EVALUATING LOW-LEVEL FEATURES FOR BEAT CLASSIFICATION AND TRACKING

Fabien Gouyon, Simon Dixon

Austrian Research Institute for Artificial Intelligence
Vienna, Austria

Gerhard Widmer

Johannes Kepler University
Linz, Austria

ABSTRACT

In this paper, we address the question of which low-level acoustical features are the most adequate for identifying music beats computationally. We consider 172 features computed on consecutive signal frames and systematically evaluate their individual value in the task of providing reliable cues for the presence and localisation of beats in music signals. We compare two ways of evaluating features: their accuracy in a song-specific classification task (classifying beats vs non-beats) and their performance as a front-end to a beat tracking system.

Index Terms— Music, Beat tracking, Rhythm analysis, Feature extraction, Learning systems

1. INTRODUCTION

Many algorithms have been proposed for beat induction and tracking from music audio signals (see [1] for a review), and most of these algorithms share a common general scheme. First, the audio data is parsed into temporal series of features which hopefully convey the predominant rhythmic information. These features are then processed in order to highlight intrinsic periodicities (pulse *induction*). Music data are rarely exactly periodic, so algorithms implement strategies to cope with deviations from constant tempo (beat *tracking*).

While, on the one hand, many diverse formalisms have been proposed to induce periodicities from feature lists (e.g. autocorrelation, Fourier transform, comb filterbanks) and to track changing periodicities (e.g. rule-based models, adaptive oscillators, agents), on the other hand, the literature on low-level features for beat induction and tracking is scarce. Music perception literature does refer to many different cues to beat induction and tracking ("phenomenal accents"), as patterns of time intervals, sudden changes in dynamics or timbre, long notes, pitch leaps and harmonic changes. However, most of these theories have been developed with simplified music stimuli (artificial sequences of synthesised notes), and cannot easily be translated into algorithms dealing with audio signals.

A scan of the literature [1] reveals that relatively few low-level features have been considered so far; these are energy values or temporal variations thereof in several frequency bands, some onset detection functions [2], spectral flux [3] and a few other spectral features.

In this paper, we aim at identifying systematically among a large number of low-level features computed at a regular sampling rate those whose temporal behavior would best indicate the presence and localisation of beats, as measured on audio data whose beats have been previously manually annotated.

2. DATA, METADATA AND FEATURES

A total of 1360 audio files in linear PCM format were used, taken from commercial CDs, and containing 90643 beats in total (with a minimum of 7 beats per piece and a maximum of 262 beats per piece). The audio data is not publicly available for copyright reasons. The data is grouped in 10 categories as follows: *Acoustic*, 84 pieces; *Afro-American*, 93 pieces; *Balkan/Greek*, 144 pieces; *Choral*, 21 pieces; *Classical*, 204 pieces; *Classical solo*, 79 pieces; *Electronic*, 165 pieces; *Jazz/Blues*, 194 pieces; *Rock/Pop*, 334 pieces; and *Samba*, 42 pieces. See [4] for more details on the data.

A total of 172 features were used. For all the features, the frame size was set to 23.2 ms and the hop size to 11.6 ms (1024 and 512 samples, respectively, at a sampling frequency of 44100 Hz). The feature sampling rate is therefore 86.1 Hz. Here follows a list of features considered. The first-order differential of the following features: spectral peak (from now on SP) harmonic deviation (f_1), spectrum low-frequency energy relation (f_2), SP third tristimulus (f_3), spectrum maximum magnitude frequency (f_4), SP second tristimulus (f_5), spectrum rolloff (f_6), SP first tristimulus (f_7), spectrum magnitude kurtosis (f_8), spectrum magnitude skewness (f_9), zero-crossing rate (f_{10}), SP harmonic centroid (f_{11}), energy (f_{12}), spectrum spread (f_{13}), spectrum high frequency content (f_{14}), spectrum centroid (f_{15}), spectrum flatness (f_{16}), spectrum magnitude geometric mean (f_{17}), SP magnitude mean (f_{18}) and spectrum magnitude mean (f_{19}). 13 Mel-Frequency Cepstral Coefficients (f_{20} to f_{32}). The magnitude of the energy in frequency subbands, as well as feature differentials thereof, diverse filterbank definitions being considered:[1] those promoted in [5] (magnitude values: f_{33} to f_{40}, first-order differ-

