Audio Engineering Society

Convention Paper

Presented at the 118th Convention
2005 May 28–31 Barcelona, Spain

Beat Tracking Towards Automatic Musical Accompaniment

Matthew E P Davies[1], Paul M Brossier[1], and Mark D Plumbley[1]

[1] *Centre for Digital Music, Queen Mary University of London, Mile End Road, E1 4NS, UK*

Correspondence should be addressed to Matthew E P Davies (matthew.davies@elec.qmul.ac.uk)

ABSTRACT

In this paper we address the issue of causal rhythmic analysis, primarily towards predicting the locations of musical beats such that they are consistent with a musical audio input. This will be a key component required for a system capable of automatic accompaniment with a live musician. We are implementing our approach as part of the aubio real-time audio library. While performance for this causal system is reduced in comparison to our previous non-causal system, it is still suitable for our intended purpose.

1. INTRODUCTION

The task of beat tracking within the musical information retrieval community is well known [1]. The principal aim of such research, for which numerous approaches exist (e.g. [2, 3, 4]) is the replication of the innate human ability of tapping in time to music. Not only does this phenomenon occur in a passive listening environment, where the subject apparently requires no musical training to synchronise with the stimulus [4], but the act of tapping one's foot is also a primary tool for keeping time when performing music. We wish to exploit this behaviour in the development of an automatic musical accompaniment system (e.g. [5]) where the beat locations will be used as anchor points around which the temporal structure of an algorithmic accompaniment will be placed. Given that this rhythmic interaction of tapping along to music can only be performed in real-time, it is surprising to note that relatively few published approaches attempt real-time (or even causal) analysis, and indeed even fewer when the musical input is specified as an audio signal rather than comprised of symbolic data, such as a MIDI event list. Of those which do attempt causal analysis, Goto's system [2] is constrained by the need for the musical input to be in 4/4 time, and within a small subset of musical genres (a focus on popular music); Scheirer's approach [3] is very susceptible to switching between metrical levels i.e. the flexibility which allows the tracking of tempo variation is that which produces most errors, and although Hainsworth's beat tracker [1], which uses a particle filter to model tempo vari-

