A TWO-FOLD DYNAMIC PROGRAMMING APPROACH TO BEAT TRACKING FOR AUDIO MUSIC WITH TIME-VARYING TEMPO

Fu-Hai Frank Wu , Tsung-Chi Lee, Jyh-Shing Roger Jang
Department of Computer Science
National Tsing Hua University
Hsinchu, Taiwan
{frankwu, leetc, jang}@mirlab.org

Kaichun K. Chang
Department of Computer Science
King's College London
London, United Kingdom
{ken.chang}@kcl.ac.uk

Chun Hung Lu, Wen Nan Wang
Institute For Information Industry
(IDEAS)
Taipei,Taiwan
{enricoghlu, wennen}@iii.org.tw

ABSTRACT

Automatic beat tracking and tempo estimation are challenging tasks, especially for audio music with time-varying tempo. This paper proposes a two-fold dynamic programming (DP) approach to deal with beat tracking with time-varying tempo. In particular, the first DP computes the tempo curve from the tempogram. The second DP identifies the optimum beat positions from the novelty and tempo curves. Experimental results demonstrate satisfactory performance for music with significant tempo variations. The proposed approach was submitted to the task of audio beat tracking in MIREX 2010 and was ranked no. 1 for 6 performance indices out of 10, for the dataset with variable tempo.

Index Terms – Beat tracking, Tempogram, Time-varying tempo, Dynamic programming, Viterbi search

1. INTRODUCTION

Tempo and beat are two essential elements in music. Such information is useful in several applications such as query by tempo (querying a large database based on tempo), beat slicing [17] (segmentation into basic music units separated by beats), and beat synchronous mixing. However, automatic beat tracking and tempo estimation are still challenging tasks when the music has time-varying tempos.

Conventional beat tracking schemes [1] rely on certain assumptions about music contents such as stable tempo over time, periodical percussions/onsets, and four beats per measure. Under these assumptions, most approaches of beat tracking are accomplished by two phases. In the first phase, the onset strength of music along time, called *novelty curve*, is estimated to indicate the possible positions of note onsets. In the second phase, the quasi-periodic patterns in novelty curve are analyzed to discover the possible tempo value and the corresponding beat positions. Here, tempo is assumed to be stable throughout the whole piece of music.

However, the above-mentioned assumptions do not hold true universally, especially for music of classical and jazz. Music of these genres often has significant tempo variations, making it difficult to detect the periodical patterns. In order to detect the variations in tempo, *Frequency Mapped Auto-Correlation Function* (FM-ACF) and *Short-Time Fourier Transform* (STFT) [2] are frequently used to derive a time-frequency representation of the novelty curve, called *tempogram* [3]. The tempo information is embedded in tempogram. We can then apply dynamic programming (DP) to the tempogram to derive the so-called *tempo curve*, which represents the most likely tempo at each time frame.

A number of beat tracking algorithms have been proposed in the literature under different methodologies, including beat-template training [2], neural networks [4], an agent-based method [5], and so on. Among them, DP is still considered an efficient and effective way for determining beat positions. The use of DP for beat tracking has been proposed in [1] with good performance, but it is based on a pre-estimated stable tempo which is estimated by time-domain autocorrelation with window weighting.

There are several important previous studies that attempted to deal with time-varying tempos. Klapuri et al. [18] used the bandwise time-frequency method to obtain accentuation information, then used comb filter resonators and probabilistic models to estimate pulse width and phase of different metrical levels, including tatum, tactus, and measure. Davies and Plumbley [19] proposed the use of complex spectral difference onset function [15] to obtain middle level representation. Their algorithm employs two-state switching model, including general state and context-dependent state, to obtain final beat positions. Groshe and Muller [16] used the novelty curve to generate predominant local pulse (PLP) for estimating time-varying tempos.

In this study, we follow the three-phase framework [2, 6] of beat tracking and attempt to remove the stable-tempo restriction by developing a two-fold DP approach for robust beat tracking with time-varying tempos. To this end, the first DP estimates the time-varying tempo curve from the tempogram (which is obtained from the novelty curve). Then the second DP uses the time-varying tempo curve to identify the optimum beat positions on the novelty curve.

