hz) would lead to 40.000 states for a 4 minutes track

[2]Note that too small values of τ (hence large temporal horizon) leads to reassign several states s_i to the same time (since the successive horizons overlap), while too large values of τ (hence small temporal horizon) can lead to the miss-detection of the real beat locations (since the horizons do not overlap anymore)

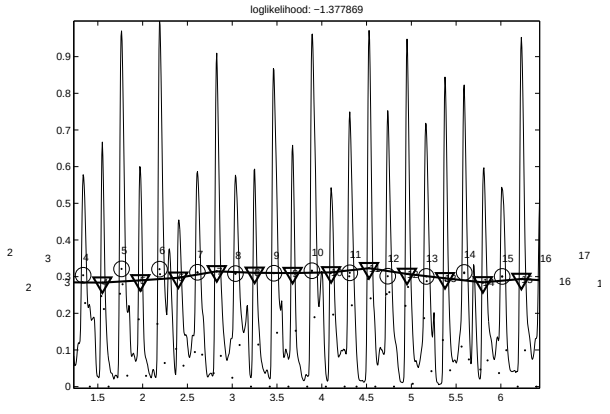

Figure 7: *Viterbi decoding and backtracking: onset-energy-function (continuous thin line), states s_i and associated observation probability (dots), maximum observation probability of each b_k (O sign), best path (continuous thick line and $\bigtriangledown$ sign).*

from the algorithm of [8], the results obtained also depend on the quality of these estimations. What we measure is therefore the performances of the whole system[3]. It is important to note that the same estimation of tempo and meter is used for both the P-sola and the Viterbi algorithm; hence the comparison between the P-sola and Viterbi algorithms is possible.

We propose here a set of measures to evaluate the performances of a beat-tracking algorithm. Considering a given beat-marker annotation and a given track, we note - A the number of annotated beats, - D the number of detected beats and - CD(PW) the number of correctly detected beats within a given Precision Window (PW). From this we derive the following measures:

- Recall(PW) = CD(PW) / A
- Precision(PW) = CD(PW) / D
- FMeasure(PW) = 2 R(PW) P(PW) / (R(PW)+P(PW))

Note that the Precision Window is centered on the annotated beat for the Recall and on the estimated beat for the Precision. For a correct beat marking but at twice (three time) the tempo (tatum marking), the Recall will be 1 but the Precision 0.5 (0.33). For a correct beat marking at half (one third of) the tempo, the Precision will be 1 but the Recall 0.5 (0.33).

In our evaluation the Precision Window depends on the local tempo. This is done in order to avoid drawing misleading conclusion from the results [4]. The Precision Window is defined as a percentage of the local annotated beat length Tb: PW=α means that the estimated beat should be at a maximum distance of $\pm\alpha$Tb the annotated beat ($\alpha = 0$ considers only exact estimations, $\alpha = 0.5$ considers the counter-beat estimations as correct[5]).

For a given track, the considered value of Tb is the minimum value of Tb(t_i) over time (the fastest annotated local tempo of the track). The values given in the "table of results" correspond to the average (over all tracks of a test-set) of the Recall(PW=0.1), Precision(PW=0.1) and F-measure(PW=0.1).

We have also computed for each test-set the average (over all tracks of a test-set) curve of the F-measure versus Precision-Window. This curve indicates the influence of the PW on the F-measure. As a summary of this curve we give the Area Under this Curve (AUC). Given that the maximum considered PW is 0.5, the maximum possible value of the AUC is also 0.5. This is illustrated in the left part of Figure 8.

Since this curve only represents average (over the tracks) values of the F-measure (it does not represent the spread over the tracks), we also provide two other measures. For this we compute the histogram of the values of the F-measure(PW=0.1) for all the track of a given test-set. This histogram indicates the percentage of tracks having a specific F-measure(PW=0.1). This is illustrated in the right-top part of Figure 8. From this histogram we compute a cumulated-histogram. This cumulated-histogram indicates the percentage of tracks having "at least" a specific F-measure(PW=0.1). This is illustrated in the right-bottom part of Figure 8. From this cumulated histogram we derive the two following measures:

1. percentage of tracks with F-measure(PW=0.1)$\geq 50\%$,
2. Area Under Curve (AUC) of the cumulated-histogram.

