HW3
Tempo estimation and beat tracking

Li Su

May 8, 2023

In this assignment we will need to accomplish 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. You might also need to use the following functions in `librosa`:

- `librosa.feature.fourier_tempogram`
- `librosa.feature.tempogram`
- `librosa.beat.tempo`
- `librosa.beat.beat_track`
- `librosa.tempo_frequencies`
- `librosa.fourier_tempo_frequencies`

And maybe others. Please read the documentation of librosa carefully and discover the useful functions by yourself. Also, you can choose to implement the tempograms and the related features by yourself.

1. **Task 1: tempo estimation**

   (a) Q1 (30%): Design the tempo estimation algorithm using 1) Fourier tempogram and 2) autocorrelation tempogram. 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. Design your algorithm and evaluate it on the Ballroom dataset. Please compare and discuss the results computed from the Fourier tempogram and the autocorrelation 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

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

$$T_{ti} = \begin{cases} 1, & |(G - T_i)/G| \le 0.08 \\ 0, & \text{otherwise} \end{cases} \quad \text{for } i = 1, 2 \tag{2}$$

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

$$P_{\text{ALOTC}} = \begin{cases} 1, & |(G - T_1)/G| \leq 0.08 \text{ or } |(G - T_2)/G| \leq 0.08 \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

Compute the average P-scores and the ALOTC scores of the ISMIR2004 dataset and 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 `mir_eval.tempo.detection` directly, you have to find some ways to let it output two tempi.)

(b) Q2 (20%): The window length is also an important factor in tempo estimation. Try to use 4s, 8s, 12s of window lengths for both Fourier tempogram and the autocorrelation tempogram and compare the ALOTC of the eight genres in the Ballroom dataset.

(c) Q3 (20%): Could you design a tempo estimation algorithm which utilizes the advantages of Fourier tempogram and autocorrelation tempogram to achieve better performance? You might consider the early fusion and the late fusion approaches we mentioned in HW #1.

2. Task 2: using dynamic programming for beat tracking

   Q4 (30%): 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 $F := 2PR/(P+R)$, 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 $P = TP/(TP + FP)$ and $R = TP/(TP + FN)$. 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.

3. Task 3: meter recognition (bonus)

   Q5: The meter of a song can be 2-beats, 3-beats, 4-beats, or others. As a task combining both beat tracking and downbeat tracking, meter recognition is a challenging task. Could you design an algorithm to detect the meter of a song? Test the algorithm on the Ballroom dataset, and report the accuracy.

The Ballroom dataset is available at:
https://drive.google.com/drive/folders/1ZeFeCu4tBfdKw1Wqv0TINxgXmAf6CM08?usp=share_link. Please submit your .zip file containing the report and your code, with the file name "HW2_[your ID]" to the course website. The deadline of Assignment #3 is May 30th.