AUDIO TEMPO ESTIMATION USING FUSION OF TIME-FREQUENCY ANALYSES AND METRICAL STRUCTURE

Elio Quinton
Queen Mary University of London

Christopher Harte
Queen Mary University of London

Mark Sandler
Queen Mary University of London

ABSTRACT

This paper presents an audio tempo estimation system submitted to MIREX 2014. Musical onsets are represented by a novelty curve generated using a spectral difference method. Two time-frequency analyses of this novelty curve are then carried out in parallel, one using the Fourier transform and the other an autocorrelation function (ACF), resulting in the computation of two rhythmograms which are then fused. Metrical level rates representing aspects of the meter are extracted. A Technique based on perceived tempo resonance model is then employed to select two of them to be the two tempi.

1. INTRODUCTION

The MIREX Audio Tempo Estimation test set consists of 140 music tracks of various genre, instrumentation, tempo and meter. These tracks are 30 seconds long and provide a stable tempo value. As a consequence the algorithm presented here will assume a consistent tempo value throughout the track.

The dataset is annotated with the *perceived* tempo; data having been gathered by asking a group of listeners to tap along with each track. Ambiguity can arise because listeners may tap at different rates [4]. However, these rates are not uncorrelated; they correspond to different metrical levels of the music. As a result, at least in the case of western music, they are usually related to each other in integer ratios. A well known problem in tempo extraction, the *octave error*, is also related to the metrical structure of the music and caused by a similar ambiguity between several closely related metrical levels. This usually manifests as errors of a factor of two or three, typically corresponding to duple and triple meters respectively. In order to take this into account, the annotation data for each track has two tempo values, corresponding to two distinct metrical levels.

The algorithm presented here estimates the two tempo values from the metrical structure of the music. Firstly the metrical levels are extracted, without any prior knowledge, from the fusion of two rhythmograms into a single rhythm spectrum. The result is subsequently filtered following a set of rules so that only two levels remain.

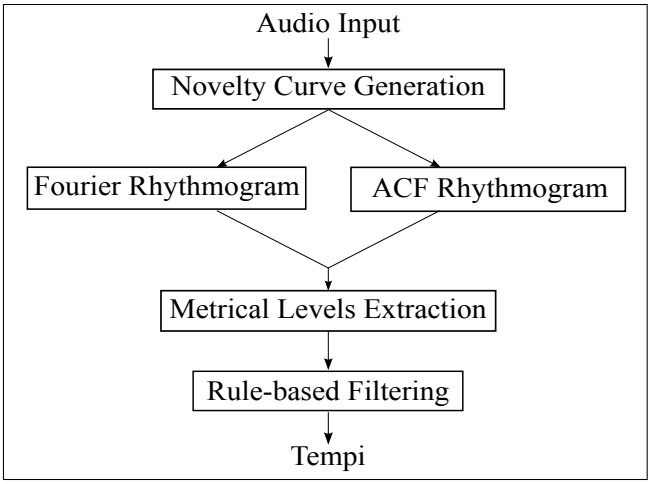

Figure 1. Audio tempo estimation algorithm flowchart

2. PRE PROCESSING AND TIME-FREQUENCY ANALYSIS

The algorithm comprises a number of processing blocks organised as shown in Figure 1. The novelty curve is obtained using a spectral difference method [1] implemented using the corresponding module in the *tempogram toolbox* released by Grosche [3].

Then two time frequency analyses of the novelty curve are performed in parallel. On the one hand, a framed Fourier transform is applied to generate a first rhythmogram $\mathcal{R}_F(t, f)$, where t represents time and f frequency. Long windows (12 seconds) are used in order to retain a good frequency resolution. On the other hand, a framed autocorrelation of the novelty curve is computed in order to produce a second rhythmogram $\mathcal{R}_A(t, f)$. Again, 12s long windows are used and the range of time lags corresponds to frequencies ranging from 20 to 800 BPM.

In the context of this MIREX challenge, the tempi have to be processed across a whole track so we average the rhythmograms $\mathcal{R}_F(t, f)$ and $\mathcal{R}_A(t, f)$ over time. To obtain the average rhythmic spectrum for each rhythmogram we calculate

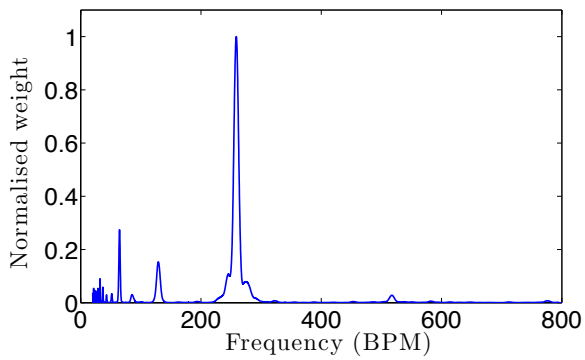

Figure 2. Spectrum $S(f)$ obtained from the fusion of averaged Fourier and ACF rhythmograms for track *train1.wav* from the MIREX train set

$$\Omega_X(f) = \sum_t \mathcal{R}_X(t, f), \tag{1}$$

where X denotes which rhythmogram (F or A) is being averaged.