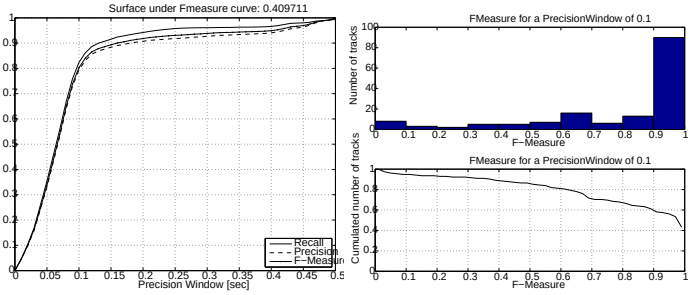

Figure 8: *[Left part] Average curves of Recall(PW), Precision(PW) and F-measure(PW) versus Precision Window; [Right-top part] Histogram of the F-measure(PW=0.1) values; [Right-bottom part] Cumulated histogram of the F-measure(PW=0.1) values; for the "PopRock extract" test-set and P-sola algorithm.*

5.2. Test-set

For the evaluation, we have used the following four test-sets. The "PopRock extract" is a collection of 155 major top-ten hits of the past decades. Only 20s extract of the tracks are considered. The annotations have been made by the author. The "RWC Popular Music" [13] is a collection of 100 tracks in full-duration of Pop-rock-ballad-heavy-metal popular music. The "RWC Jazz Music" [13] is a collection of 50 tracks in full-duration of Jazz-music with solo piano, guitar, small ensemble or modern-jazz orchestra. The difficulty of this test-set comes from the complexity of the rhythms used in Jazz-music. The "RWC Classical Music" [13] is a collection of 59 tracks in full-duration of Classical-music. The difficulty of this test-set comes from the tempo variations used in Classical-music. The annotations of the three RWC test-sets are provided by the AIST [15].

[3] The performances of the tempo and meter estimation algorithm of [8] have been thoroughly evaluated in [8] and in the MIREX-2005 contest [14].

[4] Indeed a fixed PW of 0.166s would be restrictive for slow tempi (half-beat duration of 0.5 at 60bpm) but will mean accepting counter-beat as correct for fast tempi (half-beat duration of 0.166s at 180bpm).

[5] In case of binary subdivision of the beat.

5.3. Beat-templates comparison

Before evaluating the beat-tracking, we first validate the assumption that LDA-trained beat-templates provide a better discrimination between the "beat" and "non-beat" classes than usual beat-templates. The usual beat-template considered here is composed of values of 1 at the beat-positions and 0 otherwise (as in [3]).

In order to check this assumption, we compute the values of the correlation between $f(t)$ and $g(t)$ when using the LDA-trained or the usual beat-templates for $g(t)$. From the correlation values, we then compute the ratio r of the Between-Class-Inertia to the Total-Inertia (the larger this ratio is, the best the separation is between the two classes beat and non-beat). In Figure 9, we give as example the histogram of the correlation values when using as training-set and test-set the RWC-Popular-Music (note that the y-axis for the class "beat" has been reversed - negative y-values - for better visualization). A larger separation is observed when using the LDA. For this example, we obtain the following ratio: $r_{LDA} = 0.73$ and $r_{usual} = 0.54$. We now test the generability of our approach: training on a specific set A and testing on a different set B. In Table 1, we indicate the various ratios r obtained. The lower rows of the table gives for comparison the value r obtained with the usual beat-template (which is independent of the training set). In all cases, even when $A \neq B$, the discrimination is better when using an LDA-trained beat-template. On average (over the test-sets), the most generalizable LDA-trained beat-template is the one trained on the RWC-Jazz-Music.