(In fact, we have proposed similar concepts for speech analysis, including DP-based robust pitch determination [13] for Mandarin tone recognition, and DP-based pitch marking [14] for TD-PSOLA synthesis.) In addition, we also propose partial-FFT-based tempo curve estimation and peak picking in tempogram for DP, which enhance the overall efficiency with almost no accuracy loss. The proposed approach was ranked no. 1 for 6 performance indices out of 10, for the dataset of time-varying temp in the audio beat tracking task of MIREX 2010.

The remainder of this paper is organized as follows. Section 2 describes the details of the proposed framework. Performance evaluation is given in Section 3. Section 4 concludes this work with potential future work.

2. SYSTEM DESCRIPTION

The proposed beat tracking system is shown in Figure 1.

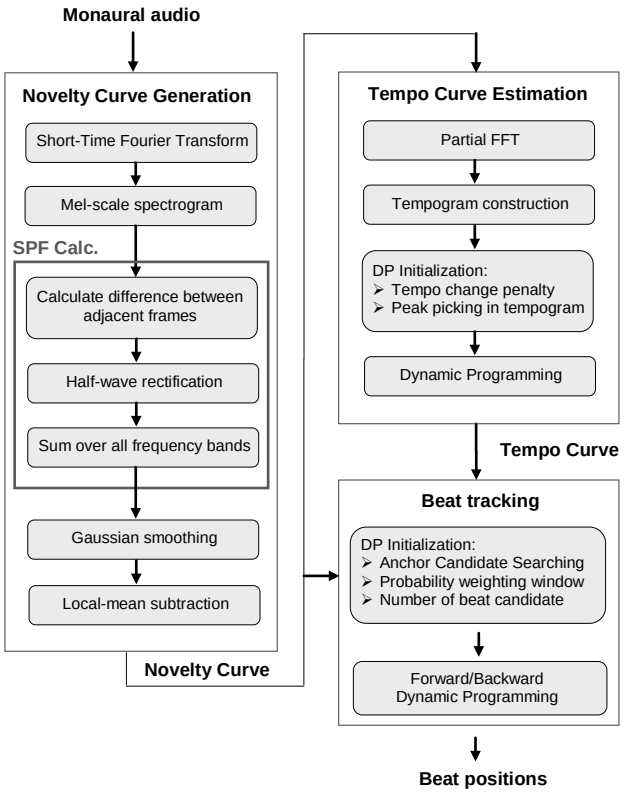


Figure 1. Flowchart of the proposed beat tracking system

The first block computes the novelty curve based on [1, 6]. The second block generates the tempogram and estimates the tempo curve from the novelty curve. In the third block, beat positions are estimated by using the information from previous two blocks. Details of each block will be explained in the following subsections.

2.1 Novelty Curve Estimation

Figure 2 shows typical outputs of various steps in novelty curve estimation. A power spectrogram is first obtained by applying STFT to the source audio with a frame size 31.6 milliseconds and 87.5% overlap. The frequency components of spectrogram are then mapped into Mel-scale in Figure 2(a) for conforming to the characteristics of human perception [1]. Then we apply *spectral flux* (SPF) [12] to obtain the raw novelty curve, as shown in Figure 2 (b)

To be more specific, we have 40 bands in the Mel-scale spectrogram, where each band has an equal width in the Mel-scale frequency. In other words, each frame is transformed into a vector of 40 elements of mean energy within the bands. Moreover, the Mel-scale spectral flux can be defined as follows:

$$MelFlux(t_i) = \frac{1}{N}\sum_{j=1}^{N} HRF\left(MelSpectro(t_{i+1}, b_j) - MelSpectro(t_i, b_j)\right) \quad (1)$$

where t_i is the time for frame i, b_j is Mel-band j, $MelSpectro(t_i, b_j)$ is the Mel spectrogram at frame i and Mel-band j, and $HRF(\cdot)$ is the half-wave rectifier.

