COMPARING MID-LEVEL REPRESENTATIONS FOR AUDIO BASED BEAT TRACKING

Matthew E. P. Davies and Mark D. Plumbley
Centre for Digital Music
Queen Mary University of London

ABSTRACT

Rather than analyse an audio signal directly, many beat tracking algorithms perform some transformation of the input, commonly using note onset times or apply a mid-level representation which emphasises them, as the basis for extracting beat times. In this paper we investigate the importance of the input representation by comparing seven different onset detection functions as input to our beat tracker, while keeping all other aspects constant.

Results indicate that the complex spectral difference approach is both an efficient and accurate representation for identifying beat times. However we illustrate that significant improvements (which exceed the current state of the art) are possible using multiple detection functions with an adaptive input selection stage.

Towards this aim, we find that input selection based on musical genre offers a small increase in accuracy, where as beat strength fails to improve performance.

Keywords – Beat tracking, onset detection, mid-level representations

1. INTRODUCTION

There can be little argument that information related to temporal structure is useful in computational rhythmic analysis. Whether it is the explicit extraction of note onset times [1, 6, 9] or the generation of some mid-level representation (e.g. an onset detection function) which emphasises them [12, 5]; for the task of beat tracking, some form of pre-processing appears essential.

This raises the question of whether the extra computation required to extract onsets, by finding the indices of local maxima in the detection function (taking a step towards a more symbolic approach) allows for greater accuracy in beat tracking, or if onset detection functions actually encode rhythmic information beyond note onset locations which might aid the beat extraction process.

We demonstrate that while keeping all other features of our beat tracker constant, the use of a detection function rather than a train of onsets (a sequence of impulses at onset times weighted by their strength) leads to significantly better results, largely independent of the choice of detection function.

Assuming this to be a general result (beyond the small test database we use), we then seek to discover how important the choice of detection function is as the primary input.

We examine seven different detection functions to identify which onset detection function is the best "beat detection function". While spectral difference based approaches appear to generalise well over the entire dataset, preliminary results indicate that some examples within our test set are better suited to particular detection functions than others. In the most extreme case of over-fitting (where the best performing detection function is matched to each input file) a 15 % improvement in performance can be made. However the manual selection of a detection function in this manner is by no means practical. We would therefore like to discover some means to adaptively select an appropriate input representation to match the characteristics of the input signal. Towards this aim we examine whether meta-data related to musical genre is sufficient to improve performance, i.e. select detection function X for jazz, and Y for rock. We compare this manual approach to automatically extracting a feature related to beat strength [13] within a fully adaptive process.

Selecting detection functions based on musical genre offers a small improvement, but surprisingly the somewhat intuitive feature of beat strength fails to yield any improvement. Analysis beyond these two features is left as a topic for further research, which we believe should include the collection of more meta-data in combination with machine learning techniques to better identify those features which are most salient to our beat tracking formulation.

The remainder of this paper is structured as follows: in section 2 we describe the onset detection functions used as inputs to our beat tracker, followed in section 3 by an overview of the beat tracker itself as well as our chosen evaluation method. Results are presented in section 4, with a final overview and conclusions in sections 5 and 6.

2. MID-LEVEL INPUTS

The mid-level inputs, or onset detection functions we use as the input to our beat tracker are taken from two recent onset detection comparative studies [2, 4]. As illustrated in these comparisons, the performance of any individual detection function is dependent on the properties of the input signal. Two main types of onsets have been identified: percussive and tonal. Percussive onsets can be either pitched (a piano note) or non-pitched (a drum hit) and are characterised by sharp changes in signal energy and are typically easy to detect. Tonal onsets (a bowed violin note) however are harder to identify as little or no energy change may be perceptible. The detection of tonal onsets is therefore reliant on some pitch or harmonic based analysis.

A breakdown of the onset detection functions we use are given in table 1, indicating their sensitivity to both types of onsets along with the approximate time taken to process 1 minute of mono audio sampled at 44.1kHz [1]. Those without specific names are given the name of the primary author. Each detection function is shown graphically in figure 1 for a 6 second audio segment.

