EFFICIENT, CLASSIFICATION-ASSISTED TEMPO ESTIMATION

Hendrik Schreiber
tagtraum industries incorporated
hs@tagtraum.com

ABSTRACT

This extended abstract details a submission to the Music Information Retrieval Evaluation eXchange (MIREX) 2013 for the Audio Tempo Estimation task. We submitted an implementation of a simple and efficient tempo estimator augmented by an equally simple tempo classifier for octave error correction. We briefly summarize the algorithm.

1. INTRODUCTION

Probably the biggest problem in current tempo estimation is the so-called *octave error*, i.e., the halving or doubling of the perceived tempo. Current algorithms are generally reliable when ignoring the octave error and achieve more than 90% accuracy. When not ignoring the tempo octave, accuracy decreases to roughly 65% [4]. Therefore the problem of tempo estimation can be broken down into three tasks: Computing the dominant tempi while largely ignoring the tempo octave, then determining the perceived tempo, and finally combining the two results in a meaningful way.

In this extended abstract we will briefly describe how we approached all three sub-tasks, starting with the dominant tempi estimation in Section 2, continuing with a coarse tempo classification in Section 3, and tying the results together using a set of rules in Section 4.

The submitted code was implemented using the open source audio feature extraction framework *jipes* [3].

2. ESTIMATING DOMINANT TEMPI

To estimate the tempo we first convert the signal to mono with a sample rate of 11025Hz. Then we compute the spectra $X(t)$ of 93ms long windows with $1/2$ overlap, by first applying a Hamming window and then performing an FFT. Each resulting power spectrum $X(t)$ is split in two bands: X_L with frequencies $30 - 184\text{Hz}$ and X_H with frequencies $184 - 5512.5\text{Hz}$. The power for each bin k at time t is given by $X(t,k)$. As an indicator for onsets $O(t)$ we then compute the sum of the logarithmic powers in each band for each window using Eq. 1 and 2.

$$I(t,k) = \begin{cases} 1 & \text{if } X(t,k) > X(t-1,k), \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

$$O(t) = \sum_k \log_{10}\left(X(t,k) - X(t-1,k) + 1\right) \cdot I(t,k) \quad (2)$$

To reduce the computational burden, $O(t)$ is decimated by a factor of 2. Subsequently, it is transformed using a FFT with length 8192. This length ensures a resolution of 0.079 BPM. However, for 8192 values of O, we would need more than six minutes of audio. Therefore, for shorter signals, we zeropad the FFT input at the end.

The peaks of the resulting beat spectrum B represent the strength of BPM values present in the signal. They do not, however, take into account the fact that a 60 BPM peak usually implies a 30 BPM peak (assuming a duple meter). To make up for this shortcoming, we create two derived spectra. B_D for duple meters and B_T for triple meters, both are described in Equation 3.

Similar to computing a spectral sum [1], B_D models duple meters by simply adding to each bin the magnitudes of the bins denoted by half and quarter of its own frequency. Note, that for very low frequencies $\leq 1/3$ Hz we use the mean magnitude $\overline{|B|}$ rather than the in this particular range rather meaningless $B(k)$. Correspondingly, B_T is modeled by adding to each bin the magnitude of the bin with a third of its frequency and—to allow for a direct comparison with B_D—the mean magnitude $\overline{|B|}$.

$$\begin{aligned} B_D(k) &= |B(k)| \\ &+ |B(\lfloor k/2 + 0.5 \rfloor)| \\ &+ |B(\lfloor k/4 + 0.5 \rfloor)| \\ B_T(k) &= |B(k)| \\ &+ |B(\lfloor k/3 + 0.5 \rfloor)| \\ &+ \overline{|B|} \end{aligned} \quad (3)$$

By not adding magnitudes over all integer fractions of a given frequency, but only those which correspond to duple or triple meter, we effectively obtain two different models, each supporting a meter hypothesis. For further processing we pick the one with the greater maximum peak p (Eq. 4). From it we extract the BPM value for p and its strength denoted by s (Eq. 5).

$$p = \max_k(B(k)) \quad (4)$$

$$s(p) = \frac{p - \overline{|B|}}{\overline{|B|}} \quad (5)$$

Because we are performing these last steps for both frequency bands represented by X_L and X_H spectra, we obtain two corresponding BPM candidates C_L and C_H, each with an indicator of strength s, and a meter classification (duple or triple).

