MIREX 2012 AUDIO BEAT TRACKING EVALUATION: BEAT.E

Florian Krebs
Department of Computational Perception
Johannes Kepler University, Linz, Austria

Gerhard Widmer
Department of Computational Perception
Johannes Kepler University, Linz, Austria

ABSTRACT

In this paper, we present a Hidden Markov Model (HMM) based beat tracking system that simultaneously extracts downbeats, beat times, tempo, meter and rhythmic patterns. Our model builds upon the basic structure proposed by Whiteley et. al [9], which we further modified by introducing a new observation model: rhythmic patterns are learned directly from data, which makes the model adaptable to the rhythmical structure of any kind of music. The MIREX beat tracking evaluation - 30 results using ten measures and three datasets - placed our algorithm among the top three performing algorithms thirteen times and always inside the top ten.

1. INTRODUCTION

From its very beginnings, music has been built on temporal structure - a musical beat - to which humans have been able to synchronize via dance or musical instruments. We remain far from understanding the underlying principles of this synchronization - the perception of beat - and far from being able to replicate this phenomenon with a computer program.

From an application point of view, knowing the temporal structure of a music piece would be of great interest for a number of music-related applications such as content and performance analysis.

We define the musical *beat* as the joint of approximately equally spaced *beat times* at the most salient level of temporal structure, as it evokes human actions such as foot tapping, head nodding, and dancing.

We present a beat tracking system which models the statistical properties of the temporal structure of an audio signal. We use a HMM to model the time sequence of beats, tempo, meter, and rhythmic patterns and find the most likely (hidden) state sequence by using the Viterbi algorithm. We introduce a new observation model for the dynamic bar pointer model, which was proposed by Whiteley et. al [9].

The paper is structured as follows: We introduce the beat tracking system in section 2, describe the training and test dataset used in the evaluations in section 3, present and discuss the evaluation results in section 4, and finally draw conclusions and present ideas for future work in section 5.

2. MODEL ARCHITECTURE

2.1 Dynamic Bar Pointer Model

Proposed in [9], the dynamic bar pointer model assigns each time instance k of an audio file to the hidden states:

1. current position inside a bar $p_k \in [0, 1)$;
2. current velocity of the bar pointer $v_k \in [v_{min}, v_{max}]$;
3. current meter $\theta_k \in \{\theta_1, \theta_2, ...\theta_{N_\theta}\}$;
   e.g., $\theta_k \in \{3/4, 4/4\}$, and
4. current rhythmic template $r_k \in \{r_1, r_2, ...r_{N_r}\}$;

The conditional independence relations of the bar pointer model are shown in the dynamic Bayesian network in figure 1.

Hence, the state space consists of a mixture of continuous (p_k, v_k) and discrete variables (θ_k, r_k). To infer the sequence of hidden states exactly, the state space must be completely continuous with Gaussian dynamics (Kalman filter) or completely discrete (hidden Markov model) [1]. As both conditions are not met here, the hidden states can only be inferred approximately: we discretize the continuous variables p_k and v_k to N_p and N_v grid-points, which yields the discrete variables $\tilde{p}_k$ and $\tilde{v}_k$. Hence, the total number of discrete states is given by $N_s = N_p \times N_v \times N_\theta \times N_r$. Next, we combine all random variables in one vector $\mathbf{x}_k$, which yields

$$\mathbf{x}_k = [\tilde{p}_k, \tilde{v}_k, \theta_k, r_k]^T. \quad (1)$$

As the transformed model consists of only discrete hidden states and a single random variable $\mathbf{x}_k$ now, it reduces to a standard HMM and inference becomes feasible.

2.2 Transition model parameters

The transition model (dynamic bar pointer model) was specified in [9] and is not reviewed here further. For this submission we use the following parameters: $N_p = 1000$, $N_v = 21$, $N_\theta = 2$, $N_r = 2$, $framelength = 20ms$, $v_{min} = 50$ bpm, $v_{max} = 220$ bpm, $p_v = 0.02$, $p_\theta = 0$ and $p_r = 0.5$, where p_v, p_θ and p_r are the probability of a change in velocity, meter and rhythmic pattern respectively.