Conceptually, the simplest of the detection functions is the *High Frequency Content* (HFC) approach [11, 2]. Which finds the

Email: matthew.davies@elec.qmul.ac.uk

[1] 2.8 GHz Linux machine running Matlab 7.0

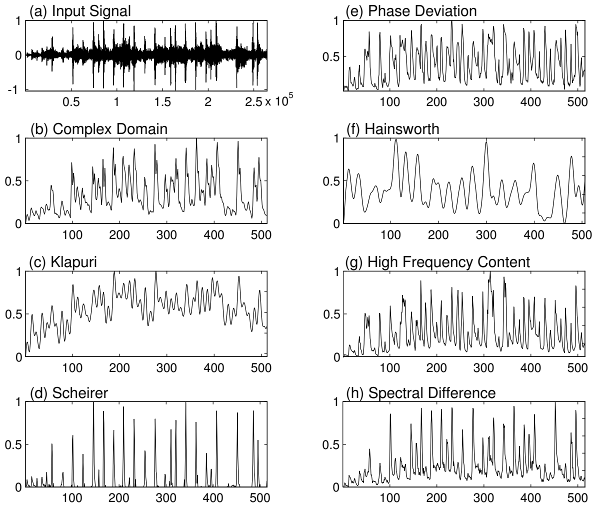

Figure 1. Comparing Detection Functions over a 6 second frame. (a) - (d): Input Signal, Complex Domain, Klapuri, Scheirer. (e) - (h): Phase Deviation, Hainsworth, High Frequency Content, Spectral Difference

Detection Function	Percussive onsets	Tonal onsets	Psycho-acoustic	Processing time (s)
Complex	x	x		1.68
Klapuri	x	x	x	9.27
Scheirer	x		x	5.06
Phase	x	x		7.60
Hainsworth		x		8.53
HFC	x			5.44
Specdiff	x	x		1.49

Table 1. Overview of detection functions. Processing time refers to the time to generate a detection function for 60 seconds of mono audio sampled at 44.1 kHz. Note that processing time is very low for Complex and Specdiff, as these algorithms produce better results when then input signal is down-sampled by a factor of 8 to give a more musically meaningful spectral range (of the seven detection functions tested only these two showed improved performance at this limited spectral range)

sum of frequency weighted short term spectral frames, and is most suited to detecting wide-band percussive events.

The next group of detection functions are based around spectral difference between adjacent frames. *Spectral Difference* [2] is a measure of the Euclidean distance between the magnitude spectra of two adjacent frames. *Hainsworth's* approach [8] uses longer analysis frames with Kullback-Leibler divergence as the distance metric. Its primary emphasis is on harmonic change, as short term percussive events tend to be averaged out. The *Complex Domain* [3] method combines magnitude spectra with phase information to simultaneously detect percussive and tonal onsets, by measuring the spectral difference directly in the complex domain. If only the phase information is used, this equates to the *Phase Deviation* [2] approach which reacts to deviations in phase velocity (which is approximately constant in steady-state, sinusoidal regions of music signals).

The two remaining detection functions, those of *Scheirer* [12] and *Klapuri* [10] have psychoacoustic formulations. Both methods combine amplitude envelopes from sub-bands to generate a detection function type representation. Klapuri's method employs a logarithmic compression to emphasise small energy changes.

3. BEAT TRACKER

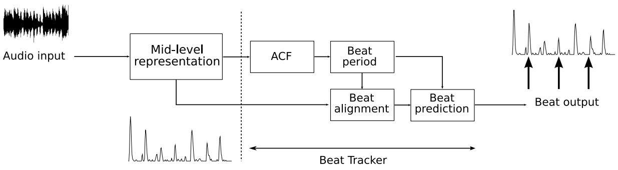

Figure 2. Overview of beat tracker : we investigate varying the mid-level representation

3.1. Algorithm overview

Having presented each of the detection functions to be tested with our beat tracker, we now provide a brief overview of the beat tracker itself (see figure 2).

The first stage in the algorithm is to further process the detection function. A moving median threshold is calculated and then subtracted from the raw detection function. The unbiased autocorrelation function of this modified detection function is then taken. This is then passed through a shift-invariant comb filterbank to identify the beat period (the index of the maximum value of the output). The comb filterbank is weighted by a tempo preference curve to encourage the beat period to be within the approximate range of 80 to 160 beats per minute (bpm). The beat period extraction process is shown in figure 3, where the value in applying the median threshold to the detection function can be seen by the height of the main peak (fig. 3(f)), compared to when the raw detection is used (fig. 3(c)). Having found the beat period, an impulse train (with impulses at beat period intervals) is cross-correlated with the detection function to find the phase of the beats. The process is repeated over a frame basis to track beats across the length of a file. Context dependent parameters are incorporated to force the beats to remain within a single metrical in the beat period stage, and to prevent beats switching between on and off-beats in the phase extraction stage. Further details of the beat tracker may be found in [5].