In general, we neglect the locally periodical information above 500 BPM (beats per minute) due to the limitation of human perception [7]. Thus we use Gaussian smoothing (which acts as a low-pass filter) to filter out the redundant high-frequency parts in raw novelty curve, as shown in Figure 2 (c). The Gaussian filter has a cutoff frequency equal to the sampling frequency divided by 5. At last, we subtract the local mean (dotted curve in Figure 2 (c)) to obtain the final novelty curve, as shown in Figure 2 (d). The local mean is derived from Gaussian smoothed raw novelty curve filtered by another Gaussian filter with a cutoff frequency equal to the sampling frequency divided by 125.

2.2 Tempo Curve Estimation

In this block, we estimate the tempo curve by analyzing locally periodical patterns in novelty curve. Generally speaking, local periodicity estimation is usually accomplished by STFT, FM-ACF or a combined method [2]. However, the autocorrelation-based method generates non-uniform tempo grids in tempogram, since the tempo is the inverse of the beat time difference. More specially, the lower the tempo is, the finer resolution (via interpolation, for instance) is required to achieve a high precision. To avoid such extra work for maintaining the precision, here we use STFT to obtain the tempo curve in our study.

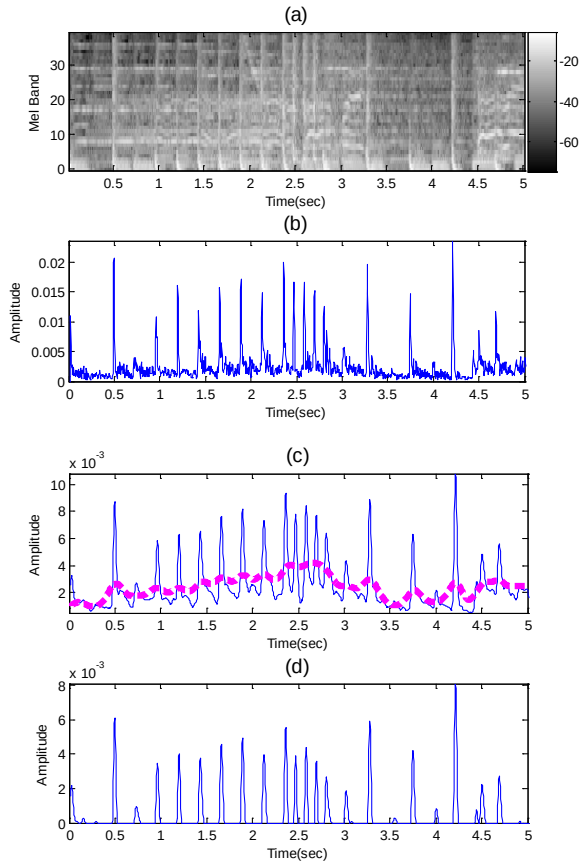


Figure 2. (a) Power spectrogram. (b) Raw novelty curve. (c) Smoothed novelty curve with local mean curve (the dash curve). (d) Novelty curve after local mean subtraction

As mentioned above, we do not have to analyze all frequency components in the novelty curve. Therefore, a partial FFT method is employed to eliminate high-frequency computation in STFT. Furthermore, the selection of analyzing window length significantly influences the capability for tracking tempo variation. In our implementation, the frame size is set to be 4 seconds with 99.6% overlap. The resulting tempogram is shown in Figure 3(a).

In order to strike a balance between tempo continuity and novelty curve strength, a DP-based approach is used to obtain the tempo curve. Given the magnitude $M_{i,j}$ of a point in the tempogram with time index i ($1 \le i \le n$), we want to find a tempo path $\mathbf{P} = [\boldsymbol{p_1}, \cdots, \boldsymbol{p_i} \cdots, \boldsymbol{p_n}]$, with p_i is tempo value, such that the over-all objective utility function is maximized:

$$\mathrm{J}(\mathbf{P}, \theta) = \sum_{i=1}^{n} M_{i,p_i} - \theta \times \sum_{i=1}^{n-1} |p_i - p_{i+1}|, \qquad (2)$$

where θ is the transition penalty factor incurred by the difference of the tempo path within two consecutive frames. The first term in the utility function is the magnitude values along the path over the tempogram, while the second term controls the smoothness of the path (thus the computed tempo curve). If θ is larger, then the tempo curve will be smoother. In particular, if θ = 0 in the extreme case, then maximizing the utility function is equivalent to maximum-picking of each column (or equivalently, each frame) of the tempogram.