3. TEMPO CLASSIFICATION

Goal of the coarse tempo classification is to determine whether a song's tempo is perceived as slow, fast, or medium. This classification is supposed to allow us to pick the right BPM candidate and/or adjust it according to the estimated listener perception.

Experiments using Last.FM labels as ground truth have shown a remarkable correlation between the tags *slow* and *fast* and the mean spectral novelty (SNM) as proposed in [2]. To compute the mean spectral novelty, we are reusing the already computed spectra $X(t)$ to build a self-similarity matrix using the cosine of the angle between different $X(t)$ as similarity score. For calculating the novelty score we use a 92×92 Gaussian checkerboard kernel. With the given sample rate and window overlap, this is equivalent to a 4.3s kernel. To obtain SNM we simply average all obtained novelty scores $N(t)$ as defined in [2].

$$\text{SNM} = \overline{N(t)} \quad (6)$$

In above mentioned experiment, we observed that the tempo class $T \in \{slow, medium, fast\}$ is related to SNM as described in Eq. 7.

$$T(\text{SNM}) = \begin{cases} slow & \text{if SNM} > 53.4, \\ fast & \text{if SNM} < 37.4, \\ medium & \text{otherwise} \end{cases} \quad (7)$$

Thus we obtain a simple tempo classification T.

4. RULE-BASED OCTAVE CORRECTION

One or both of the BPM candidates C may be too high or too low for their associated tempo class T. Therefore we define a BPM validity interval $I(T)$ for each tempo class (Eq. 8) with $\tau = 91$ as the pivot between slow and fast, $\alpha = 40$ as the lower boundary, and $\beta = 230$ as the upper boundary.

$$I(T) = \begin{cases} [\alpha, \tau] & \text{if } T = slow, \\ [\tau, \beta] & \text{if } T = fast, \\ [\tau - 42, \tau + 42] & \text{if } T = medium \end{cases} \quad (8)$$

If BPM candidate $C \notin I(T(C))$, i.e., it does not fall into the validity interval of its tempo class, it is either increased or decreased until it does. If the candidate stems from a duple meter model B_D, the used decrease/increase factor is $m = 2$, for triple meters it is $m = 3$. We use the candidates' strength $s(C)$ to determine the stronger of the two candidates and call it C_1', the weaker C_2'.

If the weaker candidate differs less than 8% from the stronger, or the ratio between the two is not roughly ⅓, ⅔, ½, ¾, or ¼, the weaker candidate is dropped, and we derive a new second BPM value C_2'' by increasing or decreasing the remaining candidate by its meter factor m (Eq. 9).

$$C_2'' = \begin{cases} \frac{C_1'}{m} & \text{if } C_1' > \tau, \\ C_1' m & \text{otherwise} \end{cases} \quad (9)$$

In the end, we arrive at two BPM candidates, C_1' and either C_2' or C_2''.

5. MIREX 2013 RESULTS AND DISCUSSION

TODO

6. CONCLUSION

7. REFERENCES

[1] M. Alonso, B. David, and G. Richard. A study of tempo tracking algorithms from polyphonic music signals. In *In 4-th COST 276 Workshop*, 2003.

[2] Jonathan Foote. Automatic audio segmentation using a measure of audio novelty. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, pages 452–455, New York, NY, USA, 2000.

[3] tagtraum industries incorporated. jipes. Website `http://www.tagtraum.com/jipes/`, last accessed 9/8/2013.

[4] G Tzanetakis and G Percival. An effective, simple tempo estimation method based on self-similarity and regularity. In *ICASSP*, 2013.