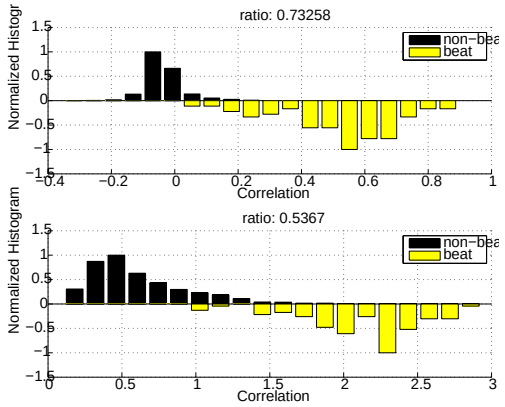

Figure 9: *Histogram of the values of the correlation using [top] LDA-trained beat-templates $g(n)$ [bottom] Usual beat-templates for the two-classes "beat" and "non-beat".*

		Test-set			
		PopRock	RWC-Popular	RWC-Jazz	RWC-Classical
Training	PopRock	**0.71**	0.71	0.53	0.37
	RWC-Popular	0.69	**0.73**	0.53	0.38
	RWC-Jazz	0.64	0.66	**0.61**	0.45
	RWC-Classical	0.60	0.64	0.56	**0.49**
	Normal template	0.48	0.54	0.35	0.23

Table 1: *Cross-database evaluation of the LDA-trained beat template in comparison with the usual beat-template. Each cell represent the value of the ratio r.*

5.4. Results and discussion

The results of the beat-marking evaluation are summarized in Table 2. In this table we compare the results obtained with the P-sola based algorithm and with several versions of the Viterbi algorithm. For the Viterbi algorithm, three parameters must be fixed: - τ: the correlation horizon for the reassignment of the time of the states, - σ: the standard deviation for the transition probability between states, - the choice of the beat-template. In Table 2, we show the results obtained with the following parameters: $\tau = 32$ and $\tau = 8$, $\sigma = 0.02$ and $\sigma = 0.05$. The choices of the beat-template are: a) "LDA shared": a beat-template created by manually analyzing the shared properties of the various LDA-trained beat-templates over test-sets, b) "LDA sam" a sampling at the sixteen notes of the LDA beat-template trained specifically for each test-set, c) "LDA all" an interpolation of the whole LDA beat-template ($N = 64$) trained specifically for each test-set, d) "usual": the "usual" beat-template with value of 1 on-beats and 0 otherwise.

	Method	Tau	Sigma	Beat-template	Recall PW=0.1	Precision PW=0.1	F-Meas. PW=0.1	AUC FMeas/PW	%Track F-Meas(PW=0.1)>0	AUC Percent/ Cumul FMeas
Poprock	P-sola				,91	,88	,89	,44	,92	,88
	Viterbi	32	0.05	LDA-shared	,93	,90	,91	,44	,96	,90
	Viterbi	8	0.05	LDA-shared	,94	,91	,92	,45	,96	,91
	Viterbi	8	0.02	LDA-shared	,94	,91	,91	,45	,96	,90
	Viterbi	8	0.05	LDA-sam	**,95**	**,92**	**,93**	**,45**	**,96**	**,91**
	Viterbi	8	0.05	LDA-all	,94	,91	,92	,45	,96	,91
	Viterbi	8	0.05	Usual	,94	,90	,91	,44	,95	,90
Popular	P-sola				,78	,73	,75	,38	,81	,74
	Viterbi	32	0.05	LDA-shared	,87	,83	,84	,42	,90	,83
	Viterbi	8	0.05	LDA-shared	,88	,83	,85	,42	,91	,84
	Viterbi	8	0.02	LDA-shared	,88	,84	,85	,42	,91	,84
	Viterbi	8	0.05	LDA-sam	**,88**	**,83**	**,85**	**,42**	**,91**	**,84**
	Viterbi	8	0.05	LDA-all	,88	,83	,84	,42	,89	,84
	Viterbi	8	0.05	Usual	,88	,84	,85	,42	,90	,85
Jazz	P-sola				,51	,42	,45	,30	,36	,33
	Viterbi	32	0.05	LDA-shared	,64	,53	,57	,33	,60	,47
	Viterbi	8	0.05	LDA-shared	,64	,53	,57	,33	,56	,48
	Viterbi	8	0.02	LDA-shared	,65	,54	,58	,33	,60	,50
	Viterbi	8	0.05	LDA-sam	,63	,52	,56	,33	,60	,47
	Viterbi	8	0.05	LDA-all	,64	,53	,57	,33	,62	,49
	Viterbi	8	0.05	Usual	**,66**	**,55**	**,59**	**,34**	**,68**	**,53**
Classical	P-sola				,48	,35	,38	,25	,25	,33
	Viterbi	32	0.05	LDA-shared	,52	,36	,41	,26	,42	,36
	Viterbi	8	0.05	LDA-shared	,53	,37	,42	,27	,42	,38
	Viterbi	8	0.02	LDA-shared	,51	,37	,41	,27	,36	,31
	Viterbi	8	0.05	LDA-sam	,52	,38	,41	,27	,42	,39
	Viterbi	8	0.05	LDA-all	,52	,36	,40	,26	,41	,38
	Viterbi	8	0.05	Usual	**,54**	**,38**	**,43**	**,27**	**,42**	**,41**

