David Sears
MUMT 621
11/11/09
Audio Beat Tracking

Introduction

Beat tracking in audio is the task of identifying and synchronizing with the basic pulse of a piece of music (Dixon 2007). Although the feat of tapping one's foot in time to music is an intuitive and often unconscious human response, the aim of recovering a sequence of beat onset times using machine learning techniques still falls short of human beat tracking ability (Davies 2007). In this report I will outline the goals of a beat tracking system, discuss an early beat tracking system developed by Simon Dixon, called BeatRoot, and then provide an example of a current beat tracking system developed by Matthew Davies and Mark Plumbley. I will conclude with a few examples of some of the analytical applications available that use beat tracking systems.

Goals

Davies et al (2007) suggest 4 main desirable properties for a beat tracker:

1. Both audio and symbolic musical signals can be processed
2. No *a priori* knowledge of the input is required
3. Perceptually accurate beat locations can be identified efficiently and in real-time
4. Changes in tempo can be followed, thereby indicating the rate of change of tempo

Although many of the current systems possess one or more of these properties, no system to date has managed to provide all of them.

Simon Dixon's *BeatRoot*

BeatRoot, first presented in 2001, tracks the beat in music in a two-stage process: a tempo induction process, which finds the rate of the beats, and a beat tracking process, which synchronizes the pulse sequence with the music. Before either stage can begin, however, *BeatRoot* preprocesses the input audio file with an onset detection function. Initial iterations of *BeatRoot* used the "surfboard" method, which involves smoothing the signal to produce an amplitude envelope before finding peaks in its slope using linear regression (Dixon 2003). However, because this method is necessarily lossy (it fails to detect onsets in which simultaneously sounding notes mask one another), the current iteration of *BeatRoot* employs a spectral flux onset detection function, which sums the change in magnitude in each frequency bin where the energy is increasing. Peak picking is then performed using a set of empirically determined constraints.

Using the calculated onset times from preprocessing, the tempo induction algorithm computes clusters of inter-onset intervals and then combines them by recognizing the approximate integer relationships between clusters (Dixon 2007). In the final stage, beat tracking is performed using a tempo hypothesis derived from the tempo induction stage and the first onset of a given window is used to then predict further beats. Subsequent onsets that fall within the prediction window are treated as beats by *BeatRoot*, while those falling outside the window are taken to be *possible* beats, or disregarded altogether.

At the MIREX 2006 beat tracking contest, *BeatRoot* outperformed all its competitors. As a beat tracking system, *BeatRoot* is also highly accessible, as Dixon designed it with a GUI interface. Though *BeatRoot* did not fare as well at MIREX 2009, it is still the most accessible system to date, and its ability to track the beat in the face of extreme temporal deviation (Romantic piano music) separates it from its peers.

Context-Dependent Beat Tracking

Much like *BeatRoot*, the system developed by Matthew Davies *et al* consists of a preprocessing stage and a two-stage model. The preprocessing stage uses a complex spectral difference function to detect onsets, in which note onsets are emphasized either as a result of significant change in energy in the magnitude spectrum, and/or a deviation from the expected phase values in the phase spectrum (ie a change in pitch). The first processing stage, the *General State*, attempts to infer the time between successive beats, the beat period, and the offset between the start of the frame and the locations of the beats, the beat alignment. The *General State* operates in a memoryless fashion, which causes two errors: 1)the switching of metrical levels, and 2) switching between the on and the off-beat (problems both seen in *BeatRoot*). To account for these errors, the *Context Dependent* state operates in a similar way to the General State, but incorporates prior knowledge of the tempo of the input in order to maintain IOI continuity. Finally, the system employs both states simultaneously in a two-stage model, since the Context Dependent state is unable to react to tempo changes as they occur. The two-stage model runs one state as the active state while the other state continues determining the beat independently and passively, so that in the event that the active state gets out of phase, the passive state can pick up the thread.

Davies *et al* evaluated their system and determined that it performs favorably with the other beat trackers in the research community, and its computational efficiency is much better than its peers. Indeed, at the MIREX 2009 contest it finished 2nd in beat tracking the McKinney collection, and 1st for Sapp's Mazurka Collection.

Analytical Applications

Using *BeatRoot*, Widmer *et al* (2003) considered how individual composers varied expressive timing information during the performance of the First Eight Bars of *Von Fremden Landern und Menschen* (*Kinderszenen,* op. 15, by Robert Schumann). Automatic extraction of expressive timing information can therefore provide a means for studying individual differences in performance nuance, as well as provide keen insights into note-level performance rules that can be observed using machine learning techniques.

References

Davies, M. A. Robertson, and M. Plumbley. 2009. MIREX 2009 Audio beat tracking evaluation: Davies, Robertson and Plumbley. In *Proceedings of the International Society for Music Information Retrieval*.

Davies, M. E. P., and M. D. Plumbley. 2007. Context-dependent beat tracking of musical audio. *IEEE Transactions on Audio, Speech and Language Processing*, 15(3): 1009-20.

Dixon, S. 2003. On the analysis of musical expression in audio signals. Draft for *SPIE/IS&T Conference on Storage and Retrieval for Media Databases*. Online http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.15.5379&rep=rep1&type=pdf.

Dixon, S. 2007. Evaluation of the Audio Beat Tracking System BeatRoot. Preprint for *Journal of New Music Research*, 36. Online http://www.elec.qmul.ac.uk/people/simond/pub/2007/jnmr07.pdf.

Lee, T. 2009. Audio beat tracking. In *Proceedings of the International Society of Music Information Retrieval.*

Peeters, G. 2009. MIREX-09 "Audio beat tracking" task: IRCAMBEAT submission. In *Proceedings of the International Society for Music Information Retrieval.*

Tzanetakis, G. 2009. Marsyas submissions to MIREX 2009. In *Proceedings of the International Society for Music Information Retrieval*.

Widmer, G., S. Dixon, W. Goebl, E. Pampalk, and A. Tobudic. 2003. In Search of the Horowitz Factor. *AI Magazine*, 24(3): 111-29.