For efficiency, we shall employ DP to find the maximum of the utility function, where the optimum-valued function $D(i,j)$ is defined as the maximum utility starting from frame 1 to i, with the frequency/tempo index ending at j ($1 \le j \le m$). Then the recurrent equation for DP can be formulated as follows:

$$D(i,j) = M_{i,j} + max_{k,j\in[1,m]}\{D(i-1,k) - \theta \times td(k,j)\} \qquad (3)$$
where $i \in [2,n], k\ and\ j$ are tempo index
$td(\cdot)$ is tempo difference function

The initial conditions are

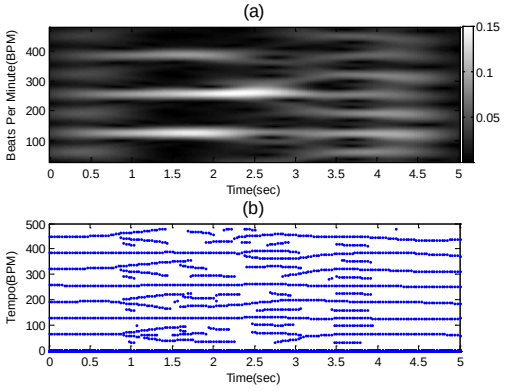
$$D(1,j) = M_{1,j},\ j \in [1,m] \qquad (4)$$

And the maximum utility is equal to $MAX_{j\in[1,m]}D(n,j)$. A similar DP-based pitch tracking method has been proposed for tone recognition in our previous work [13].

In practice, we can replace $td(k,j)$ in the recurrent equation with $td(k,j) = |p_k - p_j|$, which represents the tempo difference between tempo indices k and j. This is adopted in our implementation. Figure 4 demonstrates typical results of DP over a tempogram, with (a) and (b) being the tempogram M and the DP table D, respectively, together with the optimum path obtained via DP. Figure 4 (c) and (d) shows the same plots using a 3D surface for easy visualization.

As a common practice in DP, after the maximum utility is found, we can backtrack to find the optimum path together with the most likely tempo curve, as shown in Figure 3(b).

The transition penalty factor θ controls tempo variations, that is, it determines the smoothness of tempo curve, as shown in Figure 3(c), where a larger value of θ leads to a smoother tempo curve. In our experiment, the transition penalty factor θ is set to 0.01 empirically in order to track the correct tempos.



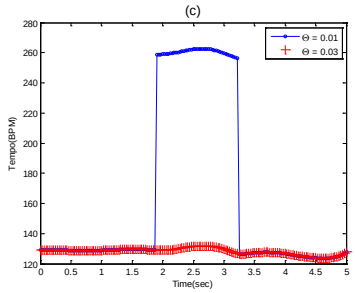


Figure 3. (a) The tempogram obtained from the novelty curve (b) Local maxima of each column of the tempogram and the final optimum tempo path (solid line) with θ = 0.01 (c) Tempo curves obtained with θ = 0.01 and 0.03, respectively.

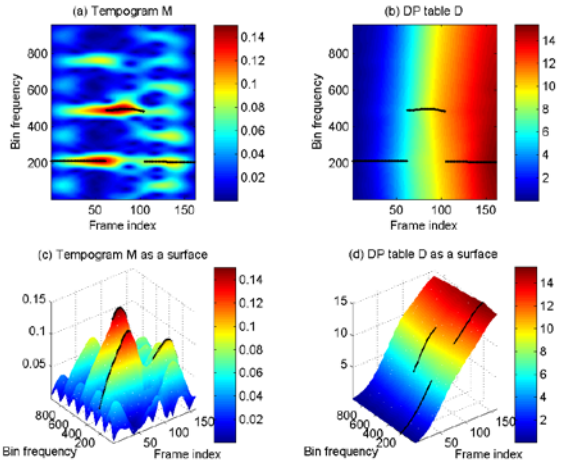


Figure 4. (a) Tempogram (as a contour map) and the optimum path. (b) DP table (as a contour map) and the optimum path. (c) Tempogram (as a 3D surface) and the optimum path. (d) DP table (as a 3D surface) and the optimum path.