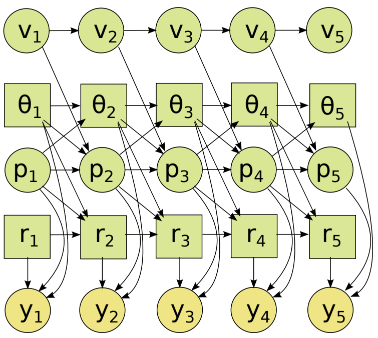

Figure 1. Dynamic Bayesian network

2.3 Observation model

The observation model relates the observations y_k (given by features extracted from the audio) to the hidden states $\mathbf{x}_k$. It is specified by the definition of an *audio feature* and a *likelihood function* $p(y_k|\mathbf{x}_k)$, which maps each state and feature value to a likelihood value.

2.3.1 Audio features

As the perception of beat depends strongly on the perception of the musical notes played, we believe that a good onset feature is also likely to be a good beat tracking feature. Therefore, we use the *LogFiltSpecFlux* onset feature z', which performed well in recent comparisons of onset detection functions [3, 4]. To compress the range of the feature, we subtract the moving average to obtain $z''(k)$ and then compress it using the following function:

$$y_k = \begin{cases} z''_k & \text{if } z''_k \leq \tau \\ \tau + \log[z''_k - \tau + 1] & \text{otherwise} \end{cases} \tag{2}$$

where τ is the threshhold above which compression is applied.

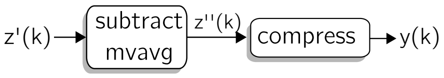

Figure 2. Feature computation

2.3.2 Rhythmic patterns

We expect the feature values to be higher at specific bar positions, depending on rhythmic pattern and meter. In the following we describe how we learn rhythmic patterns from the training data: as beat and bar annotations are available for our training set, each bar is divided into J_θ discrete positions (we chose $J_{3/4} = 48$ and $J_{4/4} = 64$). For each bar in the training data, we compute the mean feature value for each of the J_θ bar positions. This results in a matrix V of size $B_\theta \times J_\theta$, where B_θ is the number of bars with meter θ in the training set. This matrix can be decomposed into F non-negative basis vectors using non-negative matrix factorization (NMF) [7]:

$$V \approx W \times H \tag{3}$$

where H is a $F \times J_\theta$ matrix of the F basis factors, and W is a $B_\theta \times F$ matrix of weights that specifies the prominence of a basis factor in a bar.

An example is given in figure 3, where we show three basis factors obtained by NMF. From these three factors we manually selected the most characteristic ones. Criteria for automatical filtering of the factors could be defined by considering the variance or entropy of the basis factors. We chose the upper two factors in figure 3 because the factor at the bottom is a "residual" that compensates for the data that cannot be represented sufficiently by the upper two factors.

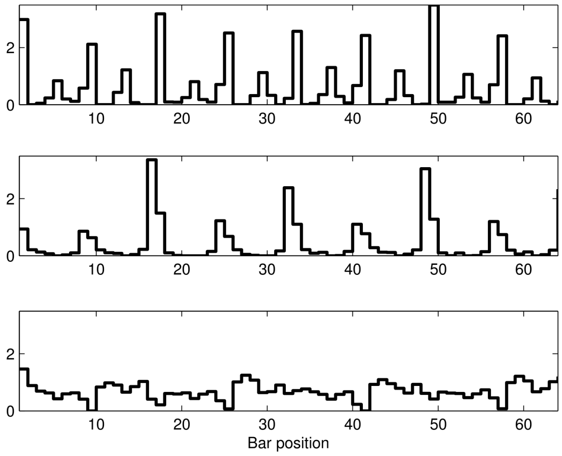

Figure 3. Three factors obtained by NMF of 1244 bars (4/4 meter)

Having selected the relevant rhythmic patterns from H, we assign each bar to its most prominent rhythmic pattern according to W and learn the parameters of the likelihood function for each pattern separately.

2.3.3 Likelihood function

The likelihood of being in state $\mathbf{x}_k$ while observing the feature value y_k is modeled by a set of Gamma distributions. Gamma distributions have also been used by other authors [8] and seem to be a reasonable choice when considering the distribution of feature values in figure 4: The left panel shows the distribution of feature values at a position with high onset frequency (first position in a bar), whereas the right panel shows a position where lower feature values appear more frequently (position 4/48 of a bar).