[1] Here, all frequency subbands are ordered from low to high frequencies.

entials: f_{41} to f_{48} and magnitude-normalised first-order differentials: f_{49} to f_{56}), 36 ERB (Equivalent Rectangular Bandwidth) bands distributed between 50 Hz and 20 kHz (magnitude values: f_{57} to f_{92}, first-order differentials: f_{93} to f_{128} and magnitude-normalised first-order differentials: f_{129} to f_{164}). An implementation of the energy features proposed in [6] (f_{165} to f_{168}). The implementation by [7] of 4 onset detection features, i.e. high-frequency content (f_{169}), phase deviation (f_{170}), spectral difference (f_{171}) and complex spectral difference (f_{172}). More details on feature implementation can be found in [4].

3. METHODS

3.1. Classification

We define two classes: beats and non-beats, and evaluate features on each music piece according to the following criterion: relevant features are those whose values permit a machine learning algorithm to achieve high levels of accuracy in beat classification. Given the time indexes of beats and the time series of frame feature values, the feature value associated with each beat is taken from the frame in the near vicinity of the beat where the feature value is maximum [4]. Instances of non-beats are generated by selecting a random point between each pair of beats. We used a total of 89283 non-beats.

Features are evaluated according to the predictive accuracy of an instance-based classifier (k-NN, with k=3).[2] Classification accuracies are computed via 10-fold cross-validations, computed on *individual music pieces*. An accuracy estimate of a given feature subset is obtained for each piece, and the final accuracy estimate is then computed as the average over the whole set of pieces (or the pieces of a specific music category, when indicated). The evaluation of a given feature accounts for a reduced number of instances taken from the same music piece, hence the obvious danger of overfitting. However, we get a valid estimate of relevance of this feature by averaging over a significant number of music pieces.

As we define the same number of beats and non-beats for each piece, the classification rate when always guessing the most probable class (i.e. the baseline) is 50%. This value should be kept in mind when assessing the goodness of any feature set (an accuracy of 50% is bad as it corresponds to the chance level).

3.2. Beat tracking

The second evaluation procedure focuses on the performance of each feature as front-end to the beat tracking system BeatRoot [8]. In BeatRoot, initial processing of the audio signal is concerned with finding the onsets of music notes. The original version of BeatRoot used a time-domain onset detection algorithm, which found local peaks in the slope of a smoothed amplitude envelope. Although well-suited to music with drums, this method was less reliable at finding onsets of other instruments, especially in a polyphonic setting, so it was replaced with an onset detector based on spectral flux (see [9]). In these experiments, the spectral flux function is replaced by the feature which is being evaluated, and peaks in this feature are considered as onsets for the purposes of beat tracking.

Given a feature vector $f(i)$, the peak-picking algorithm selects a peak at frame number n, subject to the following constraints:

$$f(n) \geq f(k) \text{ for all } k \text{ such that } n - w \leq k \leq n + w$$
$$f(n) \geq \frac{\sum_{k=n-mw}^{n+w} f(k)}{mw + w + 1} + \delta$$
$$f(n) \geq g_\alpha(n-1)$$

where $w = 3$ is the size of the window used to find a local maximum, $m = 3$ is a multiplier so that the mean is calculated over a larger range before the peak, δ is the threshold above the local mean which an onset must reach, and $g_\alpha(n)$ is a threshold function with parameter α given by $g_\alpha(n) = \max(f(n),\ \alpha g_\alpha(n-1) + (1-\alpha)f(n))$