Table 2: *Comparison of the P-sola and Viterbi based beat-tracking algorithms on the four test-sets.*

Variations among test-set: The performances are best for the PopRock extract (FMeas=0.93) and RWC-Popular-Music (FMeas=0.85) test-sets than for the more complex Jazz rhythms (FMeas=0.59) or the time-variable tempi of Classical music (FMeas=0.43).

P-sola against Viterbi: Considering all criteria (all the columns of the table) and all test-sets, the Viterbi method leads systematically to better results than the P-sola one. In particular, the improvements of the F-Measure(PW=0.1) for the test-sets RWC-Popular-Music (from FMeas=0.75 to 0.85), RWC-Jazz-Music (0.45 to 0.59) and Classical Music (0.38 to 0.43) are large. Considering that the values given in the table are only estimates of the average F-measure, we perform a set of statistical tests (Student

T-test) in order to decide on the statistical significance of these differences. For this we test the H0 hypothesis that the average F-measure(PW=0.1) are equal for the P-sola and Viterbi ($\tau = 8$ and $\sigma = 0.05$) algorithms against the H1 hypothesis that they are different. The results of the tests are that for the test-sets RWC Popular-Music and RWC Popular-Jazz we can reject the null hypothesis at a 5% significance level, i.e. there is a statistical significance: the results are better with the Viterbi algorithm.

Best parameters for the Viterbi algorithm: All the results obtained with the Viterbi approach are pretty close. On average (over the test-sets), a slight improvement is obtained when using the following parameters $\tau = 8$ and $\sigma = 0.05$. This means that using a larger horizon for state reassignment ($\tau = 8$) and allowing more marker-discontinuities ($\sigma = 0.05$) helps the algorithm. Despite the results obtained in part 5.3, the larger discrimination obtained with the LDA-trained beat-templates seems of few uses for the final beat-tracking problem. All beat-template methods give very close results except for the Jazz-Music and Classical-Music where, surprisingly, the usual beat-template performs slightly better (from Fmeas=0.57 to 0.59 and from 0.4 to 0.43). This disappointing result must be taken with care since the differences are not statistically significant.

Discussions: The use of the proposed Viterbi method allows to improve the beat-tracking estimation for all test-sets. Considering the difficulty of beat-tracking for Jazz and Classical music, this result is particularly important. The Recall and Precision values obtained for the Jazz (R=0.66 and P=0.55) and Classical (R=0.54 and P=0.38) test-sets indicates that a large part of the errors are insertions errors. This is representative of an estimation of twice the correct tempo (which was considered as an error in this study). The use of LDA-trained beat-templates (over usual beat-templates) allows to slightly improve the results for the PopRock extract test-set. However, this is not the case for the Jazz and Classical test-sets. This can be explained by the fact that using LDA-trained beat-templates somehow assumes tracks with a specific constant rhythm pattern. This is usually the case for pop-rock music but surely not for Jazz and Classical music. Moreover for Classical music, the main problem comes from rapid tempo changes (this problem is partly solved using the proposed Viterbi method) rather than ambiguities of rhythms.