The likelihood function $p(y_k|\mathbf{x}_k)$ can therefore be written as

$$p(y_k|\mathbf{x}_k) = \Gamma(y_k; \zeta_\mathbf{x}, \theta_\mathbf{x}) \tag{4}$$

where ζ and θ are the shape and scale parameters of the Gamma function Γ. We fit one gamma distribution for each of the J_θ bar positions, each rhythmic pattern, and

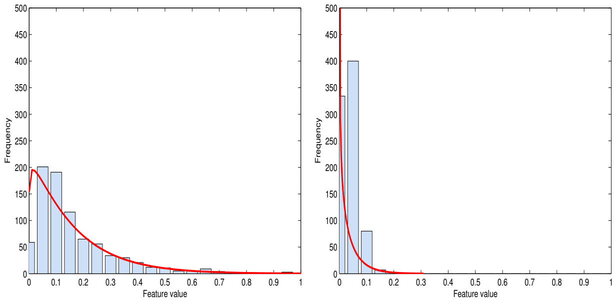

Figure 4. Histogram of feature values at bar positions 1 (left) and 4 (right) using a grid of 48 bins per bar

each meter, which yields $J_\theta \times \Theta \times R$ Gamma distributions. Hence, the number of parameters to be learned from the data is $2 \times J_\theta \times \Theta \times R$.

Figure 5 shows the mean values of the feature values that correspond to the upper two factors in figure 3. The pattern at the top is more likely to represent bars with energy at the eigth and sixteenth note level, whereas the pattern at the bottom reflects bars with strong beats at the quarter note level. Also, the “noise” represented by the basis factor at the bottom of figure 3 is now distributed among the two selected rhythmic patterns in figure 5.

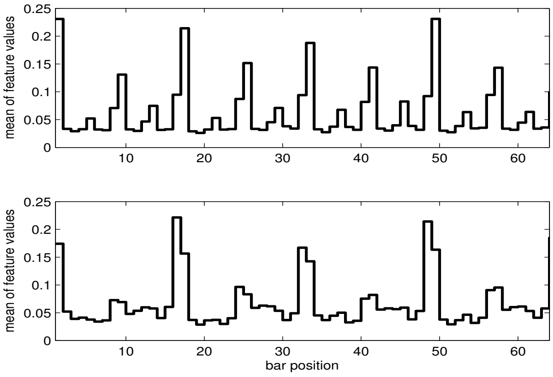

Figure 5. Mean values of Gamma functions that correspond to the upper two rhythmic patterns in figure 3

3. DATASETS

3.1 Training data

Our training set consists of 188 audio excerpts: 89 from the ISMIR 2004 tempo induction contest (also known as the ”Ballroom set”), 26 training and bonus files from the MIREX 2006 beat tracking contest, 6 musical pieces from [2], and 67 pieces from [4]. For each musical piece, the beat and its corresponding bar position on the beat grid (e.g., 1/4, 2/4, 3/4, 4/4 for a 4/4 meter) were annotated manually. The 188 files have a total length of 58.3 minutes and contain 6,469 annotated beats.

3.2 Test data

Currently, three evaluation datasets are used in the yearly Music Information Retrieval Evaluation eXchange (MIREX) for audio beat tracking. They are briefly described in this section:

3.2.1 MCK dataset

The MCK dataset contains 160 30-second audio excerpts and was created by the MIREX team in 2006. The recordings are characterized by a stable tempo and a wide variety of instrumentations and musical styles. About 20% of the files have non-binary meters.

3.2.2 MAZ dataset

The MAZ dataset contains piano recordings of 322 Chopin Mazurkas, which also include tempo changes. It was contributed by Craig Sapp in 2009.

3.2.3 SMC dataset

The third collection was contributed by Holzapfel et al [6] in 2012. It consists of 217 excerpts around 40 s each, of which the majority is difficult to track (e.g., because of changes in meter and tempo, bad sound quality, expressive timing). It includes romantic music, film soundtracks, blues, chanson, and solo guitar.

4. EVALUATION

4.1 Evaluation measures

The evaluation measures are specified in [5].

4.2 Results and discussion

The tables 2, 3, and 4 show the results of MIREX 2012 for the three datasets MCK, MAZ and SMC respectively. Note that the ranking of algorithms depends heavily on the evaluation measure used.

On the MCK dataset, our system outperforms all other systems in five of ten measures (*F-Measure*, *Cemgil*, *P-score*, *CMLt* and *Dg*), but scores worse in continuity-based measures (*AMLc*, *AMLt*).

On the MAZ dataset, our system performs best in three of ten measures and is only outperformed by *FW5* which scores best in the remaining seven performance measures.

On the SMC dataset, *KB1* and *GKC2* perform best in four of ten and our algorithm performs best in the remaining two measures.

