

# BEAT ESTIMATION ON THE BEAT

*Kristoffer Jensen and Tue Haste Andersen*

Department of Computer Science, University of Copenhagen  
Universitetsparken 1  
DK-2100 Copenhagen, Denmark  
{krist,haste}@diku.dk

## ABSTRACT

This paper presents a novel method for the estimation of beat interval, and the exact location of the beats from audio files. As a first step, a feature extracted from the waveform is used to identify note onsets. The estimated note onsets are used as input to a beat induction algorithm, where the most probable beat intervals are found. The note onsets corresponding to the beat locations are then identified. Several enhancements are proposed in this work, including methods for identifying the optimum audio feature, a novel weighting system in the beat induction algorithm and a simple robust method for identifying the beat locations. The resulting system runs in real-time, and is shown to work well for a wide variety of contemporary and popular rhythmic music.

## 1. INTRODUCTION

Beat estimation is the process of predicting the musical beat from a representation of music, symbolic or acoustic. The beat is in this work defined to represent what humans perceive as a binary regular pulse underlying the music. In western music the rhythm is divided into measures, e.g. pop music often has four beats per measure. The problem of automatically finding the rhythm include finding the time between beats, finding the time between measures, and finding the location of beats and measures. This work develops a system to find the time between beats and the location of the beat from a sampled waveform in real-time.

Beat estimation systems has many uses. Here we are primarily interested in *control* and *synchronization* of music. Specifically the beat system has been implemented in the DJ software Mixxx [1], where the tempo of the music can be controlled during playback by tapping the beat, or using a conductors baton [2]. Furthermore synchronization of the beat of two tracks is supported, either automatically or by visualization of the beat locations. For all of these applications it is of primary importance to have a reliable output, that is a beat interval measure which is slowly varying over time, rather than jumping between integer multiples of the human perceived beat.

The beat in music is often marked by transient sounds, e.g. note onsets of drums or other instruments. Some onset positions may correspond to the position of a beat, while other onsets fall *off beat*. By detecting the onsets in the acoustic signal, and using this as input to a beat induction model, the beat is estimated.

Goto and Muraoka [3] presented a beat tracking system, where two features were extracted from the audio based on the frequency band of the snare and bass drum. The features were matched against pre-stored drum patterns and resulted in a very robust system, but only applicable to a specific musical style. Later Goto and

Muraoka [4] developed a system to perform beat tracking independent of drum sounds, based on detection of chord changes. This system was not dependent on the drum sounds, but again limited to simple rhythmic structures. Scheirer [5] took another approach, by using a non-linear operation of the estimated energy of six band-pass filters as feature extraction. The result was combined in a discrete frequency analysis to find the underlying beat. The system worked well for a number of rhythms but made errors that related to a lack of high-level understanding of the music. As opposed to the approaches described so far Dixon [6] built a non-causal system, where an amplitude-based feature was used as clustering of inter-onset intervals. By evaluating the inter-onset intervals, hypotheses are formed and one is selected as the beat interval. This system also gives successful results on simpler musical structures.

There are a very large number of possible features to use in segmentation and beat estimation. The approach adopted here consists of identifying promising audio features, and subsequently evaluating the quality of the features using error measures. Many audio features were found to be appropriate in beat detection systems and one feature significantly better, although it is necessary to introduce a high-level model for beat induction from the selected audio feature. The beat induction is done using a running histogram, the beat induction histogram, which has been inspired by the work of Desain [7]. The location of the beats, finally, is estimated by a simple method that compares the weights of the peaks of the audio feature within one beat interval.

## 2. SYSTEM

Coupling the beat induction histogram with the beat location estimation results in a complete algorithm for finding the beat location and interval, as shown in figure 1. The algorithm consists of a feature extraction part, a peak detection algorithm, a beat induction histogram for finding the beat interval, and finally the beat location estimation. It is implemented in the open source DJ application Mixxx [1, 8] for automatic beat synchronization with external input devices or other sound sources. The algorithm is applicable not only to the beat synchronization of two audio files, but also synchronization of pre-recorded audio with live music, as the algorithm operates in a causal way. The system requires little CPU time and incorporates a minimum of adjustable parameters. It has been demonstrated to work well on wide variety of popular and contemporary music, including music with tempo changes or breaks.

