Homework 4

Tempo estimation and beat tracking

Li Su
Research Center for IT Innovation, Academia, Taiwan
lisu@citi.sinica.edu.tw

In this homework we will implement algorithms for the following tasks: (1) compute the tempo given a song, (2) identify every beat position given a song and (3) Get familiar with the evaluation of these tasks in the MIR field.

Prerequisite:

(1) Download the Ballroom dataset and annotation from:
http://mtg.upf.edu/ismir2004/contest/tempoContest/node5.html
(2) For MATLAB users, refer to the Tempogram Toolbox from:
http://resources.mpi-inf.mpg.de/MIR/tempogramtoolbox/
(3) For Python users, refer to the following link:
https://bmcfee.github.io/librosa/generated/librosa.feature.tempogram.html
(4) Or you could also implement the tempograms by yourself (see following):

Steps of computing the tempogram:

Step 1: Computing the *novelty curve* of the input audio signal. In this assignment we consider the *spectral flux*, which is one of the most commonly used novelty curves. Given a STFT representation $X(n,k) \in R^{K\times N}$ where n refers to the time index and k refers to the frequency index. The spectral flux is represented as

$$\Delta_{\mathrm{Spectral}}(n) := \sum_{k=0}^{K} |X(n+1,k) - X(n,k)|_{\geq 0}.$$

$|\cdot|_{\geq 0}$ is the *half-wave rectification function*

$$|r|_{\geq 0} := \frac{r+|r|}{2} = \begin{cases} r, & if\ r \geq 0 \\ 0, & if\ r < 0 \end{cases}$$

See the MATLAB code attached for more implementation details.

Step 2: The Fourier tempogram $F(n,\tau)$ is just the STFT of the novelty curve. Notice that the unit of τ is in BPM.

Step 3: The autocorrelation (ACF) tempogram $A(n,\tau)$ is the time-varying ACF of the novelty function. A lazy way in implementing the ACF is using Wiener-Khinchin theorem, which states that the Fourier transform of the power spectrum is the ACF. More specifically, for a Fourier tempogram $F(n,\tau)$, the ACF tempogram is

$$A(n,\tau) = \mathrm{STFT}^{-1}(|F(n,\tau)|^2)$$

where STFT^{-1} is the inverse STFT. Please refer to the lecture slides for details.

Notice: always be clear about the *timestamp* and the *feature rate* you are using. Take a short-time windowed signal segmented from 10.0 second to 10.1 second (window size = 0.1 second) as an example. In this case, the computed STFT of this frame is at 10.05 second (the average of 10.0 and 10.1). And, if you take the spectral difference of two consecutive frame spaced by 0.01 second (this happens when you let the hop size of STFT be 0.01 second), then the feature rate is 100 Hz (i.e. 100 features per second).

Step of computing the predominant local pulse (PLP) curve for beat tracking:

Step 1: Finding the local phase value of the predominant tempo around T_1 (and T_2) in the Fourier tempogram:

$$\tau_n := \operatorname{argmax}_{\tau\in[\mathrm{T}_1-\delta, T_1+\delta]} F(n,\tau)$$

$$\phi_n = \frac{1}{2\pi}\arccos\left(\frac{Re\left(F(n,\frac{\tau_n}{60})\right)}{\left|F(n,\frac{\tau_n}{60})\right|}\right)$$

Notice that we should convert the unit of tempo frequency, BPM, to Hz, in order to correctly compute the phase value.

Step 2: Reconstructing the local sinusoid representing the beat positions.

$$\kappa_n(m) := w(m-n)\cos\left(2\pi\left(\frac{\tau_n}{60}\cdot m - \phi_n\right)\right)$$

Step 3: The beat sequence of the song is obtained by overlap-adding those local sinusoids. See lecture slides for details.