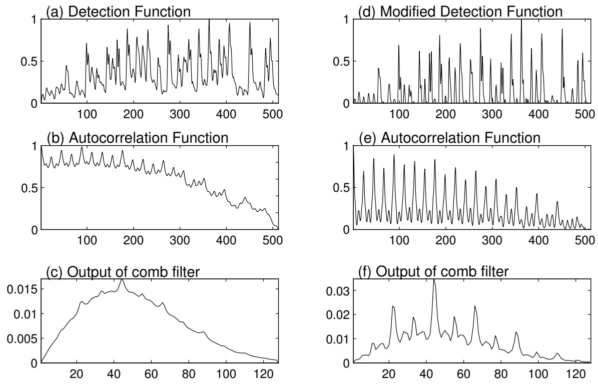

Figure 3. Analysis of the comb filterbank output. (a) - (c): using raw detection function. (d) - (f): effect of applying adaptive threshold to df

3.2. Evaluation

In order to successfully compare the effect of varying the input to our beat tracker, we must first address the issue of how to evaluate the output of our beat tracker.

There does not currently exist a widely accepted evaluation metric for audio based beat trackers. This is due, in part, to the difficulties involved in obtaining accurate ground truth data, to which algorithmically generated beats can be compared. A common approach to obtaining ground truth is to ask trained musicians to 'tap' in time to musical examples; recording and then manually correcting beat locations such that they are perceptually acceptable. This is the approach we adopt; our labelled database contains a subset of 100 files (20 files over 5 musical genres) from a larger test database, chosen to maintain an equal distribution across musical genre [9].

The beat tracking evaluation metric is based on the work of Goto and Muraoka [7] and finds the ratio of longest continuously correctly tracked segment to the length of the input as a measure of beat accuracy (figure 4). We accept an individual beat b_k iff:

- b_k is within +/- 20% of annotation a_j
- b_{k-1} is within +/- 25% of a_{j-1}
- b_{k+1} is within +/- 25% of a_{j+1}

Although this may appear to be a strict metric, (where a single mis-placed beat can cause the accuracy to drop from 100 % to 50 %) we can reliably say that high performance when continuity is required will amount to high performance when it is not. However we cannot say that the reserve will be true. It should be noted also, that we have chosen not to allow multiple metrical levels or pi-phase error (tracking off-beats instead of on-beats) as acceptable, to prevent the possibility of a detection function which reliably produced off-beat 1/8th notes to be assumed equal to one which gave on-beat 1/4 notes.

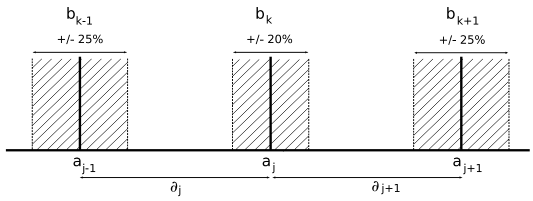

Figure 4. Continuity based beat evaluation: acceptance of beat b_k is dependent not only on distance from the nearest annotated beat a_j, but also how close the previous and subsequent beats are to their annotations

4. RESULTS

The results of running each detection function across the 100 files in the database is shown in table 2 in addition to the performance when the input is comprised of a sequence of weighted impulses at onset locations. For each detection function we can see noticeably reduced performance when onsets alone are used. This may be the result of inaccuracies in the peak picking process - either false negatives (missed detections) or false positives (erroneous detections) which tend to flatten the autocorrelation function taken, and therefore reduce the certainty of the output (the extent to which a single peak is stronger than all other beat period candidates). Alternatively, the increased width of peaks within the detection function, compared to a sole impulse at onset locations, may allow beats to be tracked more effectively in cases of variable tempo (expressive timing). Failure to track such changes can break the continuity requirement and lead to poorer performance.

Input DF	Accuracy using DF (%)	Accuracy using onsets (%)
Complex	**55.89**	**29.63**
Klapuri	46.13	29.31
Scheirer	41.13	22.05
Phase	41.81	20.71
Hainsworth	34.86	23.81
HFC	41.08	22.76
Specdiff	55.32	29.46
Best DF per file	70.64	32.00

Table 2. Results for mid-level inputs compared to using onsets directly

When comparing each of the individual detection functions, the spectral difference based approaches (Complex and Specdiff) perform strongest. While the accuracy figures appear somewhat low (a result of the continuity requirement), the best performing (accuracy = 55%) are equivalent to the state of the art. The Hainsworth approach however, which is also based on spectral difference performs least strongly of all. As shown in figure 1(f), this is the smoothest detection function, and with a primary sensitivity towards harmonic change it fails to accurately localise beats, particularly in signals where the beat structure is predominantly conveyed through percussive events.

