Homework 2

Tempo estimation, beat/downbeat tracking, and meter recognition of audio and symbolic data

Li Su
Institute of Information Science, Academia, Taiwan
lisu@iis.sinica.edu.tw

In this homework we will implement algorithms for the following tasks: (1) compute the tempo of a song, (2) identify every beat/downbeat position of a song, and (3) identify the meters of a song.

The definitions of beat and downbeat have been mentioned in the course slides. *Meter* refers to the regularity of repeating patterns of music. In a narrow sense, meter here refers to the relationship between beats and bars. For example, time signature 3/4 means that each bar contains 3 beats, and each beat is a quarter note; therefore, its meter is 3-beats. The commonly seen meters in our everyday music are 3-beats, 4-beats or their multiples, while 5-beats and 7-beats are also used sometimes.

Prerequisite:

(1) Download the Ballroom dataset and annotation from:
https://drive.google.com/drive/folders/1ZeFeCu4tBfdKw1Wqv0TINxgXmAf6CM08
(2) Download the SMC beat tracking dataset and annotation:
http://smc.inesctec.pt/research/data-2/
(3) Download the JCS beat and downbeat tracking dataset and annotation:
https://drive.google.com/drive/folders/18OP9LU8YflZtkULOk7qLAZkdBY8cOQfn
(4) Download the madmom library, the state-of-the-art Python library for beat and downbeat tracking:
https://madmom.readthedocs.io/en/latest/
(5) There are also Python resources available for tempo and beat computation:
https://librosa.org/doc/main/generated/librosa.feature.tempogram.html#librosa.feature.tempogram
https://librosa.org/doc/main/generated/librosa.beat.tempo.html#librosa.beat.tempo
https://librosa.org/doc/main/generated/librosa.beat.beat_track.html#librosa.beat.beat_track
https://librosa.org/doc/main/generated/librosa.tempo_frequencies.html#librosa.tempo_frequencies
https://librosa.org/doc/main/generated/librosa.feature.fourier_tempogram.html#librosa.feature.fourier_tempogram
https://librosa.org/doc/main/generated/librosa.fourier_tempo_frequencies.html#librosa.fourier_tempo_frequencies
(6) Or you could also implement the tempograms by yourself (see the lecture slides).

Task 1: tempo estimation

Q1 (20%): Design an algorithm that estimate the tempo of each clip in the Ballroom dataset. Assume that the tempo of every clip is constant. Note that your algorithm should output two predominant tempi for each clip: T_1 (the slower one) and T_2 (the faster one). For example, you may simply try the two largest peak values in the tempogram over the whole clip. Please compare and discuss the results computed from the Fourier tempogram (`librosa.feature.fourier_tempogram`) and the autocorrelation tempogram (`librosa.feature.tempogram`).

The evaluation metrics of tempo estimation is as follows. We 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 tempogram $F(t,n)$, we have the saliency $S_1 = F(T_1,n)/(F(T_1,n)+F(T_2,n))$ for tempo value t at a specific time at n. For an excerpt with ground-truth tempo G, the *P-score* of the excerpt is defined as

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

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

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

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

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 algorithms. The above process can all be found in the evaluation routine `mir_eval.tempo.detection`.

Note 1: if you want to use `librosa.beat.tempo` directly, you have to find some ways to let it output two tempi.
Note 2: please compare the results of both the Fourier and the autocorrelation tempogram.
Note 3: the ground-truth tempo G of each excerpt could be obtained from the labeled beat sequence in

the dataset. Given a beat sequence $\mathbf{b} = [\mathrm{b}_1, \mathrm{b}_2, \ldots, \mathrm{b}_M]$, the average tempo (in BPM) could be represented as $\mathrm{mean}(60/\mathrm{diff}(\mathbf{b}))$.)

Q2 (20%): Instead of using your estimated $[T_1, T_2]$ in evaluation, try to use $[T_1/2, T_2/2]$, $[T_1/3, T_2/3]$, $[2T_1, 2T_2]$, and $[3T_1, 3T_2]$ for estimation. What are the resulting P-scores? Also, please compare and discuss the results using the Fourier tempogram (`librosa.feature.fourier_tempogram`) and the autocorrelation tempogram (`librosa.feature.tempogram`).

Q3 (20%): The window length is also an important factor in tempo estimation. Try to use 4s, 6s, 8s, 10s, 12s instead of default window length (8.9s). Compare the ALOTC of eight genres.

Task 2: using dynamic programming for beat tracking

Q4 (20%): Using `librosa.beat.beat_track` to find the beat positions of a song. Evaluate this beat tracking algorithm on the Ballroom dataset. The F-score of beat tracking is defined as $F \coloneqq 2PR/(P+R)$, with Precision, *P*, and Recall, *R*, being computed from the number of correctly detected onsets *TP*, the number of false alarms *FP*, and the number of missed onsets *FN*, where $P \coloneqq TP/(TP+FP)$ and $R \coloneqq TP/(TP+FN)$. 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, then one true positive and one false negative will be counted. This process can be done with `mir_eval.beat`. Similarly, please compute the average F-scores of the eight genres in the Ballroom dataset and discuss the results.

Q5 (20%) Also use this algorithm on the SMC dataset and the JCS dataset. Compare and discuss the results together with the results of the Ballroom dataset. Could you explain the difference in performance?

Note: since these datasets are larger than the ones used in Homework 1, please run these datasets as early as possible in order not to be late in submitting your homework.

Task 3: meter recognition (bonus)

Q6 (20%): The meter of a song can be 2-beats, 3-beats, 4-beats, 5-beats, 6-beats, 7-beats, or others. There might be multiple meters existing in a song (e.g., a 4-beats section followed by a 3-beats section). As a task combining both beat tracking and downbeat tracking, meter recognition is still a challenging task. Could you design an algorithm to detect the instantaneous meter of a song? Test the algorithm on the clips in the JCS dataset, and report frame-wise accuracy. The 1, 2, 3, 4, 5 after every line in the annotation file is the meter annotation. You can simply use `madmom.features.beats` (the state-of-the-art beat tracker) or combine other functions mentioned above.

The deadline for this homework is June 1 (Tue). Late submission within 2 days will result in 95% of your original score, withing 5 days for 90%, within 8 days for 85%. Late submission by more than 8 days will result in an 80% score. This time you do not have to zip all files. Please name your every file as hw2_109012345.