COMBINATION OF AUDIO TEMPO ESTIMATION APPROACHES (MIREX 2011 SUBMISSION)

Jose R. Zapata, Emilia Gómez
Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain
{joser.zapata,emilia.gomez}@upf.edu

ABSTRACT

This paper describes a tempo estimation algorithm submitted to the MIREX 2011, and is an extension of the work presented in [2]. Using the tempo estimation results from four different approaches (BeatIt, Ellis, Davies and MPEG7-XM), we use a heuristic strategy to obtain the slow and fast tempo from audio music combinig their results, based on the hypothesis that the tatum and tactus tempo hierarchal levels can be obtained from these estimations analyzing their relations.

1. INTRODUCTION

This tempo Combination algorithm is based on the analysis and the combination method in [2], in this work we observed that for more than the 90% of the song-excerpts of the used music collection (consisting on 465 songs), the metrical level can be estimated with the results of all the algorithms and the estimations of the best approach can be improve with a heuristic combination of the results of the other methods. Based on this hypothesis and the best available approaches for tempo estimation and beattracking (evaluated in [2]) we built a MATLAB script to combine the results and obtain the slow, fast tempo and their relative strength for the MIREX Audio Tempo Estimation Task.

2. APPROACHES

The combination method proposed in this paper uses all the tempo estimations obtained from four different approaches: BeatIt, Ellis, Davies and MPEG7-XM. A general description of each of the methods are presented. All the methods are used with default configuration parameters.

2.1 BeatIt

In 2006, Jordi Bonada and Fabien Gouyon from Music Technology Group, Universitat Pompeu Fabra, proposed an inhouse approach called BeatIt. It is a C++ implementation of a beat tracking algorithm. The input signal is split into several frequency bands. For each band, the energy is computed, compressed, and differentiated. Next, the peak-to-peak distances between the maximum peaks of the autocorrelation function of each band are computed and stored. Those are added to a histogram of one BPM octave. The maximum of the histogram (the tatum, fastest metrical level) sets the wrapped BPM estimation. Some statistics on the peak-to-peak distribution are used to select the output BPM octave. The output is a single tempo estimation from the pulse induction block.

2.2 Ellis

Ellis.from Columbia University, proposed an approach for tempo estimation [1] implemented in Matlab [1]. The algorithm computes an onset energy envelop obtained from a 40 mel-frequency spectrogram in the feature list block, and for the Pulse induction block an autocorrelation function is computed over the onset envelop to obtain the periodicity peaks. The output of the pulse induction block of this algorithm is given as a MIREX audio tempo estimation task (a slower tempo, T1, a faster tempo, T2, and the strength of T1 relative to T2).

2.3 Davies

Davies and Landone from the Centre for Digital Music Queen Mary, University of London, proposed an algorithm for beat tracking which consists on a hybrid of the two-state beat tracking model [3] and a dynamic programming method [1]. for the feature list creation uses and onset likelihood function, estimate the tempo contour and, then, given the tempo, to recover the beat locations. The induction estimation is based on autocorrelation, and this signal is weighted by a filter-bank and grouped together into a matrix of observations of periodicity through time. The best path of periodic-

[1] http://labrosa.ee.columbia.edu/projects/beattrack/tempo2.m

ity is found using the Viterbi algorithm. Given the estimates of periodicity, the beat locations are recovered by applying the dynamic programming algorithm. the algorithm is a matlab script and Its output consists on Beat locations, so the IBI (Inter-Beat-interval) is computed and different BPM values are obtained from the calculation.

2.4 MPEG7-XM

Jan Rohden from Fraunhofer Institute for Digital Media Technology IDMT, wrote this algorithm along MPEG output document w5212 (15938-4:2001/FPDAM) in Matlab. The algorithm starts by extracting energy using the same process proposed by Scheirer. It considers 6 frequency bands in segments of 4 sec. each. The envelopes obtained for each band are then weighted and the pulse induction is based on autocorrelation based periodicity detection via forward and inverse fft (biased Autocorrelation). The source code and the algorithm description can be downloaded at [2]. The output is a BPM value each times the estimated tempo changes.

3. COMBINATION METHOD

Based on the hypothesis [2]that the tatum and tactus tempo hierarchal levels can be obtained from the tempo estimations of all these approaches, we built a MATLAB script and used a heuristic strategy to analyze the relations between the estimations, and obtain the slow and the fast tempo from a audio song. Combination steps:

1. Sort all the tempo estimation values and eliminate the repeated ones.

2. Cluster tempo values with differences of 4% (e.g: 127.6, 125.8 , 128.4) and calculated the median value of each cluster.