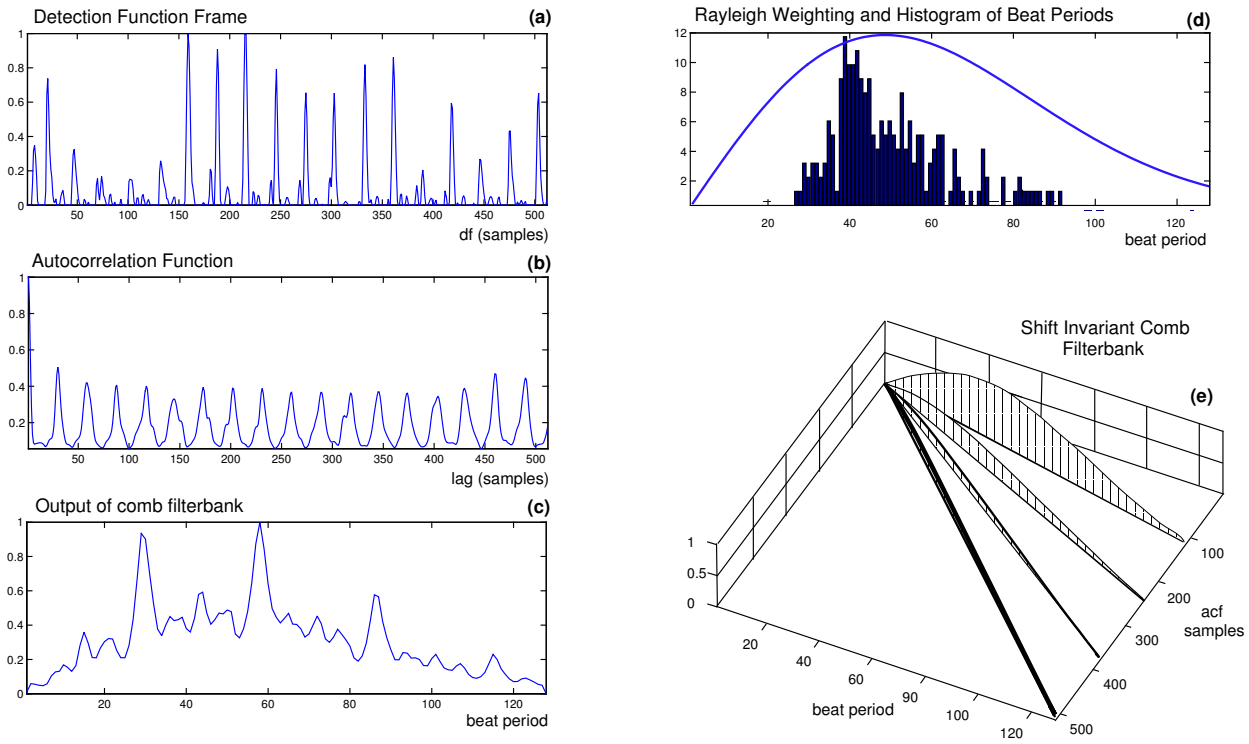

Fig. 1: (a) 6 second DF frame (b) unbiased ACF of DF (c) output of passing ACF through comb filterbank (d) Rayleigh tempo preference curve and histogram of beat periods from evaluation data set (e) matrix representation of shift-invariant comb filterbank.

ation has been shown to give accurate performance over a wide test corpus, it is too computationally expensive to be realised in real-time.

1.1. Developing a beat tracking model

In order to develop a real-time beat tracker our audio analysis must first be *causal*, because without access to future signal information, all beats must be predicted solely from past evidence. However, by its very nature, human musical performance is not rhythmically constant. It is perturbed by both natural variation in tempo as well as intentional expressive timing changes - factors which remain largely unpredictable without analysis of past individual performances [5]. With this in mind, we must concede that our best guess for a future beat location (typically the last beat plus the previous inter beat interval) will not be as accurate as the value provided by a non-causal model which may directly observe performance variation through analysis of past and future data.

We formulate our approach to beat tracking around three empirical musical assumptions chosen to represent a wide variety of musical signals with the aim of minimizing those errors which are most common to beat trackers (e.g. [3]):

- beats mostly correspond with note onsets
- the tempo of a musical piece remains approximately constant
- phase changing of beats - switching between on and off-beats is rare

That is, when tapping in time to music, we are able to do so without active concentration because there are obvious synchronisation points such as

strong musical events and repeating rhythmic patterns. Then, once we induce the beat, we can reliably predict future beats because the rate of beats, as well their phase, will remain roughly constant. We have attempted to replicate this behaviour in our approach to beat tracking through the use of a context dependent model. By allowing the system to identify to the beat rate and phase, and then encouraging it to rely on this information until such a point when contradictory evidence is observed, we are able to impose the kind of contextual continuity required for an accurate beat tracker.

The remainder of this paper is structured as follows: section 2 contains a description of the design of the beat tracking algorithm, followed in section 3 where the causal operation is considered. In section 4 we address the evaluation procedure and results, with conclusions in section 5.

2. ALGORITHM DESIGN

The algorithm we present for causal beat analysis is a conversion of our recent off-line model such that it is able to predict future beats solely from past data, rather than directly align them to an observed signal. A more detailed explanation of the algorithm may be found in [6], as in this section we will provide a brief overview, concentrating on those facets which enable causal, and eventually real-time execution. Analysis occurs over 6 second frames, with a 1.5 second step size (75% overlap).

2.1. General Model

From the input audio signal, we first generate a mid-level representation, known as the *onset detection function* [7], as the primary signal on which to perform our beat analysis. The detection function (DF) can be considered a continuous representation of onset emphasis, highlighting the complex spectral difference between overlapping short term (22ms) analysis frames where the peaks can correspond to both percussive and tonal note onsets. An example DF can be seen in Fig 1(a).