What is perhaps most striking is that by selecting the best performing detection function for each file, an improvement of almost 15% can be found. Despite this being a rather extreme case of 'over-fitting' the data, it does suggest that increases in performance can be achieved by investigating adaptive approaches to detection function selection.

Towards the aim of adaptively selecting a detection function, we investigate two intuitive methods: firstly, whether detection functions can be associated with musical genres, and secondly whether the concept of beat strength (a measure of the confidence with which the tempo can be identified) is an appropriate feature.

4.1. Adaptive selection using meta-data: genre labels

The breakdown of results across each musical genre is shown in table 3. Despite the fact that overall the Complex and Specdiff approaches were of similar accuracy, we can see that for those genres which are typically more percussive oriented: dance, rock and jazz, the Specdiff detection function is better than the Complex, while the converse is true for folk and classical music. By simply averaging the best performing detection function for each genre, we find a modest improvement to give an overall accuracy of 59% (although for only 100 files in our test database, this improvement may not be statistically significant).

4.2. Adaptive selection using a feature: beat strength

The use of genre provided some improvement over the Complex or Specdiff approaches alone, this however, is overshadowed by the fact that for a fully automated algorithm, some means of extracting the genre would have to pre-empt any beat analysis. It would seem more desirable to be able to identify a feature which

Input DF	Rock (%)	Dance (%)	Jazz (%)	Folk (%)	Classical (%)
Complex	66.48	85.68	46.96	49.31	**31.03**
Klapuri	39.49	68.79	41.14	**51.37**	29.86
Scheirer	43.39	61.87	32.26	40.50	27.62
Phase	61.14	37.93	40.65	41.75	27.57
Hainsworth	34.83	57.70	26.32	35.26	20.22
HFC	61.82	51.10	39.37	25.17	27.93
Specdiff	**70.12**	**93.26**	**49.42**	37.86	25.92

Table 3. Breakdown of results according to musical genre

Input method	Acc. frame based (%)	Acc. file based (%)	Theoretical limit (%)
Best 2 DFs	48.12	46.12	63.18
Best 3 DFs	52.46	50.82	66.81
All DFs	50.42	49.88	70.64

Table 4. Results for adaptive techniques. Best 2 DFs: Complex and Scheirer. Best 3 DFs: Scheirer, Phase and Specdiff

could be automatically extracted from the signal to guide the selection process. Given that our interest is that of beat tracking, we investigate “beat strength” as one such feature.

Originally proposed by Tzanetakis and Cook [13] (coincidentally as part of a genre classification system) we derive a version of beat strength peculiar to our beat tracker. The approach we adopt is to find the relative height of the strongest peak of the output of the comb filterbank (which is normalised to sum to unity) and is shown in figure 3 (c) and (f). In cases where the autocorrelation function is quite flat, the beat strength will be low, however when there is strong periodicity it will be much higher. Therefore the detection function with the highest beat strength is selected as the input.

The selection between detection functions can be made either at file level - where a single beat strength feature is extracted from a long term autocorrelation function across the length of the entire file. Alternatively this feature can be examined on a frame basis. The frame based approach offers the ability to adapt the detection function to changing properties within a single file. It could also be implemented in a causal beat tracking algorithm (where future beat locations are predicted solely from analysis of past data) - two factors which do not apply when selecting a particular detection function for an entire input.

We investigate the effect of both file and frame based selection, using all seven detection functions and compare to the best combination of two and three detection functions.

The best performing two (or three) detection functions were not simply those with the highest overall accuracy. The method for their selection was based on the independence of their performance across the dataset. A search of the results for each combination of detection functions was carried out, which gave [Complex, Scheirer] as the best two, and [Scheirer, Phase, Specdiff] as the best three.

Table 4 illustrates that in each case the file based approach was outperformed by the frame based selection, but not by any significant amount. More importantly, none of the approaches (either two, three or all seven detection functions) were able to give better results than using just the Complex domain approach.

Input Method	Accuracy (%)
Complex DF	55.89
Best 3 DFs	52.46
Genre specific DF	59.15
Best DF per file	**70.24**

Table 5. Overview of results. Best 3 DFs refers to frame based adaptive selection

5. OVERVIEW

In presenting results comparing which single detection function is the best mid-level representation as well two approaches to adaptive input selection, we have found somewhat mixed results, as shown in table 5. Selecting detection functions based on musical genre offers a small improvement, but the beat strength feature can only degrade performance (coupled with a significant increase in computation - as most is taken up in the generation of the detection function, rather than the beat tracker itself).