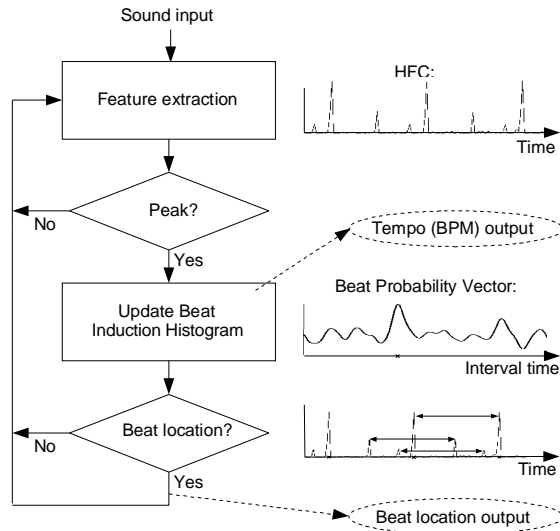


Figure 1: Block diagram of the complete system.

### 3. AUDIO FEATURE SELECTION

The basis of the beat estimation is one audio feature that responds to the transient note onsets. Many features have been introduced in research of audio segmentation and beat estimation. This includes parameters previously used in segmentation, discrimination and auditory perception research. Some of the considerations employed in the choice of the features are processing speed, real-time behavior, robustness and precision. The features considered in this work are: amplitude, spectral centroid, high frequency energy, high frequency content, spectral irregularity, spectral flux and running entropy, all of which have been found in the literature, apart from the high frequency energy and the running entropy.

In the following, a number of features are reviewed, a peak detection algorithm is described, and considerations for the selection of the optimal feature are given.

The features are all, except the running entropy, computed using the magnitude of a short time Fourier transform with a sliding Kaiser window. All the features are calculated with a fixed block and step size. More information on these features can be found in [9].

The amplitude feature is useful for percussive instruments, such as the piano or guitar. However, it is often very noisy for other instruments and for complex music. Fundamental frequency is difficult to use in complex music, since it is dependent on the estimation method. It has been used in segmentation of monophonic audio [10] with good results, though.

One of the most important timbre parameters is the spectral centroid (brightness). The spectral centroid is a measure of the relative energy at the high frequencies. Therefore it seems appropriate in transient detection, which contain relatively much high frequency energy. The spectral irregularity (SPI) and the spectral flux (SPF) are two other features known from the timbre perception research. These features react to the noise level and the transient behavior that often are indicators of beats.

An absolute measure of the energy in the high frequencies (HFE) and the frequency weighted high frequency content (HFC) are two interesting audio features because they indicate both high

energy, but also relatively much high frequency energy.

Note onsets can be considered new information in the audio file. Therefore the running entropy is also evaluated. The estimation of the probability used in the entropy calculation is error prone, though, and this feature is rather noisy.

The note-onsets generally occur in the middle of the attacks, but the features generally peak at the end of the attacks. To compensate for this delay the time derivative is taken on the features. The derivative of the amplitude has also been shown to be important in the perception of the attack location [11].

The features considered in the previous section all exhibit local maximums at most of the perceptual note onsets. To identify a note onset from a given feature a peak detection algorithm is needed. The peak detection algorithm used here chooses all local maximums, potentially using a threshold,

$$p = (F_{n-1} < F_n > F_{n+1}) \wedge (F_n \geq th) \quad (1)$$

where  $F$  is an arbitrary audio feature and  $th$  the threshold. In addition to the peak detection, a corresponding weight,  $w_k$  is also calculated at each peak  $k$ , corresponding to the time step  $t_k$  where  $p$  is true. This weight is later used in the beat induction histogram, and in the detection of the beat locations.

To compare features, different pieces of music have been analyzed manually by placing marks at every perceptual note onset. These manual marks are used in the comparison of the different features. In all eight pieces were used, with an average of 1500 found note onsets per piece. To select the optimum feature, three different error measures are used, based on matched peaks, that is peaks located within a time threshold (20 msec) to a manual mark. An unmatched peak is located outside the time threshold from a manual mark.