We then identify two tasks: find the beat period, τ (the time between successive beats), then use the beat period to identify the beat alignment, ϕ (phase). The beat period is found as the maximal output of passing the unbiased autocorrelation function (ACF), $\hat{r}_{df}[l]$, of a detection function frame $df[n]$ (see Fig 1(b)) through the shift-invariant comb filterbank shown in Fig 1(e).

$$\hat{r}_{df}[l] = ((\sum_{n=0}^{N-1} df[n]df[n-l])(|(l-N)|)) \quad (1)$$

where $N = 512$ samples and is the length of one analysis frame. The elements of the comb filterbank, each of which represent a different beat period hypothesis, are weighted by a tempo preference function, $R_w[l]$, derived from the Rayleigh distribution function (eqn (2)), which seeks to extract a lag corresponding to a salient metrical level, within the range of 0.375 - 0.75 seconds (between 80 to 160 bpm) as the beat period. The parameter b is set to a lag of 48 samples and corresponds to the mean beat period for the test database used in our evaluation, see Fig 1(d)

$$R_w[l] = \frac{l}{b^2} e^{\frac{-l^2}{2b^2}} \quad (2)$$

Though this is not guaranteed to be the correct beat level, it should correspond to the preferred rate at which a human would tap along to the input. The example output given in Fig 1(c) illustrates prominent resonance at 3 metrical levels, those corresponding to 180bpm, 90bpm and 45bpm (lags of 28, 58 and 116 samples respectively). The tempo preference weighting causes the beat period corresponding to 90bpm to be selected.

The beat alignment, ϕ, is then found by cross-correlating an impulse train (with elements at beat period intervals) through the detection function to identify the last beat before the end of the current analysis frame. As shown in the top plot of Fig 2, an exponential weighting, $A_w[n]$, (eqn. (3)) which is dependent on the current value of τ is applied to the detection function to emphasise the most recently observed region of the signal.

$$A_w[n] = e^{\frac{\log(2)}{\tau} n} \quad (3)$$

Beats are then predicted up to one step increment (1.5 seconds, or 128 detection function samples) into the future, at beat period intervals from the measured alignment location, as is shown in the lower plot of Fig 2.

2.2. Context Dependent Model

We expanded the general model for beat tracking, by adding a Context Dependent Model to our system which allows for our musical assumptions by

imposing continuity to the output. Once three consistent beat period values from the General Model have been observed, we are able to generate a new tempo preference function with a Gaussian weighting, $G_w[l]$, (eqn (4)) which is much tighter than the Rayleigh weighting of the General Model.

$$G_w[l] = e^{\frac{-(l-\tau)^2}{2\sigma^2}} \quad (4)$$

The mean of this new weighting is equal to the beat period, τ and variance σ^2 was empirically set to $\tau/8$. This is to allow for some small variation in the allowable beat period, while remaining close to its expected value, and to prevent any chance of the beat period corresponding to a different metrical level being identified as the strongest output of the comb filterbank.

The beat alignment stage of the Context Dependent Model uses previously predicted beats to obtain a more reliable estimate of the location of the last beat. The basic process is the same - an impulse train is again passed through the exponentially weighted current detection function frame. However in this case, we use the final predicted beat from the previous analysis frame γ_{last}, as the best guess of the alignment for our current frame (see Fig 2). The impulses themselves are now weighted by a separate Gaussian function $GA_w[n]$, with a mean at the predicted alignment (or last beat prediction) and variance again set to $\tau/8$. In the same way that the localised Gaussian weighting for the beat period was able to prevent the beat period switching between metrical levels, this Gaussian alignment weighting can prevent switching from the on-beat to the off-beat. Proceeding in this manner we perform causal beat tracking by a process of *repeated induction*.

$$GA_w[n] = e^{\frac{-(n-\gamma_{\text{last}})^2}{2\sigma^2}} \quad (5)$$

3. IMPLEMENTATION

3.1. Causal operation

We now address the operation of our beat tracker in a simulated automatic accompaniment setting. That is, given an input from an isolated musical instrument, we would like to see how our beat tracker performs in predicting the beat locations in time with the music. For our example, we passed the beat tracker a recording of an electric bass guitar track consisting of 8 bars of 1/8th notes (onsets at half-beat intervals), performed at approximately 90 beat per minute (bpm).