The reason for the decrease in performance is that selecting detection functions based on our formulation of beat strength is biased towards those detection functions which are least smooth (Scheirer and High Frequency Content for example). This can cause a more prominent peak in the output of the comb filterbank, and hence lead to the selection of generally poorer performing detection functions - and was particularly evident in file-based selection (where there is no opportunity to recover from a bad detection function selection which is possible in the frame based approach).

Being generally smoother, the spectral difference approaches (Specdiff and Complex) were able to provide longer continuously correct beat outputs. This is seen to aid in the tracking of music with expressive timing, perhaps at the expense of precision in beat localisation. Given the difficulties in obtaining ground truth - related to ranges of acceptable beat locations, we do not consider this to be a failure of the spectral difference approaches.

In identifying the failure of beat strength, we intend to investigate other low level features which can be extracted from detection functions, in combination with further hand labelled meta-data, to improve our approach to adaptive input selection for audio based beat tracking.

6. CONCLUSIONS

We have presented a comparison of onset detection functions for beat tracking. Results indicate the spectral difference based approaches are most effective over a wide range of musical genres, but that significant improvements are possible by adaptively selecting between several detection functions to match the properties of individual input signals. Towards this aim we have investigated using musical genre to select the appropriate detection function as well as a feature related to beat strength, but so far we have only found an improvement when using genre related meta-data. We intend to pursue the topic of adaptive input selection as part of our future work.

7. ACKNOWLEDGEMENTS

MEPD is supported by a College Studentship from Queen Mary University of London.

This research has been partially supported by EPSRC Grant Reference: GR/S82213/01 Techniques and Algorithms for Understanding the Information Dynamics of Music and by the EU-FP6-IST-507142 project SIMAC (Semantic Interaction with Music Audio Contents). More information can be found at the project website http://www.semanticaudio.org

Thanks to Juan Bello and Nick Collins for making their code available.

8. REFERENCES

[1] P. E. Allen and R. B. Dannenberg, "Tracking musical beats in real time," in *Proceedings of International Computer Music Conference*, pp. 140-143, 1990

[2] J.P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M.E. Davies and M.B. Sandler, "A tutorial on onset detection in music signals," *IEEE Transactions on Speech and Audio Processing* - to appear 2005

[3] J. P. Bello, C. Duxbury, M. E. Davies and M. B. Sandler, "On the use of phase and energy for musical onset detection in the complex domain," *IEEE Signal Processing Letters*, vol. 11, no. 6, pp. 553-556, 2004, July

[4] N. Collins "A comparison of sound onset detection algorithms with emphasis on psychoacoustically motivated detection functions," in *Proceedings of 118th AES Convention*, Barcelona, Spain, May 28–31, 2005

[5] M. E. P. Davies and M. D. Plumbley, "Beat tracking with a two state model," in *Proceedings of ICASSP*, Philadelphia, USA, March 18–23, 2005

[6] S. Dixon, "Automatic extraction of tempo and beat from expressive performances," *Journal of New Music Research*, vol. 30, no. 1, March, 2001

[7] M. Goto and Y. Muraoka, "Issues in evaluating beat tracking systems," in *Working Notes of the IJCAI-97 Workshop on Issues in AI and Music - Evaluation and Assessment*, pp 9-16, August, 1997

[8] S. Hainsworth and M. Macleod, "Onset detection in musical audio signals," in *Proceedings of the International Computer Music Conference*, Singapore, September, 2003

[9] S. Hainsworth, "*Techniques for the automated analysis of musical audio*," Ph.D. thesis, Department of Engineering, Cambridge University, April, 2004. Also available at http://www-sigproc.eng.cam.ac.uk/∼swh21/thesis.pdf

[10] A. Klapuri, "Sound onset detection by applying psychoacoustic knowledge," in *Proceedings of ICASSP*, pp. 3089-92, Phoenix, USA, March 15–19, 1999

[11] P. Masri and A. Bateman, "Improved modelling of attack transients in music analysis-resynthesis," in *Proceedings of International Computer Music Conference*, pp. 100-103, Hong Kong, August 19–24, 1996

[12] E. Scheirer, "Tempo and beat analysis of acoustic musical signals," *Journal of the Acoustical Society of America*, vol. 103, pp. 588-601, January, 1998

[13] G. Tzanetakis and P. Cook, "Musical Genre Classification of Audio Signals," *IEEE Transactions on Speech and Audio Processing*, vol. 10 no. 5, July, 2002