Fourier rhythmograms reveal harmonics of the periodicities present in the novelty curve whereas ACF rhythmograms reveal sub-harmonics of these same periodicities [2]. Multiplying the spectra $\Omega_F(f)$ and $\Omega_A(f)$ therefore allows us to keep only the common periodicities, the most salient of which correlate with the metrical rates present in the music [5]. Following this rationale, to produce a fused rhythmic spectrum vector $S(f)$ we calculate the Hadamard product [1] of the spectra $\Omega_F(f)$ and $\Omega_A(f)$ and normalise the result:

$$S(f) = \frac{\Big(\Omega_A(f) \circ \Omega_F(f)\Big)}{\max_f \Big(\Omega_A(f) \circ \Omega_F(f)\Big)} \tag{2}$$

Figure 2 is the fused spectrum obtained with this processing for the track *train1.wav* from the MIREX train set. The metrical levels are extracted by pick peaking the spectrum $S(f)$ and selecting the peaks that best represent the metrical hierarchy of the track via a process that is not detailed in this paper because of its limited length.

Each metrical level of frequency f_i is associated with a weight w_i from spectrum $w_i = S(f_i)$. Consequently, the weight is related to the energy of that frequency in the time-frequency transform of the musical signal. Because they receive a lot of energy from the musical signal, frequencies with large weight are considered as important pulse rates of the music.

3. TEMPO ESTIMATION

The final step of processing is to filter the metrical levels obtained at the previous stage so that only two of them are retained as tempo estimates.

The filter uses the resonance curve parametrized to fit the perceived tempo tapped by listeners in [4]. The curve tapers at its extremities, which accounts for the unlikeliness of listeners to tap tempo typically below 50 BPM and above 200 BPM. As a consequence, the tempi are limited to the 30-230 BPM range in the present algorithm. The selection of the two successful candidates among metrical levels extracted at the previous stage is achieved using the following logic: First of all, the number of metrical levels present in range 30-230 BPM is calculated. Secondly, depending on that number, the appropriate processing is performed. Four cases can be anticipated:

- There are exactly two metrical levels in the 30-230 BPM range: they are chosen as the tempo estimates.
- There are more than two metrical levels in the 30-230 BPM range: The level with the heaviest weight and the heaviest first adjacent level are chosen as the two estimates.
- There is only one level in the 30-230 BPM range: Two candidates are generated from this metrical level. One with double frequency, one with half frequency. Both are weighted using the normalised resonance curve which equation is given on page 3 of [4], and the heaviest one is chosen as the second estimate.
- There is no metrical level present in the 30-230 BPM range: The heaviest metrical level present outside the range is taken as a reference and its frequency is divided by two if it is above the higher bound of the range until two candidates in the range are found. The converse processing is applied in the reference metrical level is below the lower bound of the range. Tempo estimates are then both weighted using the normalised curve on page 3 of [4].

It has to be noted that the two latter cases are implemented to enable recovery from any possible failure of the previous processing stage (i.e. that would fail at detecting some metrical levels). In that type of scenario, the information required to infer the tempo values is missing and is therefore 'guessed' at the expense of a major bias. The factor two (rather than three for instance) is chosen because popular western music tends to use duple meters more than triple. Most musical pieces are expected to fall under one of the two first conditions.

[1] An element by element multiplication denoted as $\circ$

4. REFERENCES

[1] Juan Pablo Bello, Laurent Daudet, Samer Abdallah, Chris Duxbury, Mike Davies, and Mark B. Sandler. A tutorial on onset detection in music signals. *Speech and Audio Processing, IEEE Transactions on*, 13(5):10351047, 2005.

[2] Peter Grosche, M. Muller, and Frank Kurth. Cyclic tempograma mid-level tempo representation for musicsignals. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, page 55225525. IEEE, 2010.

[3] Peter Grosche and Meinard Muller. Tempogram toolbox: Matlab implementations for tempo and pulse analysis of music recordings. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR), Miami, FL, USA*, 2011.

[4] M. Mckinney and Dirk Moelants. Deviations from the resonance theory of tempo induction. *Abstracts of the Conference on Interdisciplinary Musicology (Full text included on CD-rom)*, pages 124–125, 2004.

[5] Geoffroy Peeters. Time variable tempo detection and beat marking. In *Proceedings of the ICMC*, 2005.