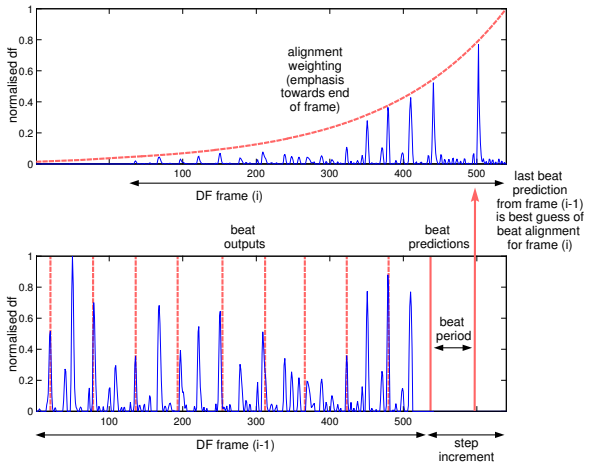

Fig. 2: DF frame (i) with exponential alignment weighting to emphasise end of frame (top) and previous DF frame (i-1) showing last predicted beat is best guess of beat alignment for frame (i) (bottom).

The beats predicted by the algorithm are shown as vertical dotted lines, together with the detection function of the input in Fig 3. Reference to the figure shows that, after a short period of indecision, the algorithm correctly induces the tempo and is able to predict beats in time with the input - a result confirmed on audition of the excerpt with the cowbell clicks synthesized at beat locations.

As described in Section 2.1, the algorithm begins analysis without any prior knowledge of the input (beyond the tempo preference curve used to weight the shift invariant comb filterbank). Unless we provide our system with a more accurate tempo (and starting phase) we cannot expect it to generate an accurate output until has begun to analyse the input data. It should be noted that a full 6 second window need not have elapsed though, as in this example beats begin to be accurately predicted after 3 seconds. Another important aspect to our system is that, once the context dependent model is operational it will continue to output beats even when the input contains no rhythmic structure, or is silent - a factor we consider to be vital for our application.

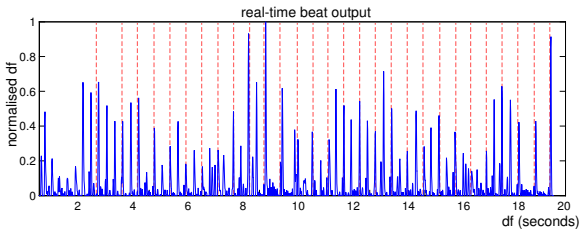

Fig. 3: Detection Function (DF) of bass guitar performance with predicted output beats

We have presented our beat tracker in the context of automatic musical accompaniment. We are currently investigating an appropriate qualitative evaluation strategy for this purpose and will therefore focus on quantitative data obtained using a previously published beat tracking metric [8].

3.2. Porting to Aubio real-time library

We have chosen to pursue the real-time implementation of our causal beat tracking system within the `aubio` library for musical audio labelling as an extension to the real-time onset detection system [9]. Although our approach does not explicitly require the detection of note onsets, it does use the complex domain onset detection function [7] which is generated within the onset detection framework.

Due to the detection function being a heavily subsampled signal representation (sample resolution of 11.6ms) and the relative infrequency of the beat analysis in comparison to the generation of the detection function (beat analysis occurs once every 1.5 seconds), we are confident that the timing constraints of a real-time beat tracking system will easily be met. A further advantage of the beat tracking system is that all beats are predicted solely from past evidence. This in contrast to the onset detection process which must detect onsets within 30ms of their occurrence to maintain an imperceptible delay.