The error measures used are the signal to noise ratio, the missed ratio and the spurious ratio. Generally the missed ratio rises when the spurious ratio fall, and vice-versa. It is thus difficult to select the optimum audio features, and feature calculation parameters, since both low missed and spurious ratio is the optimization goal and they are mutually exclusive.

An initial analysis of the error values for all features and pieces gives no clear indication of the best feature. However, it is clear that several parameters perform poorly, in particular the amplitude, the spectral irregularity, and the entropy. The best parameters seem to be the spectral centroid, the high frequency content and the spectral flux. An alternative evaluation method, involving determining the signal to noise ratio for a predefined percentage of matched beats [9] demonstrates that the high frequency content (HFC) performs significantly better than the other features, in particular for the block sizes 2048 and 4096 samples, which has the best overall signal to noise ratio. The HFC is calculated as the sum of the amplitudes and weighted by the frequency squared,

$$HFC_n = \sum_{l=1}^{N_b/2} (a_l^n l^2) \quad (2)$$

where  $n$  is the current block,  $a_l$  is the magnitude of FFT bin  $l$ , and  $N_b$  is the block size. The block size is set to 2048 samples, and the step size is set to 1024 samples.

### 4. BEAT ESTIMATION

The analysis of the audio features has permitted the choice of feature and feature parameters. There is, however, still errors in the

detected peaks of the chosen features. As described in other beat estimation systems found in the literature, a beat induction system, that is a method for cleaning up spurious beats and introducing missing beats, is needed. This could be based on artificial neural nets, as in [10], but this method demands manual marking of a large database, potentially for each music style. Another alternative is the use of frequency analysis on the features, as in [5], but this system reacts poorly to tempo changes.

Some of the demands of a beat estimation system are stability and robustness. Stability to ensure that the estimation is yielding low errors for music exhibiting stationary beats and robustness to ensure that the estimation continues to give good results for music without stationary beats. In addition, the system should be causal, and instantaneous. Causal to ensure real-time behavior, and instantaneous to ensure fast response. Finally, a tempo range is needed to avoid the selection of beat intervals that do not occur in the music style. The tempo is chosen in this work to lie between 60 and 200 BPM.

These demands are fulfilled by the use of the memory-based beat induction histogram that is based on the model of rhythm perception by Desain [7].

#### 4.1. Beat induction histogram

The beat induction histogram is a dynamic model of the beat intervals that permits the identification of the beat intervals from noisy features. It is a histogram of note onset intervals,  $\Delta t$ , as measured from the previous note onset. For each new note onset the histogram,  $H(\Delta t)$  is updated (along with its neighboring positions) by a Gaussian shape at the intervals corresponding to the distance to the previous peak. To maintain a dynamic behavior, the beat induction histogram is scaled down and updated at each peak location,

$$H(\Delta t) = W^{t_k - t_{k-1}} H(\Delta t) + G(t_k - t_{k-1}, \Delta t), \Delta t = 0 \dots \infty \quad (3)$$

where  $W$  is the time weight that scale down the weights of the older intervals, and  $G$  is a Gaussian shape which is non-zero at a limited range centered around  $t_k - t_{k-1}$ . The current beat interval is set to the maximum in the beat induction histogram, or alternatively, to  $t_k - t_{k-1}$  if the interval is located at the vicinity of the maximum in the beat induction histogram. The memory of the beat induction histogram allows the detection of the beat interval in breaks with missing or alternative rhythmic structure.

In [7] and in earlier work [9], multiples of the intervals are also increased. Since the intervals are found from the audio file in this work, the erroneous intervals are generally not multiples of the beat. Another method must therefore be used to identify the important beat interval.

