THE AUBIO LIBRARY AT MIREX 2006

Paul M. Brossier
Centre for Digital Music
Queen Mary University of London
Mile End Road, London
piem@altern.org

ABSTRACT

For the 2006 edition of the Music Information Retrieval Evaluation eXchange (MIREX), we presented the latest enhancements to the aubio library, participating this year in four different tasks: Audio Melody Extraction, Audio Tempo Extraction, Audio Beat Tracking and Audio Onset Detection. The algorithms we submitted are tailored for causal operation and interactive systems. We describe here the outline of these methods with pointers to the relevant references. The evolution of the algorithm performances from last year results is discussed, and this year's results are compared to that of other systems.

Keywords: MIREX, aubio, onset, melody, tempo.

1 INTRODUCTION

In (Brossier, 2006), we describe our investigations on the automatic annotation of musical signals for interactive systems. This study includes the evaluation of several methods for onset detection, pitch extraction and beat tracking. Different algorithms were implemented to work in real time and evaluated on large databases of hand annotated data.

Our implementation, the aubio library, was designed to use these annotation routines in interactive systems. The library, along with the programs we submitted to the 2006 edition of MIREX, is available under the GNU General Public License (Brossier, 2003).

2 ALGORITHMS

We give here a brief description of the algorithms we submitted for the different tasks of MIREX. For a detailed description of these methods, see (Brossier, 2006).

2.1 Onset detection

In (Bello et al., 2005), a number onset detection methods were reviewed, several of which based on a phase vocoder. For our causal implementation, we have implemented different onset detection methods with several modifications for real time operations (Brossier et al., 2004).

The onset detection algorithms consist of a phase vocoder, a detection function built from consecutive spectral frames obtained from the phase vocoder, and a peak picking algorithm tailored to find peaks in the detection function corresponding to actual onsets and within a short delay. A silence gate is also used to avoid spurious detections in areas of low energy.

We submitted four different functions, each to be run with different pick peaking threshold values: high frequency content (Masri, 1996), spectral difference function (Foote and Uchihashi, 2001), complex domain function (Duxbury et al., 2003), and a dual function, built using both HFC and Kullback Liebler function (Hainsworth and Macleod, 2003). A detailed study of this implementation is given in (Brossier, 2006, Chap. 2).

2.2 Beat tracking

The algorithm submitted for beat tracking was first described in (Davies and Plumbley, 2004), with further improvements for its real-time implementation in aubio in (Davies et al., 2005). The approach makes use of an onset detection function built on a phase vocoder, similar to that used in Section 2.1. A modified autocorrelation of the onset detection function is then computed to determine the beat period and phase alignment. Based on the detected period and phase, beats are predicted for the following frames, so that the beats can be tapped along live audio streams. A detailed study of this implementation is given in (Brossier, 2006, Chap. 4).

2.3 Tempo extraction

The method we use for the extraction of tempo is based on beat tracking algorithm described in Section 2.2. The MIREX evaluation metrics used for tempo extraction requires two different tempo values and their corresponding phases. The primary tempo values is computed by our algorithm as the median of all beat periods found in the file. In order to produce the two tempi output required by the evaluation metrics, we use a set of simple rules: if the primary tempo period is greater than 95 beats per minute (BPM), the secondary tempo period is set to half the primary period; if instead the primary tempo period is found smaller than 95 BPM, the secondary period is set to twice the primary period. The value of 95 was determined empirically based on the training set provided for this task.

2.4 Melody extraction

The algorithm we have submitted to the melody extraction task is designed for fundamental frequency estimation. The method is derived from the YIN algorithm (de Cheveigné and Kawahara, 2002), and was detailed in (Brossier, 2006, Chap. 3). A major modification to the YIN algorithm is the use of a tapered squared difference function to build the cumulative normalised sum. This modification was found to facilitate the selection of the fundamental period and to increase the robustness of the method in the presence of noise and other instruments.

3 SOFTWARE IMPLEMENTATION

The aubio library consists in a collection of C routines. The programs submitted for MIREX make use of the Python external for aubio. For code availability, see (Brossier, 2003).

Using the Python interface is convenient for rapid prototyping and debugging. However, using Python causes an increase of the run times of the algorithms, since the Python environment has be loaded and unloaded at each run, unlike for their C equivalent. The actual computation times required for each algorithms were measured in (Brossier, 2006), where detailed comparisons of the run times obtained for different onset and pitch detection methods are given.