The tempo induction algorithm uses the calculated onset times to compute clusters of inter-onset intervals (IOIs). An IOI is defined to be the time interval between any pair of onsets, not necessarily successive. A clustering algorithm finds the most significant metrical units, and the clusters are then compared to find reinforcing groups, and a ranked set of tempo hypotheses is computed. Based on these hypotheses, the beat tracking algorithm employs a multiple agent architecture to match sequences of beats to the music, where each agent represents a specific tempo and alignment of beats with the music. The agents are evaluated on the basis of the regularity, continuity and salience of the onsets corresponding to hypothesised beats, and the highest ranked beat sequence is returned as the solution.

These beat sequences are evaluated by combining the number of matched beats b, the number of false positives p and the number of false negatives n to give a score between 0 and 1: $score = \frac{b}{b+p+n}$.

4. RESULTS

The performance of individual features in both tasks is shown on Figure 1. Table 1 gives the 4 best features for each task as well as a breakdown with respect to music categories.

For all features, classification accuracies are higher than beat tracking performance. There are several reasons for that. The first reason lies in the beat tracking performance measure itself. Unlike the classification accuracy, this measure is brittle: because the beat tracker focuses on a specific metrical

[2] Experiments described in this paper have been conducted with the free software Weka, available under GPL at `http://www.cs.waikato.ac.nz/ml/weka`.

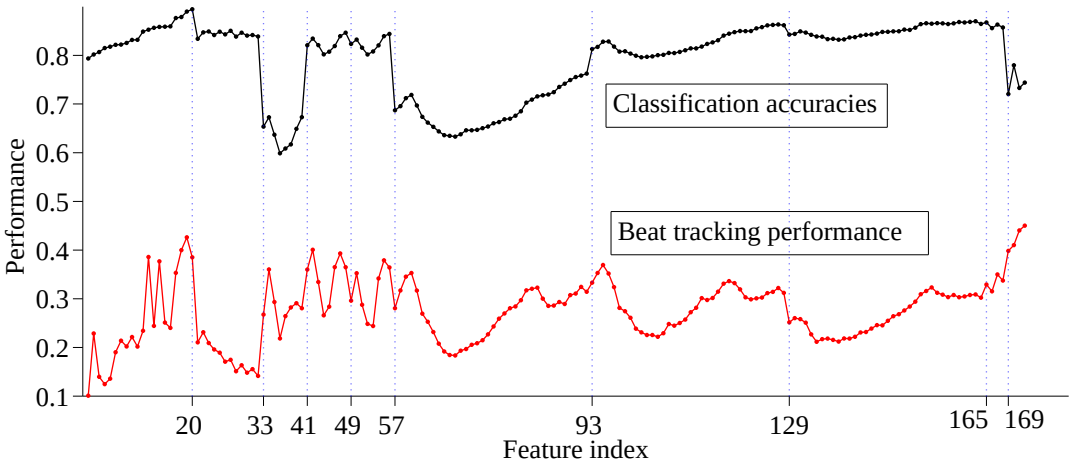

Fig. 1. Performance (in %) of individual features in the classification and beat tracking tasks. Features are indexed by families: spectral features (from 1 to 19), first-order differential of MFCCs (20-32), energy in Dixon's filterbank (33-56), energy in ERB filterbank (57-164), Klapuri's features (165-168) and Bello's onset detection features (169-172).

level, a focus on a wrong metrical level can cause the performance to decrease by e.g. one half. This is not the case for classification as the correct or incorrect classification of a given beat does not depend on its distance to other beats, but only on the feature value. Another reason is that, unlike the beat tracker, the classification process uses some ground-truth knowledge when making decisions (i.e. in each run, $\frac{9}{10}$ of the annotated beats are used to learn how to classify the remaining $\frac{1}{10}$). Further, the classifier is asked to make decisions on relatively few instances, as we defined as many non-beats as beats, while for BeatRoot, any time point is a potential beat. These factors make the classification accuracies overly optimistic with respect to the beat tracking task.