When n is big, the computational complexity is still too high to compute the recurrent equation over all states. To reduce the computation, we can simply pick the L largest local maxima within each column of the tempogram as the candidate states for DP, as shown in Figure 3 (b). In our experiment, this simplified algorithm with L equal to 10 can achieve almost the same performance as the original DP.

2.3 Beat Tracking

This block utilizes both the tempo curve and the novelty curve to find a sequence of beat positions that fits the tempo curve and the novelty strengths as much as possible. To achieve this task, we apply another DP-based method in a probabilistic framework (just like Viterbi search in speech recognition) to perform forward and backward beat tracking, starting from the anchor beat position (the position of the most prominent peak) of the novelty curve. We have proposed such a probability-based DP framework for pitch mark identification [14]. Another DP-based approach has been proposed for stable-tempo beat tracking [1], though not in a probabilistic framework.

Here we use Figure 5 to explain the weighting-based DP method for beat position identification. First of all, we find the maximum of the novelty curve as the first beat position, which is referred to as the anchor candidate. Starting from the anchor candidate, we search on both sides, one side at a time, to obtain all beat positions. The search region is generally defined as a range from 0.2 to 2.2 times T, the beat period at the anchor candidate. We use a log-time Gaussian function over the search region as a weighting window for approximating the transition probability. Note that the maximum of the log-time Gaussian window is located at T from the anchor candidate.

In practice, only the largest N peaks of the novelty curve within the next search region are selected as the candidates for the next beat positions. As a result, we need to perform normalization to guarantee that the transition probabilities sum to 1 within the search region. Similarly, the state probabilities of these N candidates are obtained based on their heights within the novelty curve.

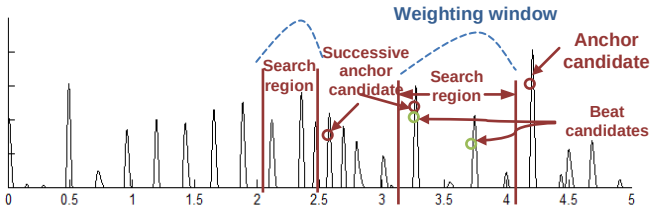


Figure 5. Backward beat search with N =2

Once the state and transition probabilities are defined, we can apply DP just like Viterbi search for the optimum beat positions. The search is performed twice for both forward and backward directions from the anchor candidate, and the results of them are merged to obtain the complete beat positions. In our experiment, we set N to 2. Figure 6 shows a typical result with θ = 0.01 and N=2.

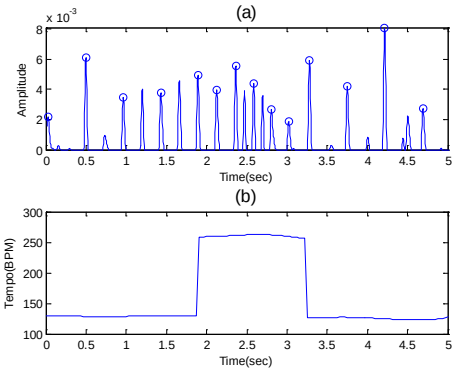


Figure 6. (a) Typical beat tracking results with θ = 0.01 and N=2 (beat positions indicated by circles). (b) The tempo curve used to obtain the beat positions in (a).

3. PERFORMANCE EVALUATION

In this section, we present the performance of the proposed algorithm by using the results of the Audio Beat Tracking contest in Music Information Retrieval Evaluation eXchange (MIREX) 2010 [8].

3.1 Performance Indices

There are a number of performance indices proposed for the audio beat tracking task in MIREX 2010 [8]. For simplicity, here we explain two performance indices which are generally adopted in beat tracking evaluation among all. The first one is F-measure [9] which considers the estimated beat as correct if it is within a tolerance window (±70ms in MIREX 2010) around the ground truth. The second one is P-score [10] which measures beat tracking accuracy by the summation of the cross-correlation between impulse trains of the estimated beats and the ground truth.

3.2 Datasets

Two music data sets are used to evaluate the performance of the proposed system with stable and time-varying tempo, respectively.