4. EVALUATION

In order to evaluate our beat tracker in a robust and meaningful way, we must address issues related to criteria, test data and ground truth. To gauge relative performance we also present a comparison of our causal algorithm with the following other approaches:

- Beatroot [4]
- Hainsworth [1]
- our non-causal system [6]
- human beat tracking - without manual correction

The human beat tracking data was obtained by recording timing information from computer keyboard taps 'in-time' to the musical examples using the open source audio editor Wavesurfer [10]. We also compare performance against Dixon's beat tracking algorithm *Beatroot* [4] and Hainsworth's particle filter approach [1].

4.1. Criteria: Continuity Emphasis

The aim of the continuity based approach to beat tracking evaluation is to find the ratio of the longest continuously correctly tracked segment to the length of the input file [8]. For this continuity condition to be met, the phase of the beats must be within +/- θ (where for our tests, $\theta = 15\%$ of the annotated value) and the beat period be accurate to within 10%. In this approach four cases have been identified: i) continuity at the correct metrical level (CML cont.); ii) the total number of beats at the correct level, with the continuity constraint relaxed (CML total); iii) continuity where tracking can occur at the metrical level above or below the annotated level (AML cont.); and iv) the total number of beats allowing for ambiguity in metrical level (AML total).

Although this appears to be a robust evaluation strategy, its principal flaw is that individual errors are punished too severely (e.g. one misplaced beat in the middle of an otherwise accurate performance causes the accuracy to drop from 100% to 50%).

4.2. Test Data and Ground Truth

The test database used in our evaluation comprised of 222 musical tracks, each between 30 seconds and 1 minute in length and were divided among the following six musical genres: Rock/Pop, Dance, Jazz, Folk, Classical and Choral [1]. The histogram of beat periods for the test database is shown in Fig 1(d). The ground truth for this dataset was obtained by hand-labelling beat locations for each of the musical excerpts. This process was completed over two stages. Initially, a musician was asked to clap along to each musical example, for which a recording was taken and beat times extracted. This was followed

Beat Tracker	CML Cont. (%)	CML Total (%)	AML Cont. (%)	AML Total (%)
Causal	**44.1**	**51.2**	**56.9**	**71.5**
Non-Causal [6]	57.9	63.7	72.2	84.2
Human	52.3	80.0	56.3	86.6
Beatroot [4]	23.0	27.8	41.7	55.8
Hainsworth [1]	45.1	52.3	65.5	80.4

Table 1: Results comparing algorithm and human performance using "longest continuously correct" criteria. CML refers to the correct metrical level, AML refers to the allowed metrical levels (i.e. one level above and below the annotated correct level).

by a lengthy process of manual correction, where each beat was adjusted such that it sounded in-time when listened back. To confirm this need for manual correction, we chose to compare the beat tracking performance of a human beat tracker against the annotated values. We also performed a test where the non-causal algorithm beats were compared to the unaltered human beat locations. Informal results (37.5%, 60.8%, 48.1%, 79.5%) for each of the criteria in Table 1 indicated a significant decrease in performance when continuity is enforced however a much smaller reduction for the total number of beats. This suggests that human beat tracking performance is likely not to perform well when continuity is required. Further details of the annotation process [1] and additional results in [6] are available.

4.3. Results

Table 1 shows the results obtained from our analysis using the continuity based scheme across each of the five approaches to beat tracking. Unsurprisingly, the results confirm that the accuracy of the causal system is lower than that of the non-causal system. This is the result of two factors implicit in the design of the causal system. Namely, the initial beat outputs are always generated before the activation of the context dependent model, and are therefore open to switching between metrical levels and phases; secondly, and more importantly, the causal beat outputs are predicted from the estimated beat alignment value - a past event in the input, where as in the non-causal case, the beats are aligned to observed data. This means that causal beats are more likely to break the continuity requirement (either by coming too soon or too late), and will hence reduce the overall accuracy of the system.