Even if the curves do not have exactly the same shape, they are correlated. To a certain extent, with the exception of some outliers, the *relative ranking* of features is similar in both tasks (at least within a given feature family). This tells us that classification accuracies are somehow representative of the worth of features for beat tracking.

However, in some cases, relative feature rankings are not similar for both methods. It is our belief that differences in relative ranking indicate features for which the use of ground-truth data makes a difference in the determination of beats. For instance, features related to the representation of the timbre —the first-order differential of MFCCs (features 20 to 32) and some spectral features such as the spectrum centroid (f_{15}) and flatness (f_{16})— are very good for classification while they do not score well in beat tracking. It may be that these timbre features work well in classification because they permit the classifier to learn global spectral shapes (i.e. rough instrument models) specific to beats of each music piece. On the other hand, the beat tracker derives discrete data (peak positions) from continuous features by peak-picking, and peaks detected in timbre features represent relatively badly (with respect to other features) beat positions and periodicities of interest. The beat tracker peak-picking algorithm adapts its threshold to each music piece, but, unlike the classifier, this adaptation is unsupervised, i.e. it has no feedback about what works and what not. In sum, timbre features are relevant in the representation of beats, but in order to take the most advantage of these features, beat tracking should adapt in some way to the particular timbre recurring on the beats of each music piece at hand.

As can be seen in Table1, on average, the best feature for classification is the first-order differential of the first MFCC (which amounts to the variation of the signal energy in dB). Other good features are the variation of the energy in low and high frequency bands (between 100 and 400 Hz and above 5 kHz) and of measures of the spectrum magnitude mean (features 17 to 19). These features are correlated with note onsets. Beat tracking also performs very well with these features, and also with [7]'s onset detection features (the best feature being the complex spectral difference). This confirms the common belief that onset times and IOIs are strongly correlated with beat positions and periodicities of interest.

Both methods show similar relative rankings of the energy (or variation thereof) in frequency subbands (e.g. features 57 to 92). They show for instance that energy between 500 Hz and 1.5 kHz (ERB bands 9 to 15) is relatively irrelevant to beat tracking. An interpretation is that the voice, whose spectral energy is maximally present between these frequencies, may be the instrument which is, on average, less representative of the metrical structure.

We can also see in Table 1 that the best features depend to some extent on music category. The union of the 4 best features for each of the 10 music categories amounts to a set of 16

		1^{st}	2^{nd}	3^{rd}	4^{th}
Whole data	Cl.	f_{20}	f_{19}	f_{18}	f_{17}
	Tr.	f_{172} (45%)	f_{171}	f_{19}	f_{170}
Acoustic	Cl.	f_{19}	f_{20}	f_{26}	f_{17}
	Tr.	f_{172} (35.9%)	f_{171}	f_{94}	f_{41}
Afro-American	Cl.	f_{19}	f_{17}	f_{20}	f_{164}
	Tr.	f_{172} (53%)	f_{171}	f_{117}	f_{120}
Balkan/Greek	Cl.	f_{19}	f_{20}	f_{18}	f_{165}
	Tr.	f_{41} (41.5%)	f_{94}	f_{93}	f_{42}
Choral	Cl.	f_{11}	f_{16}	f_{129}	f_{22}
	Tr.	f_{33} (11.5%)	f_{95}	f_{59}	f_{94}
Classical	Cl.	f_{20}	f_{21}	f_{166}	f_{15}
	Tr.	f_{170} (35.3%)	f_{55}	f_{56}	f_{47}
Classical Solo	Cl.	f_{20}	f_{21}	f_{19}	f_{166}
	Tr.	f_{170} (37.6%)	f_{55}	f_{172}	f_{171}
Electronic	Cl.	f_{19}	f_{20}	f_{17}	f_{18}
	Tr.	f_{171} (57.6%)	f_{172}	f_{18}	f_{19}
Jazz/Blues	Cl.	f_{17}	f_{19}	f_{20}	f_{165}
	Tr.	f_{41} (43.8%)	f_{170}	f_{172}	f_{94}
Rock-Pop	Cl.	f_{20}	f_{19}	f_{17}	f_{18}
	Tr.	f_{171} (62.3%)	f_{172}	f_{19}	f_{169}
Samba	Cl.	f_{128}	f_{17}	f_{125}	f_{126}
	Tr.	f_{58} (53.6%)	f_{59}	f_{95}	f_{94}