- MCK dataset:
  - Collected by Martin F. McKinney and Dirk Moelants.
  - Contains 160 30-second excerpts.
  - Ground truth is annotated as stable tempo.
  - A large variety of instrumentation and musical styles.

- MAZ dataset:
  - Collected by Craig Sapp.
  - A subset of 367 Chopin Mazurka pieces [10].
  - Ground truth is annotated as time-varying tempo.

3.3 Performance and discussion

Tables 1 and 2 show the performance of participating teams in MIREX 2010 audio beat tracking task on stable and time-varying tempo respectively. Only the best methods from each team are listed here. Algorithm TL2 uses the proposed method in this paper. LGG2 and MRVCC1 accomplish this task based on BeatRoot system proposed by Simon Dixon [5]. NW1 is based on Predominant Local Pulse curves (PLP) [6]. GP3 estimates beat and downbeat positions [11] simultaneously via an inverse Viterbi formulation and LDA-trained beat-template [3]. ZTC1 tracks beat with a global stable tempo value. BES4 is based on bidirectional Long Short-Term Memory (BLSTM) recurrent neural networks.

Algorithm ID	TL2	LGG2	MRVCC1	NW1	GP3	ZTC1	BES4
F-Measure	42.0	50.0	25.7	35.6	50.3	1.2	54.5
P-Score	50.6	55.0	38.4	45.7	56.5	0.9	59.2

Table 1. Performance on MCK dataset (stable tempos)

As shown in Table 1, the proposed algorithm (TL2) only performs moderately well on MCK dataset which has stable tempos. The performance in this dataset indicates we might have put too much emphasis on tracking tempo variations instead of identifying stable tempos. In other words, we might want to increase the value of the transition penalty factor θ such that the tempo variations can be kept small for this dataset.

Algorithm ID	TL2	LGG2	MRVCC1	NW1	GP3	ZTC1	BES4
F-Measure	68.5	41.5	49.2	27.6	47.1	24.6	58.7
P-Score	72.2	43.5	51.0	31.4	48.7	26.1	57.9

Table 2. Performance on MAZ dataset(time-varying tempos)

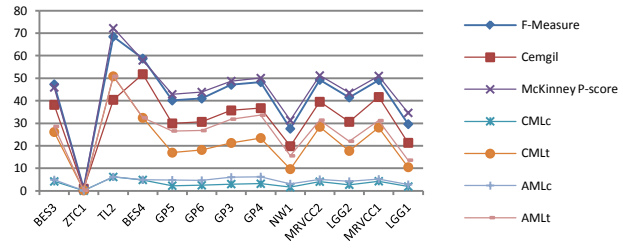


Figure 7. The performance on MAZ dataset, including all submitted algorithms and 7 performance measures.

On the other hand, in Table 2, the proposed algorithm (TL2) outperforms all the other teams based on the performance indices of F-measure and P-score. More specifically, if we consider all the submitted algorithms and all the performance measures, the proposed algorithm outperforms other 12 submitted algorithms on 6 performance indices out of 9, as shown in Figure. 7. (Note that in the Figure, we only show 7 performance indices for clarity. Moreover, the performance measure by Goto is not counted since it is close to zero for all submitted algorithms.) This clearly demonstrates the feasibility of the proposed two-fold DP strategy for dealing with music of time-varying tempos.

4. CONCLUSIONS

In this paper, we have proposed a two-fold DP approach to beat tracking, especially for time-varying tempo music. The first DP is applied to estimate the tempo curve from the tempogram, and the second DP is used to find the optimum beat positions with maximum likelihood. The proposed method is very similar to our previous work on speech analysis, where the first DP is used for robust pitch determination [13] and the second DP for robust pitch marking [14]. Based on the results of the audio beat tracking contest of MIREX 2010, the proposed method performs extremely well for music with time-varying tempos, but only moderately well for music with stable tempos. To improve

the proposed algorithm, our immediate work is to use a training based method to select the transition penalty factor θ such that it can deal with music with both stable and time-varying tempos. Moreover, we would like to develop a more systematic way of defining the state and transition probabilities used for the second-fold DP for finding the optimum beat positions. We will also investigate the possibility of incorporating more acoustic features, either time- or frequency-domain, to define the more robust novelty curve that can deal with music with no percussions.