4 RESULTS DISCUSSION

4.1 Onset detection

The Precision-Recall graphs used to display the onset detection results give an interesting comparison of the performance of the different algorithms on different classes of sounds. These graphs highlight the different performances achieved by different methods on different sound classes. The ranking of the algorithms are indeed very dependant of the type of sounds, suggesting that the final ranking is very dependant on the overall distribution in the database.

Only minor changes were applied to the onset detection routines since our submission to the 2005 edition of MIREX, so that the optimal results we obtain are very similar to that of last year. The study of the threshold parameter, with values in the range 0.1 to 0.9, demonstrates the ability of our adaptive thresholding approach to move from under-segmentation to over-segmentation, regardless of the function we use.

4.2 Beat tracking

For its first edition, the beat tracking task has seen five different submissions. The P-score evaluation measure gave results in the range 0.407 to 0.391, where our algorithm ranked last. Given the small differences between submissions, it would be interesting to have more detailed results. However, our algorithm is not designed to extract beat locations at the very beginning of the file, and the P-score we obtained is coherent with the one of M. E. Davies submission, which interpolates beat locations at the beginning of the file, and which ranked 4$^{\text{th}}$ with a P-score of 0.394.

The aubio submission obtained the shortest run time amongst submissions, with 139 seconds to run through the test set, while other submissions were executed in between 498 seconds and 1394 seconds.

4.3 Tempo extraction

With an average of 78.57% of files annotated with the correct tempo period, results showed that our algorithm actually performed worst than last year (80.71%), which seems to indicate that parameters are over fitted to the test data set. However, the empirical rule we used to determine the second tempo value seem fairly efficient, with 50.71% of the files annotated with both correct tempi, whereas M. E. Davies submission detected both correct tempi in 45.71% of the files.

4.4 Melody extraction

The results we obtained for this year MIREX edition are encouraging, since no post-processing was done on the pitch track, and the output of the pitch detection were used directly for evaluation. As we have no mechanism to decide whether or not a frame is voiced, the voicing false alarm rate is not relevant for our algorithm.

ACKNOWLEDGEMENTS

Many thanks to the MIREX organisation team for another successful edition of MIREX, and especially to Andreas F. Ehmann for pointing out several issues in the initial submissions.

References

Juan-Pablo Bello, Laurent Daudet, Samer Abdallah, Christopher Duxbury, Mike P. Davies, and Mark B. Sandler. A tutorial on onset detection in music signals. *IEEE Transactions in Speech and Audio Processing*, 13 (5):1035–1047, September 2005.

Paul M. Brossier. Aubio, a library for audio labelling. `http://aubio.piem.org`, 2003. Last visited: Fri Octobre 6 2006.

Paul M. Brossier. *Automatic annotations of musical audio for interactives systems*. PhD thesis, Centre for Digital Music, Queen Mary University of London, London, UK, 2006. submitted for publication.

Paul M. Brossier, Juan-Pablo Bello, and Mark D. Plumbley. Real-time temporal segmentation of note objects in music signals. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 458–461, Miami, Florida, USA, November 2004.

Matthew E. P. Davies and Mark D. Plumbley. Causal tempo tracking of audio. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR)*, pages 164–169, Barcelona, Spain, October 2004.

Matthew E. P. Davies, Paul M. Brossier, and Mark D. Plumbley. Beat tracking towards automatic musical accompaniment. In *Proceedings of the Audio Engeenirng Society 118th Convention*, Barcelona, Spain, May 2005.

Alain de Cheveigné and Hideki Kawahara. YIN, a fundamental frequency estimator for speech and music. *Journal of the Acoustical Society of America*, 111(4):1917–1930, 2002.

Christopher Duxbury, Mike E. Davies, and Mark B. Sandler. Complex domain onset detection for musical signals. In *Proceedings of the International Conference on Digital Audio Effects (DAFx-03)*, pages 90–93, London, UK, 2003.

Jonhatan Foote and Shingo Uchihashi. The beat spectrum: a new approach to rhythm analysis. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2001)*, pages 881–884, Tokyo, Japan, August 2001.

Stephen Hainsworth and Malcom Macleod. Onset detection in music audio signals. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 163–166, Singapore, 2003.

Paul Masri. *Computer modeling of Sound for Transformation and Synthesis of Musical Signal*. PhD dissertation, University of Bristol, UK, 1996.