Table 1. First to fourth best feature for each method, classification (Cl.) and beat tracking (Tr.), for all music categories. Percentages in parenthesis indicate beat tracking performance.

different features for classification and 19 different features for beat tracking. This indicates that a beat tracker may take advantage of a hypothetical knowledge of the music genre of the pieces it has to process. For instance, if the best feature per category is used instead of the globally best (complex spectral difference), an improvement of 3.1 percentage points is obtained (i.e. 48.1% instead of 45%).

5. SUMMARY AND FUTURE WORK

The main contribution of this paper is to bring forward a new issue in automatic rhythm description of audio signals: the question of *which* acoustical features are the most adequate for identifying music beats computationally. We evaluated the worth of a large number of features in both a classification task and a beat tracking system.

Individual features which are best for beat tracking are those which indicate the presence of onsets [7]. Energy features are more relevant in low and high frequency bands. However, feature performance depends on music category. Deeper analyses of errors will determine the extent to which features fail on specific categories. The difference between classification and beat tracking performance shows that performance could be potentially improved by using some knowledge of the acoustical characteristics of the beats of each music piece. This is especially true for the case of timbre features which, although they are shown to capture beat characteristics, are relatively irrelevant in unsupervised beat tracking. Future research could therefore focus on *adaptive* beat tracking. A starting point may be the design of interactive beat trackers where the user would have to provide some simple feedback on how well the algorithm is doing or e.g. specify a few beats manually. This feedback could be used by the algorithm to better define the concept of beat on each piece. Future work could also be dedicated to evaluate combinations of features instead of individual features [4] and extend the analysis to different beat trackers (e.g. that do not discretise features).

Acknowledgments

This research was partly funded by the projects S2S^2 and Interfaces2Music. Thanks to Anssi Klapuri, Stephen Hainsworth, Giorgos Emmanouil, Matthew Davies and Juan Bello.

6. REFERENCES

[1] F. Gouyon and S. Dixon, "A review of automatic rhythm description systems," *Computer Music Journal*, vol. 29, no. 1, pp. 34–54, 2005.

[2] M. Davies and M. Plumbley, "Beat tracking with a two state model," in *Proc. IEEE ICASSP*, 2005, vol. 3.

[3] J. Laroche, "Efficient tempo and beat tracking in audio recordings," *Journal of the Audio Engineering Society*, vol. 51, no. 4, pp. 226–233, 2003.

[4] Gouyon F., *A computational approach to rhythm description*, PhD Thesis, Pompeu Fabra University, Barcelona, 2005.

[5] S. Dixon, E. Pampalk, and G. Widmer, "Classification of dance music by periodicity patterns," in *Proc. International Conference on Music Information Retrieval*, 2003.

[6] A. Klapuri, A. Eronen, and J. Astola, "Analysis of the meter of acoustic musical signals," *IEEE Trans. Speech and Audio Processing*, vol. 14, no. 1, 2006.

[7] J. Bello, *Towards the Automated Analysis of Simple Polyphonic Music: A Knowledge-based Approach*, PhD Thesis, Queen Mary University of London, London, 2003.

[8] S. Dixon, "Automatic extraction of tempo and beat from expressive performances," *Journal of New Music Research*, vol. 30, no. 1, pp. 39–58, 2001.

[9] S. Dixon, "Onset detection revisited," in *Proceedings of the 9th International Conference on Digital Audio Effects*, 2006.