A specific function of Dixon's algorithm [4] is that when the algorithm fails to detect any reliable beat structure, it will not output any beats at all. This is in contrast to our model, which under all circumstances (including silence) will continue to output beat estimates at the rate and phase defined by the context dependent model (section 2.2), until presented with more reliable evidence. Therefore when analysing Dixon's data, it should be noted that in these *no-beat* cases (28 out of 222 excerpts) an accuracy of 0% was given for the continuity criteria. Although this reluctance to generate a beat output for certain cases has caused a reduction in the overall performance of the system, it remains the weakest of the approaches under each criteria. In contrast to Dixon's approach, Hainsworth's particle filter approach does produce more accurate results, however despite being causal, it is too computationally complex to be realised in a real-time: "on average 5 minutes and 15 seconds per file" [1] compared to less than 10 seconds per file for our approach, when run on computers with similar specifications.

When analysing the human beat tracking data, we can see that it is poorer than the performance of our non-causal algorithm when continuity is required (columns 1 and 3 in Table 1). This was a little surprising - and we should be wary of greater accuracy over human performance (with the exception of cases where the timing is very rigid or mechanical). Inspection of the data reveals the reason for this result. We see a vast improvement in the results for the human beat tracker - a factor which is not replicated with any of the computational approaches. This suggests, not only that the human beat tracking is more affected by the imposition of continuity - that in

many cases only an individual beat was out of time, but also that the human succeeded in finding the correct metrical level in many more of test instances than the algorithmic approaches. As noted with our causal algorithm, the same problems related to expressive timing were also relevant - not being able to predict intentional tempo changes, especially in musical examples unknown to the beat tracker.

The mixed performance of the human beat tracker suggests that that the continuity threshold is too low to give a fair comparison between human and computer performance. It is our intuition also, that human performance would be unaffected by an occasionally poor timed beat - giving us cause to seek an alternative to the continuity criteria when evaluating beat tracking systems. This however, remains an area for further investigation.

5. CONCLUSIONS

We have presented an approach to causal beat tracking towards the aim of a real-time automatic musical accompaniment system. Comparison of our causal model against the non-causal implementation highlights a reduction in performance, however our causal example demonstrated the approach is appropriate for its intended purpose. Analysis of human beat tracking performance suggests that the continuity based evaluation procedure is too strict, and has prompted further investigation into an appropriate metrical evaluation scheme. Towards our automatic accompaniment aim, we are currently extending our rhythmic analysis to incorporate beat subdivisions as well as grouping beats into bars.

6. ACKNOWLEDGEMENTS

The authors would like to thank Steve Hainsworth for the manual beat annotations and Fabien Gouyon for customising Wavesurfer.

Matthew Davies is supported by a College Studentship from Queen Mary University of London. Paul Brossier is supported by the Department of Electronic Engineering at Queen Mary University of London, and by EPSRC grant GR/54620.

This research has been partially funded 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

7. REFERENCES

[1] 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

[2] M. Goto, "An audio-based real-time beat tracking system for music with or without drum-sounds," *Journal of New Music Research*, vol. 30, no.2, pp. 159-171, June, 2001

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

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

[5] C. Raphael, "Automated rhythm transcription", in *Proceedings of the 2nd Annual International Symposium on Music Information Retrieval*, pp.99-106, Bloomington, Indiana, USA, October 15-17, 2001

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

[7] 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, July, 2004

[8] 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

[9] P. M. Brossier, J.P. Bello and M. D. Plumbley, "Real-time temporal segmentation of note objects in music signals," in *Proceedings of International Computer Music Conference* (ICMC) pp. 458-461, Miami, USA, November 2004

[10] F. Gouyon, N. Wack and S. Dixon, "An open source tool for semi-automatic rhythmic annotation," in *Proceedings of 7th International Conference on Digital Audio Effects* (DAFx), pp.193-196 Naples, Italy, October 5–8, 2004