$$\Gamma(m) = \left|\sum_{n\in Z}\kappa_n(m)\right|_{\geq 0}$$

Q1 (30%): Evaluate your tempo estimation algorithm on the Ballroom dataset using the Fourier tempogram. Your algorithm should generate two predominant tempo values, T_1 (the slower one) and T_2 (the faster one). Then you also need to compute a "relative saliency of T_1" defined by the strength of T_1 relative to T_2. It is to say, for the Fourier tempogram, we have the saliency $S_1 = F(n,T_1)/(F(n,T_1)+F(n,T_2))$ for a specific time at *n*. For an excerpt with ground-truth tempo *G*, the *P-score* of the excerpt is defined as

$$P = S_1 T_{t1} + (1 - S_1) T_{t2}$$

$$T_{ti} = \begin{cases} 1 & \text{if } \left|\frac{G - T_i}{G}\right| \leq 0.08 \\ 0 & \text{otherwise} \end{cases}, i = 1,2$$

Another score function is the “at least one tempo correct” (ALOTC) score, defined as

$$P = \begin{cases} 1 & \text{if } \left|\frac{G - T_1}{G}\right| \leq 0.08 \text{ or } \left|\frac{G - T_2}{G}\right| \leq 0.08 \\ 0 & \text{otherwise} \end{cases}$$

Compute the average P-scores and the ALOTC scores of the eight genres (Cha Cha, Jive, Quickstep, Rumba, Samba, Tango, Viennese Waltz and Slow Waltz) in the Ballroom dataset using your algorithm. (Hint: the ground-truth tempo G of each excerpt could be obtained from the labeled beat sequence in the dataset. Given a beat sequence $\mathbf{b} = [\mathrm{b}_1, \mathrm{b}_2, \ldots, \mathrm{b}_\mathrm{M}]$, the average tempo (in BPM) could be represented as $\mathrm{mean}(60/\mathrm{diff}(\mathbf{b}))$.)

Q2 (10%): In addition to the “relative saliency” of T_1 and T_2, the ratio T_2/T_1 might be also informative. Comparing T_2/T_1, T_1/Q, T_2/Q for the genres, what do you see? Is there any musical meaning?

Q3: (10%) Instead of using your estimated $[T_1, T_2]$ in evaluation, try to use $[T_1/2, T_2/2]$, $[T_1/3, T_2/3]$, and $[T_1/4, T_2/4]$. What are the resulting P-scores? Discuss the result.

Q4 (25%): Using the ACF tempogram and repeat Q1 and Q2. What do you see? Compare the result with Q1 and Q2.

Q5 (10%): Instead of using your estimated $[T_1, T_2]$ in evaluation, try to use $[T_1/2, T_2/2]$, $[T_1/3, T_2/3]$, and $[T_1/4, T_2/4]$. Or, try to use $[2T_1, 2T_2]$, $[3T_1, 3T_2]$, and $[4T_1, 4T_2]$. What are the resulting P-scores? Discuss the result.

Q6 (15%): From the above discussion, do you have any idea in improving the current algorithms using either the Fourier tempogram (Q1) or the ACF tempogram (Q4)? Please propose one tempo estimation algorithm that outperforms the current ones. (Hint: you may modify the definition of the weighting factor S_1, or find some ways in combining the Fourier tempogram and the ACF tempogram.)

Bonus: Evaluate your beat tracking algorithm on the Ballroom dataset. The F-score is defined as $F := 2PR/(P + R)$, with Precision, P, and Recall, R, being computed from the number of correctly detected onsets N_{tp}, the number of false alarms N_{fp}, and the number of missed onsets N_{fn}, where $P \coloneqq N_{tp}/(N_{tp}+N_{fp})$ and $R \coloneqq N_{tp}/(N_{tp}+N_{fn})$. Here, a detected beat is considered a true positive when it is located within a tolerance of ± 70 ms around the ground truth annotation. If there are more than one detected beat in this tolerance window, only one is counted as true positive, the others are counted as false alarms. If a detected onset is within the tolerance window of two annotations one true positive and one false negative are counted. Similarly, please compute the average F-scores of Cha Cha and Slow Waltz in the Ballroom dataset.

Please send your zip file containing the report and your codes, with email title "HW4 [your ID]" to lisu@citi.sinica.edu.tw.

The deadline for this homework is May 29 (Sun), and we will discuss it on June 2.