As some of the datasets have been used for several years, we also compare the performance of our algorithm to all algorithms that have been submitted so far. Because many groups submitted their algorithms multiple times with different parameter settings, we rank only the best performing in each measure. This yields a total number of 22 different algorithms for the MCK dataset (2006, 2009-2012), 16 algorithms for the MAZ dataset (2009-2012) and 9 algorithms for the SMC dataset (2012),

Table 1 shows the results of our system on all three datasets and gives the ranking of all (different) algorithms submitted from 2006 to 2012 for each measure.

On the MCK dataset, our system outperforms all other systems in the measures *F-Measure, Cemgil* and *P-score*, but scores worse in continuity-based measures
(*CMLc, AMLc, AMLt*).

On the MAZ dataset, our system generally achieves lower rankings compared to the other datasets. This seems reasonable, as our training data has closer resemblance to the MCK dataset. In order to score highly on music with more frequent tempo changes, the system should be trained with piano music and music with varying tempi. Interestingly, the differences between CML_c and CML_t and also between AML_c and AML_t are bigger than for the other datasets. It seems that many beats are found in continuous segments, but these segments are very short and equally distributed along the audio track, which could be explained by the unstable tempo of the excerpts.

The SMC dataset appears to be the most "difficult" dataset of the three, as the results are the lowest of all three datasets in seven of ten measures (for the best algorithm of each measure). Our algorithm performs well according to the measures *F-Measure, Cemgil* and *Dg*, but again performs worse in terms of continuity-based measures (*AMLc, AMLt*).

More details of the task results can be found at
`www.music-ir.org/mirex/wiki/2012:MIREX2012_Results`.

5. CONCLUSION AND FUTURE WORK

We have introduced a HMM beat tracking system that was trained with real-world music data. Compared to all submissions to MIREX from 2006 to 2012, for all three datasets and all ten performance measures it is ranked thirteen (out of 30) times among the top three performing algorithms. As it was trained mainly with pop/rock recordings, it would be interesting to see if the performance on the MAZ dataset could be improved by training the system with piano music with varying tempi. We plan to add various features and rhythmic patterns. As this increases the computational complexity of the algorithm, other approximative inference methods such as particle filtering will be required.

6. ACKNOWLEDGMENTS

This research is supported by the Austrian Science Fund (FWF) under Z159 "Wittgenstein Award". We thank Ingrid Abfalter for correcting and Maarten Grachten for inspiring discussions.

7. REFERENCES

[1] M. Sanjeev Arulampalam, Simon Maskell, and Neil Gordon. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50:174–188, 2002.

[2] J.P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M.B. Sandler. A tutorial on onset detection in music signals. *IEEE Transactions on Speech and Audio Processing*, 13(5):1035–1047, 2005.

[3] S. Böck and F. Krebs. Mirex onset detection task. *8th Music Information Retrieval Evaluation eXchange (MIREX)*, 2012.

[4] S. Böck, F. Krebs, and M. Schedl. Evaluating the online capabilities of onset detection methods. In *Proc. ISMIR, Porto, Portugal*, 2012.

[5] M.E.P. Davies, N. Degara, and M.D. Plumbley. Evaluation methods for musical audio beat tracking algorithms. *Queen Mary University of London, Tech. Rep. C4DM-09-06*, 2009.

[6] A. Holzapfel, M.E.P. Davies, J.R. Zapata, J.L. Oliveira, and F. Gouyon. Selective sampling for beat tracking evaluation. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(9):2539–2548, 2012.

[7] C.J. Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19(10):2756–2779, 2007.

[8] Y. Shiu and C.C.J. Kuo. A hidden markov model approach to musical beat tracking. In *Proc. ICASSP, Las Vegas, USA*, 2008.

[9] N. Whiteley, A.T. Cemgil, and S. Godsill. Bayesian modelling of temporal structure in musical audio. In *Proceedings of the 7th International Conference on Music Information Retrieval*, pages 29–34, 2006.