5. ACKNOWLEDGMENTS

The third author would like to acknowledge the sponsor of this work by "III Innovative and Prospective Technologies Project" of the Institute for Information Industry which is subsidized by the Ministry of Economy Affairs of the Republic of China.

6. REFERENCES

[1] D.P.W. Ellis, "Beat Tracking by Dynamic Programming," *Journal of New Music Research*, Vol. 36(1), 51–60, 2007.

[2] G. Peeters, "Template-based Estimation of Time-Varying Tempo," *EURASIP Journal on Advances in Signal Processing*, Vol. 2007, 158–171, 2007.

[3] G. Peeters, "Beat-marker Location Using a Probabilistic Framework and Linear Discriminant Analysis," in *Proc. DAFX*, Como, Italy, 2009.

[4] A.T. Cemgil, B. Kappen, P. Desain, and H. Honing, "On Tempo Tracking: Tempogram Representation and Kalman Filtering" *Journal of New Music Research*, Vol. 28(4), 259-273, 2001.

[5] S. Dixon**,** "Automatic Extraction of Tempo and Beat from Expressive Performances," *Journal of New Music Research*, 30(1):39–58, 2001.

[6] P. Grosche and M. Müller, "A Mid-level Representation for Capturing Dominant Tempo and Pulse Information in Music Recordings," in *Proc. ISMIR*, pages 189–194, Kobe, Japan, 2009.

[7] J. Bilmes, "A Model for Musical Rhythm," in *Proc. ICMC*, San Francisco, USA, 1992.

[8] MIREX 2010 Audio Beat Tracking Contest Results. http://www.music-ir.org/mirex/wiki/2010:MIREX2010_Results

[9] M.F. McKinney, D. Moelants, M.E.P. Davies and A. Klapuri, "Evaluation of audio beat tracking and music tempo extraction algorithms," *Journal of New Music Research*, Vol. 36, no. 1, pp. 1-16, 2007.

[10] The Mazurka Project. http://www.mazurka.org.uk, 2010.

[11] G. Peeters and H. Papadopoulos, "Simultaneous Beat and Downbeat-tracking Using a Probabilistic Framework: Theory and Large-scale Evaluation," submitted to *IEEE. Trans. on Audio, Speech and Language Processing*, 2010.

[12] S. Dixon, "Onset Detection Revisited" in *Proc. the 9th Int. Conference on Digital Audio Effects (DAFx-06)*, Montreal, Canada, September 18-20, 2006

[13] Jiang-Chun Chen, J.-S. Roger Jang, "TRUES: Tone Recognition Using Extended Segments", *ACM Transactions on Asian Language Information Processing*, Vol. 7, No. 10,Aug 2008.

[14] Cheng-Yuan Lin, J.-S. Roger Jang, "A Two-Phase Pitch Marking Method for TD-PSOLA Synthesis", In *Proc. Interspeech 2004 - 8th International Conference on Spoken Language Processing (ICSLP)*, pp. 1189-1192, Korea, Oct 2004.

[15] Juan Pablo Bello, Laurent Daudet, Samer Abdallah, Chris Duxbury, Mike Davies, Mark B. Sandler, " A Tutorial on Onset Detection in Music Signals" *IEEE Transactions on Speech and Audio Processing*, Vol. 13, No. 5, September 2005

[16] Peter Grosche, Meinard M¨uller, "Computing Predominant Local Periodicity Information in Music Recordings" *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2009

[17] *Adam M. Stark, Matthew E. P. Davies, Mark D. Plumbley*, "Real-Time Beat-Synchronous Analysis of Musical Audio" *in Proc. the 12th Int. Conference on Digital Audio Effects (DAFx-09)*

[18] Anssi P. Klapuri, Antti J. Eronen, Jaakko T. Astola, "Analysis of the Meter of Acoustic Musical Signals" *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 14, No. 1, January 2006

[19] Matthew E. P. Davies, Mark D. Plumbley, "Context-Dependent Beat Tracking of Musical Audio" *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 15, No. 3, March 2007