3. Check if each value has a relation of ($\frac{1}{2}$, 2, $\frac{1}{3}$ or 3) and eliminated the not related ones.

4. Heuristic analysis

    - If only two values are obtained, the lowest value is the slow tempo (T1) and the highest value is the fast tempo (T2). The strength of T1 relative to T2 is taked from the Ellis Results.
    - If there are 3 values with a binary relation between them, the lowest value is the slow tempo (T1)and the double value is the fast tempo(T2).The strength of T1 relative to T2 is taked from the Ellis Results.
    - If there are 3 values and two of these ones had a ternary relation between them we take this two numbers and the lowest value is the slow tempo (T1) and the double value is the fast tempo(T2). The strength of T1 relative to T2 is taked from the Ellis Results.
    - If there are more than 3 values we choose the values related with the BeatIt estimation, and the heuristic analysis is checked, to obtain the slow tempo (T1), the fast tempo(T2) and the strength of T1 relative to T2 is taked from the Ellis Results.

5. Save in a file the slow tempo(T1), the fast tempo(T2) and the strength of T1 relative to T2 values.

4. EVALUATION RESULTS

The results in the MIREX 2011: Audio Tempo Extraction - MIREX06 Dataset [3] are presented in Table 1 this table is sorted by Tempo P-score value of each algorithm, our algorithm called ZG1 has the second best results in the both tempi correct value (0.5714) and it's better than the other algorithms presented in the past year on MIREX 2010: Audio Tempo Extraction [4] (Table 2).

Table 1. *Results MIREX 2011: Audio Tempo Extraction - MIREX06 Dataset*

ALGORITHM	Tempo P-Score	At least One Tempo Correct	Both Tempi Correct
GKC3	0.8290	0.9429	0.6214
FW2	0.7385	0.8357	0.5429
ZG1	**0.7275**	**0.8214**	**0.5714**
SP1	0.7105	0.9286	0.3857
GKC6	0.6777	0.8214	0.4286
SB5	0.6559	0.8429	0.3500

The results of GKC3 [5] are the best results on Audio Tempo Extraction in MIREX06 Dataset (2006, 2010, 2011).

Table 2. *Results MIREX 2010: Audio Tempo Extraction - MIREX06 Dataset*

ALGORITHM	Tempo P-Score	At least One Tempo Correct	Both Tempi Correct
GKC1	0.8099	0.9643	0.5000
NW2	0.7875	0.9143	0.5000
ES1	0.7714	0.9071	0.5500
TL1	0.7639	0.8929	0.4786
BES2	0.7429	0.9143	0.4857
OL1	0.6679	0.8786	0.3786
GT1	0.6150	0.6929	0.5071

[2] http://mpeg7.doc.gold.ac.uk/mirror/v2/Matlab-XM/AudioBpmD/AudioBpmD.m

[3] http://nema.lis.illinois.edu/nema_out/mirex2011/results/ate/

[4] http://nema.lis.illinois.edu/nema_out/mirex2010/results/ate/

[5] http://www.music-ir.org/mirex/abstracts/2011/GKC3.pdf

The Combination method uses the Ellis and Davie algorithms presented in MIREX 2006: Audio Tempo Extraction in the same dataset (Table 3) [6] and the results of ZG1 are better in Tempo P-Score and the both tempi correct value compared with both algorithms. and the ZG1 algorithm had better result in at least one tempo correct value compared with the ellis result.

Table 3. *Results MIREX 2006: Audio Tempo Extraction - MIREX06 Dataset*

ALGORITHM	Tempo P-Score	At least One Tempo Correct	Both Tempi Correct
klapuri	0.806	0.9429	0.6143
davies	0.776	0.9286	0.4571
alonso	0.7242	0.8929	0.4357
alonso	0.6931	0.8571	0.4571
ellis	0.673	0.7929	0.4286
antonopoulos	0.669	0.8429	0.4786
brossier	0.62	0.7857	0.5071

5. CONCLUSIONS

The Combination method ZG1 had better results than ellis and davies algorithms alone in the Audio Tempo Extraction task in the MIREX06 Dataset, this result could be improve using the results of other algorithms.

The ZG1 algorithm only analyze the binary and ternary relations between the algorithms results, the results in the Audio tempo extraction task could be improve searching for odd or changing meter.

6. ACKNOWLEDGMENTS

Thanks to Dan Ellis, Matthew Davies and Jan Rohden authors of the methods for making their codes or executables available for this research project. Jordi Bonada and Fabien Gouyon for Beatit. Thanks to Universidad Pontificia Bolivariana (Colombia),Colciencias, Music Technology Group at Universitat Pompeu Fabra , Classical Planet and DRIMS project for the financial support.

7. REFERENCES

[1] D. Ellis: "Beat Tracking by Dynamic Programming," *Journal of New Music Research*, Vol. 36, No. 1, pp. 51–60, 2007.

[2] J. Zapata and E. Gómez: "Comparative Evaluation and Combination of Audio Tempo Estimation Approaches," *Audio Engineering Society Conference: 42nd International Conference: Semantic Audio*, Ilmenau - Germany, 2011.

[3] M.E.P. Davies, and M. D. Plumbley: "Context-dependent beat tracking of musical audio," *Audio, Speech,and Language Processing, IEEE Transactions*, Vol. 15, No. 3, pp. 1009–1020, 2007.

[6] http://www.music-ir.org/mirex/wiki/2006:Audio_Tempo_Extraction_Results