6. CONCLUSION AND FUTURE WORKS

In this paper we have proposed two approaches for the beat-tracking problem given time-variable tempo and meter as input: a P-sola approach and a Viterbi approach. For the second approach we have proposed to use a machine-learning method, the Linear Discriminant Analysis, in order to estimate the best beat-template. Measures of performances have been proposed and a large-scale evaluation performed. In all cases, the results obtained using the Viterbi approach were better than with the P-sola approach. A statistical significance at 95% between the two methods has been obtained for two test-sets over four. Concerning the choice of the best parameters for the Viterbi approach, no statistically significant differences have been observed. While the use of LDA-trained beat-templates allows a better discrimination between the "beat" and "non-beat" classes (whatever the training-set and the test-set used), their use in the framework of beat-tracking do not change the performances of the system. This point will be the subject of future works. The performances given in this study were obtained by evaluating the whole tempo, meter and beat-tracking system. Considering the octave errors in our tempo estimation for the Jazz and Classical music test-sets, future work will concentrate on evaluating the performance of the beat-tracking algorithm alone using the exact annotated tempo and meter as input. Although the computation time and the memory cost of the Viterbi method is higher than for the P-sola method, this approach is particularly promising since, apart from the better performances, the framework can easily be extended. Future works will therefore concentrate in adding new types of observations probabilities in order to allow distinguishing the role of the various beat-numbers and hence the downbeats among the succession of beat-numbers.

7. REFERENCES

[1] MIREX, "Audio beat tracking contest," 2006.

[2] Alan Marsden, *Journal of New Music Research: Special Issue on Beat and Tempo Extraction*, 2007.

[3] J. Laroche, "Efficient tempo and beat tracking in audio recordings," *J. Audio Eng. Soc.*, vol. 51, no. 4, pp. 226–233, 2003.

[4] A. Klapuri, A. Eronen, and J. Astola, "Analysis of the meter of acoustic musical signals," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 14, no. 1, pp. 342–355, 2006.

[5] D. Ellis, "Beat tracking by dynamic programming," *J. New Music Research*, vol. 6, no. Special Issue on Beat and Tempo Extraction, pp. 51–60, 2007.

[6] S. Dixon, "Automatic extraction of tempo and beat from expressive performances," *Journal of New Music Research*, vol. 30, no. 1, pp. 39–58, 2001.

[7] M. Goto, "An audio-based real-time beat tracking system for music with or without drum-sounds," *Journal of New Music Research*, vol. 30, no. 2, pp. 159–171, 2001.

[8] G. Peeters, "Template-based estimation of time-varying tempo," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, pp. Article ID 67215, 14 pages, 2007.

[9] G. Peeters, "Music pitch representation by periodicity measures based on combined temporal and spectral representations," in *Proc. of IEEE ICASSP*, Toulouse, France, 2006, vol. V, pp. 53–56.

[10] G. Peeters, *Modeles et modelisation du signal sonore adaptes a ses caracteristiques locales*, Phd thesis, Universite Paris VI, 2001.

[11] L. Rabiner, "A tutorial on hidden markov model and selected applications in speech," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–285, 1989.

[12] R. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, pp. 179–188, 1936.

[13] M. Goto, H/ Hashiguchi, T. Nishimura, and R. Oka, "Rwc music database: Popular, classical, and jazz music databases," in *Proc. of ISMIR*, Paris, France, 2002, pp. pp. 287–288.

[14] G. Peeters, "Mirex 2005: Tempo detection and beat marking for perceptual tempo induction," in *Proc. of ISMIR*, London, UK, 2005.

[15] M. Goto, "Aist annotation for the rwc music database," in *Proc. of ISMIR*, Victoria, Canada, 2006, pp. pp.359–360.