Dataset		F-Measure	Cemgil	Goto	P-Score	CMLc	CMLt	AMLc	AMLt	D (bits)	Dg (bits)
MCK	Results FK1	56.7	42.7	21.4	61.2	22.3	35.1	41.5	63.3	1.77	0.31
	Results best	56.7	42.7	22.6	61.2	26.4	35.6	51.8	66.7	1.87	0.39
	Rank FK1	**1**	**1**	5	**1**	8	2	10	7	6	3
MAZ	Results FK1	58.4	48.0	2.5	56.6	5.1	30.1	9.0	40.4	0.77	0.45
	Results best	68.5	61.5	2.5	72.2	7.8	50.9	9.7	50.9	2.93	1.95
	Rank FK1	5	4	2	4	4	7	3	4	7	6
SMC	Results FK1	39.7	30.5	7.8	50.0	14.2	22.3	22.9	36.7	0.99	0.19
	Results best	40.7	30.5	10.1	51.7	17.7	26.8	26.6	45.1	1.02	0.19
	Rank FK1	2	2	7	3	3	3	6	5	6	**1**

Table 1. Results of our algorithm (FK1), results of the best performing algorithm per measure and ranking of FK1 among all different submissions to MIREX 2006-2012

.

Algorithm	F-Measure	Cemgil	Goto	P-Score	CMLc	CMLt	AMLc	AMLt	D (bits)	Dg (bits)
FK1	**56.7**	**42.7**	21.4	**61.2**	22.3	**35.1**	41.5	63.3	1.77	**0.31**
KB1	53.5	39.6	17.5	57.7	17.5	29.9	35.9	60.2	1.62	0.23
ODGR1	50.5	38.2	17.8	55.5	21.6	30.0	49.4	64.1	1.66	0.26
FW4	52.1	39.5	21.6	57.7	23.7	34.5	42.4	59.1	1.64	0.26
KFRO1	51.1	39.0	20.7	56.0	25.0	32.0	47.1	58.8	1.66	0.29
ZDG2	53.4	40.6	**22.4**	58.2	25.0	33.4	**51.8**	**66.7**	**1.81**	**0.31**
GKC2	50.1	37.8	19.0	55.2	**25.8**	32.9	51.0	64.2	1.69	0.27
SB6	52.9	40.3	18.8	56.8	20.4	29.3	40.8	57.2	1.60	0.25
GP3	50.3	37.3	21.2	56.6	24.0	33.7	49.3	66.4	1.78	0.25

Table 2. Results MIREX 2012 of the MCK dataset

Algorithm	F-Measure	Cemgil	Goto	P-Score	CMLc	CMLt	AMLc	AMLt	D (bits)	Dg (bits)
FK1	58.4	48.0	**2.48**	56.6	5.06	30.13	**8.95**	**40.4**	0.77	0.45
ODGR3	44.3	33.1	0.00	46.7	2.78	19.78	4.27	24.5	0.26	0.11
FW5	**66.6**	**52.7**	0.62	**63.9**	**7.04**	**36.43**	8.37	39.8	**1.44**	**0.67**
ZDG1	51.4	43.6	0.93	52.3	4.88	29.53	6.31	33.3	0.56	0.31
GKC2	42.2	33.5	0.00	41.6	2.21	15.58	5.07	26.0	0.34	0.15
KB1	52.3	39.9	0.62	53.0	4.03	30.65	4.60	32.7	0.37	0.19
SB7	50.4	47.6	0.00	50.9	4.62	27.53	4.67	27.6	0.92	**0.67**
GP4	55.6	44.2	0.00	56.9	4.37	28.79	6.96	34.8	0.87	0.33

Table 3. Results MIREX 2012 of the MAZ dataset

Algorithm	F-Measure	Cemgil	Goto	P-Score	CMLc	CMLt	AMLc	AMLt	D (bits)	Dg (bits)
FK1	39.7	**30.5**	7.8	50.0	14.2	22.3	22.9	36.7	0.99	**0.19**
ODGR2	31.9	24.6	6.5	45.5	10.4	14.3	18.0	27.9	0.76	0.07
FW5	34.6	25.6	1.4	44.4	3.5	8.3	6.1	18.4	0.66	0.07
ZDG2	37.1	28.5	**10.1**	47.7	12.7	17.1	23.6	37.6	0.99	0.13
GKC2	36.6	27.9	9.2	**51.7**	**17.7**	**26.8**	24.4	40.0	**1.02**	0.13
KB1	**40.7**	**30.5**	6.9	50.0	12.8	19.2	**26.6**	**45.1**	1.00	0.15
SB7	37.5	29.4	2.3	47.1	8.8	15.1	12.6	24.1	0.75	0.11
GP2	36.4	27.2	**10.1**	47.6	13.4	20.1	23.6	38.6	0.99	0.12

Table 4. Results MIREX 2012 of the SMC dataset