#### 4.2. Update with multiple intervals

To avoid a situation where spurious peaks create a maximum in the histogram with an interval that does not match the current beat, it is updated in a novel way. By weighting each new note and taking multiple previous note onsets into account the beat induction histogram  $H(\Delta t)$  is updated with  $N$  previous weighted intervals that lie within the allowed beat interval,

$$H(\Delta t) = H(\Delta t) + \sum_{i=1}^N w_k w_{k-i} G(t_k - t_{k-i}, \Delta t), \Delta t = 0 \dots \infty \quad (4)$$

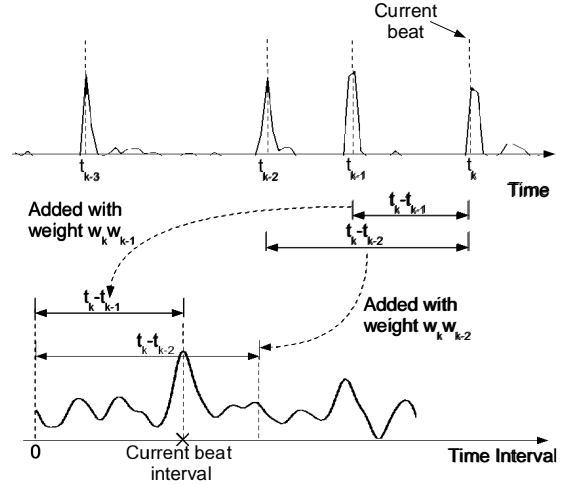


Figure 2: Selection of beats in the beat induction histogram. For each new peak, a number of previous intervals are scaled and added to the histogram. The maximum of the beat induction histogram gives the current beat interval.

where  $w_k$  is the weight of the peak  $k$ . For simplicity, the time weight  $W$  is omitted in this formula.

This simple model gives a strong indication of note boundaries at common intervals of music, which permits the identification of the current beat interval.

An illustration of the calculation of the beat induction histogram can be seen in figure 2. It consists of the estimated audio feature (top), the estimation of the current beat interval and the updating of the running beat induction histogram (bottom). The current beat interval is found as the peak interval closest to the maximum in the beat induction histogram, or at the maximum itself.

## 5. BEAT LOCATION

Even though the selected audio feature (HFC), described in section 3, gives a strong indication of the note onsets, and the beat induction histogram indicates the most probable beat interval, there are still many spurious and systematic peaks in the HFC in between the beats. This corresponds to rhythmic information (note onsets) that does not fall on the beat locations, and noise. Therefore, the beat interval may be known, but the absolute position of the beat is unknown.

In order to locate the peaks that correspond to the current beat location, the weight of the peaks are used. At every new peak, if an interval to a previous peak corresponding to the current beat interval is found then the weight of this and all the shorter peak intervals are calculated. If the current peak is the strongest in the beat interval, it is assumed to be the beat location. Otherwise it is ignored.

The estimation of the beat locations is shown in figure 3. It is clear that the correct beat onset (beat location) is larger than the other peaks in the current beat interval, as opposed to the ignored peaks, that all have larger beat onset within the interval. This method avoids the use of a threshold, potentially dependent on the music material or recording situation. There are, however, still

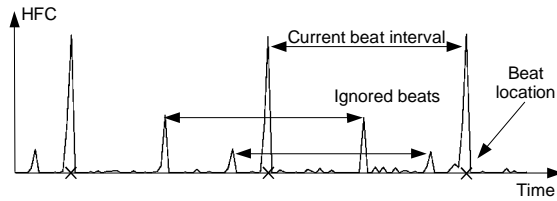


Figure 3: Locating the absolute position of beats. The current beat intervals are found using the beat induction histogram. A beat location is identified when no larger peaks are within the current beat interval. Beat locations are indicated by 'X' on the time axis.

some potential problems, in case the *off beat* peaks are as strong as the *on beat* peaks, or if the *on beats* are missing. These problems are avoided by the use of a multiplication threshold; the current peak is multiplied with a constant factor in case it is situated one beat interval later than the previous beat location. This enhances the probability that the right beat is selected.

## 6. EVALUATION

The tempo and beat estimation has been implemented in Mixxx [8], and tested using a database of 2164 songs. The database consist mainly of popular music, but it contains also classical and techno music.

The performance of the system is satisfactory, although we have found several cases where the result deviations from the expected. It seems, for instance, for some songs, that the correct tempo is easier found in the start of the song and a shift occur, to an incorrect value, in a later part of the song, where more instruments are introduced. Some of the errors thus introduced can be removed using a hysteresis function that puts more weight on the current beat interval in the histogram. Nevertheless, the tempo shifts to e.g. 4/3 of the correct tempo in BioBas / Learning Tennis 3<sup>1</sup>. Other tempo deviations, such as tempo octave errors, are not considered errors, as many humans can have similar discrepancies.

