MIREX 2012 AUDIO AUDIO TEMPO ESTIMATION EVALUATION: TEMPOKREB

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, a system is presented that simultaneously extracts downbeats, beats, tempo, meter and rhythmic patterns, using a Hidden Markov Model (HMM) framework. The basic structure of the model was proposed by Whiteley et. al [7] and was further modified by introducing a new observation model: Rhythmic patterns are learned from data way and make the model adaptable to the rhythmical structure of a certain kind of music. The MIREX tempo estimation evaluation shows average results measured by the P-score, but outperforms (together with GKC3) all other submissions between 2006 and 2012 in the category "both tempi correct".

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 an 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 use the model proposed by Whiteley et. al [7] and introduce a new observation model for audio data.

The paper is structured as follows: We present the system in section 2, describe the training and test dataset that has been 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 [7], 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

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

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 [7] 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.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page.
© 2012 International Society for Music Information Retrieval.

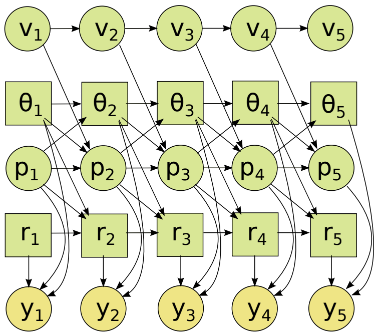

Figure 1. Dynamic Bayesian network

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 played musical notes, we believe that a good onset feature is likely to be also a good beat tracking feature. Therefore, we use the *LogFiltSpecFlux* onset feature z', that performed well in recent comparisons of onset detection functions [3,4]. To compress the range of the feature, we subtract the moving average to yield $z''(k)$ 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.

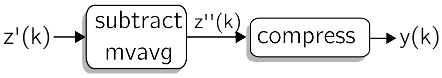

Figure 2. Feature computation

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) [5]:

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

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.

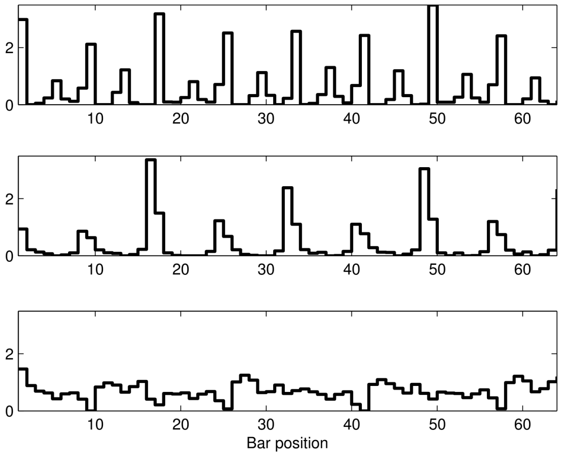

Figure 3. Three factors obtained by NMF of 1244 bars (4/4 meter)

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 [6] 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

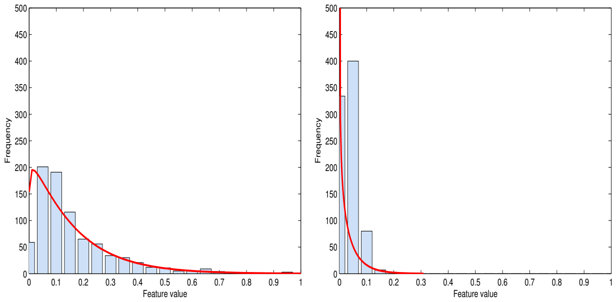

Figure 4. Histogram of feature values at bar positions 1 (left) and 4 (right) using a grid of 48 bins per bar

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.

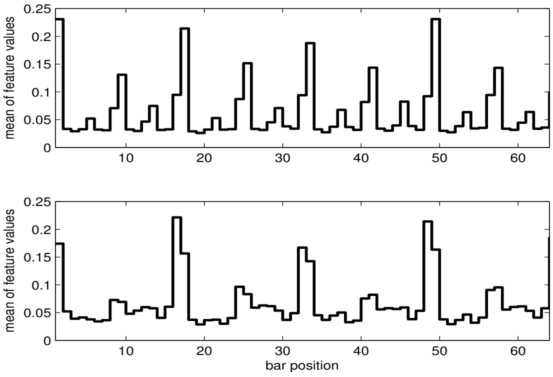

Figure 5. Mean values of Gamma functions that correspond to the upper two rhythmic patterns in figure 3

2.4 Finding the tempo estimates

The optimal hidden state sequence in the maximum a posteriori probability sense is found via Viterbi path. The most prominent, average tempo (tempo1) is found by computing the median of the tempo path. The second tempo estimate (tempo2) is computed via the following heuristic:

Algorithm 1 Compute tempo2

```
if tempo1 $<$ 100 bpm then
    tempo2 = 2 $\times$ tempo1
else
    tempo2 = $\frac{1}{2}\times$ tempo1;
end if
```

3. DATASETS

3.1 Training data

Our training set consists of 188 audio excerpts, where 89 are taken from the ISMIR 2004 tempo induction contest (also known as the "Ballroom set"), the 26 training and bonus files from the MIREX 2006 beat tracking contest, 6 musical pieces from [2] and 67 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) are manually annotated. The 188 files have a total length of 58.3 minutes and contain 6,469 annotated beats.

3.2 Test data

The MIREX06 Tempo dataset contains 160 30-second long audio excerpts and was created by the MIREX team in 2006. The recordings are characterized by a stable tempo and a broad variety of instrumentation and musical styles. About 20% of the files have non-binary meters.

4. EVALUATION

4.1 Evaluation measures

The evaluation measures are specified in [**?**].

4.2 Results and discussion

As the MIREX06 Tempo dataset has been used for several years now, we compare the performance of our algorithm to all algorithms that have been submitted so far. Because many groups submit the same algorithm with various parameter settings, we only use the best performing one in each measure for the ranking. This yields a total number of 18 different algorihms for the MIREX06 Tempo dataset in the evaluations of the years 2006 and 2010 to 2012.

Table 1 shows the results of the proposed system and the ranking for each measure considering all submitted (different) algorithms from 2006 to 2012.

Our system shows average performance as measured by the P-score and ranks 9th of 18 algorithms. Interestingly, it outperforms (together with GKC3) all other systems when both tempi have to be identified correctly. This means that in many cases, where the algorithm finds one correct tempo, the heuristic of section 2.4 succeeds to find the correct second tempo as well.

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

5. CONCLUSION AND FUTURE WORK

We presented a beat tracking and tempo estimation system that was trained on real-world music data. Compared to all submissions to MIREX from 2006 to 2012, it achieves average performance, but under the condition "both tempi correct" it manages to outperform (together with GKC3) all other submissions. In future, we would like to add various features and rhythmic patterns. As this will increase the computational complexity of the algorithm we

	P-score	At least 1 tempo correct	Both tempi correct
Results FK2	0.7474	0.8500	0.6214
Results best	0.8290	0.9643	0.6214
Rank FK2	9	13	**1**

Table 1. Results of proposed algorithm (FK2), results of the best performing algorithm per measure and ranking of FK2 for all different submissions to MIREX 2006-2012

.

will have to refer to other approximative inference methods such as particle filtering.

6. ACKNOWLEDGMENTS

This research is supported by the Austrian Science Fund (FWF) under Z159 “Wittgenstein Award”. We thank Ingrid Abfalter for correcting.

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

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

[7] 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.