The different evaluation measures on the tempo and beat marking output are the startup time, defined as the first position of the longest segment in the song with less than 10 BPM change between consecutive values. The tempo change between the start and the end of this stable segment is calculated, and found to be below 10 BPM for most songs. In addition, the deviations of the marked intervals to the histogram beat intervals are calculated, and the first quartile, median and third quartile are found for each song. These deviations, limited in the beat estimation to less than  $\pm 50$  msec., were found to lie between -23 to 2 msec, -5 to 5, and 1 to 24 msec. respectively. 67%, 66% and 24% were found to be below 2 msec., respectively.

The start-up time confound intros with no clear rhythmic information and the initialization of the beat induction histogram. No further analysis of the division of the start-up time has been performed. The start-up is measured in seconds, and the first quartile, median and third quartile of the number of songs are measured for different start-up time. 25% of the songs were found to have below 1.1 second startup time, half the songs had below 6.7 sec., and 75% below 26.0 sec.

<sup>1</sup>Available at <http://www.musicssystem.dk/sportmusic/>

## 7. CONCLUSIONS

This paper presents a complete system for the estimation of beat interval, including the estimation of the exact location of the beat in music. The system consists of the calculation of an audio feature that has been selected from a large number of potential features. A number of error measures have been calculated, and the best feature has been found, together with the optimum block size, from the analysis of the error measures. The selected feature is further enhanced in a beat induction histogram. This histogram, which keeps in memory the previous most likely intervals, renders an indication of the current interval.

An extensive evaluation on 2164 songs has shown that the beat estimation is robust and finds the correct beat before 6.7 seconds on half the songs. The deviations of the beat location intervals to the estimated beat interval were found to lie between  $\pm 25$  msec.

The paper has presented several new features, a novel approach to the feature selection, and a versatile beat estimation. In addition, the beat locations are estimated using a simple and reliable method. It is computational relatively inexpensive and it is implemented in the open source software DJ system Mixxx [8].

## 8. REFERENCES

- [1] T. H. Andersen, "Mixxx: Towards novel DJ interfaces," in *Proceedings of the New Interfaces for Musical Expression (NIME'03) conference, Montreal, May 2003*.
- [2] D. Murphy, T. H. Andersen, and K. Jensen, "Conducting audio files via computer vision," in *Proceedings of the Gesture Workshop, Genova, 2003*.
- [3] M. Goto and Y. Muraoka, "A real-time beat tracking system for audio signals," in *Proceedings of the International Computer Music Conference, 1995*, pp. 171–174.
- [4] —, "A real-time beat tracking for drumless audio signals: Chord change detection for musical decisions," *Speech Communication*, vol. 27, pp. 311–335, 1998.
- [5] E. D. Scheirer, "Tempo and beat analysis of acoustic musical signals," *J. Acoust. Soc. Am.*, vol. 103, no. 1, pp. 588–601, January 1998.
- [6] S. Dixon, "Automatic extraction of tempo and beat from expressive performances," *Journal of New Music Research*, vol. 30, no. 1, pp. 39–58, 2001.
- [7] P. Desain, "A (de)composable theory of rhythm," *Music Perception*, vol. 9, no. 4, pp. 439–454, 1992.
- [8] T. H. Andersen and K. H. Andersen, "Mixxx," <http://mixxx.sourceforge.net/>, July 2003.
- [9] K. Jensen and T. H. Andersen, "Real-time beat estimation using feature extraction," in *Proceedings of the Computer Music Modeling and Retrieval Symposium*, ser. Lecture Notes in Computer Science. Springer Verlag, 2003.
- [10] K. Jensen and D. Murphy, "Segmenting melodies into notes," in *Proceedings of the DSAGM, Copenhagen, Denmark, 2001*.
- [11] J. W. Gordon, "The perceptual attack time of musical tones," *J. Acoust. Soc. Am.*, vol. 82, no. 2, July 1987.