

# BAYESIAN METER TRACKING ON LEARNED SIGNAL REPRESENTATIONS

Andre Holzapfel, Thomas Grill

Austrian Research Institute for Artificial Intelligence (OFAI)

andre@rhythmos.org, thomas.grill@ofai.at

## ABSTRACT

Most music exhibits a pulsating temporal structure, known as *meter*. Consequently, the task of *meter tracking* is of great importance for the domain of Music Information Retrieval. In our contribution, we specifically focus on Indian art musics, where meter is conceptualized at several hierarchical levels, and a diverse variety of metrical hierarchies exist, which poses a challenge for state of the art analysis methods. To this end, for the first time, we combine Convolutional Neural Networks (CNN), allowing to transcend manually tailored signal representations, with subsequent Dynamic Bayesian Tracking (BT), modeling the recurrent metrical structure in music. Our approach estimates meter structures simultaneously at two metrical levels. The results constitute a clear advance in meter tracking performance for Indian art music, and we also demonstrate that these results generalize to a set of Ballroom dances. Furthermore, the incorporation of neural network output allows a computationally efficient inference. We expect the combination of learned signal representations through CNNs and higher-level temporal modeling to be applicable to all styles of metered music, provided the availability of sufficient training data.

## 1. INTRODUCTION

The majority of musics in various parts of the world can be considered as metered, that is, their temporal organization is based on a hierarchical structure of pulsations at different related time-spans. In Eurogenetic music, for instance, one would refer to one of these levels as the beat or tactus level, and to another (longer) time-span level as the downbeat, measure, or bar level. In Indian art musics, the concepts of *tāla* for Carnatic and *tāl* for Hindustani music define metrical structures that consist of several hierarchical

AH is supported by the Austrian Science Fund (FWF: M1995-N31).

TG is supported by the Vienna Science and Technology Fund (WWTF) through project MA14-018 and the Federal Ministry for Transport, Innovation & Technology (BMVIT, project TRP 307-N23).

We would like to thank Ajay Srinivasamurthy for advice and comments. We also gratefully acknowledge the support of NVIDIA Corporation with the donation of a Tesla K40 GPU used for this research.



© Andre Holzapfel, Thomas Grill. Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). **Attribution:** Andre Holzapfel, Thomas Grill. "Bayesian meter tracking on learned signal representations", 17th International Society for Music Information Retrieval Conference, 2016.

levels. However, important differences between meter(s) in Eurogenetic and Indian art musics are the presence of non-isochronicity in some of the metrical layers and the fact that an understanding of the progression of the meter is crucial for the appreciation of the listener, see, e.g. [3, p. 199ff]. Again, other cultures might not explicitly define metrical structure on several layers, but just define certain rhythmic modes that determine the length of a metrical cycle and some points of emphasis within this cycle, as is the case for Turkish makam music [2] or Korean music [13]. Common to all metered musics is the fact that the understanding of only one metrical level, such as the beat in Eurogenetic music, leads to an inferior understanding of the musical structure compared to an interpretation on several metrical layers; a couple dancing a Ballroom dance without a common understanding of beat and bar level will end up with four badly bruised feet, while a whirling dervish in Turkey who does not follow the long-term structure of the rhythmic mode will suffer pain of a rather spiritual kind.

Within the field of Music Information Research (MIR), the task of beat tracking has been approached by many researchers, using a large variety of methodologies, see the summary in [14]. Tracking of meter, i.e., tracking on several hierarchically related time-spans, was pursued by a smaller number of approaches, for instance by [9]. [15] were among the first to include experiments that document the importance of adapting a model automatically to musical styles in the context of meter tracking. In recent years, several approaches to beat and meter tracking were developed that include such adaptation to musical style, for instance by applying dynamic Bayesian networks [12] or Convolutional Neural Networks (CNN) [6] for meter tracking, or by combining Bayesian networks with Recurrent Neural Networks (RNN) for beat tracking in [1].

In this paper, we combine deep neural network and Bayesian approaches for meter tracking. To this end, we adapt an approach based on CNN that was previously applied to music segmentation with great success [18]. To the best of our knowledge, no other applications of CNNs to the task of combined tracking at several metrical levels have yet been published, although other groups apply CNN as well [6]. In this paper, the outputs of the CNN, i.e., the activations that imply probabilities of observing beats and downbeats<sup>1</sup>, are then integrated as observations into a dynamic Bayesian network. This way, we explore in how far an approach [18] previously applied to supra-metrical

<sup>1</sup> We use these terms to denote the two levels, for the sake of simplicity.

Dance	#Pieces	Cycle Length: mean (std)
Cha cha (4/4)	111	1.96 (0.107)
Jive (4/4)	60	1.46 (0.154)
Quickstep (4/4)	82	1.17 (0.018)
Rumba (4/4)	98	2.44 (0.274)
Samba (4/4)	86	2.40 (0.177)
Tango (4/4)	86	1.89 (0.064)
Viennese Waltz (3/4)	65	1.01 (0.015)
Waltz (3/4)	110	2.10 (0.077)

**Table 1:** The Ballroom dataset. The columns depict time signature with the names of the dances, number of pieces/excerpts, and mean and standard deviation of the metrical cycle lengths in seconds.

structure in music can serve to perform meter tracking as well. Furthermore, we want to evaluate in how far the meter tracking performed by the CNN can be further improved by imposing knowledge of metrical structure that is expressed using a Bayesian model. The evaluation in this paper is performed on Indian musics as well as Latin and international Ballroom dances. This choice is motivated by the fact that meter tracking in Indian musics revealed to be particularly challenging [8], but at the same time a novel approach should generalize to non-Indian musics. Our results improve over the state of the art in meter tracking on Indian music, while results on Ballroom music are highly competitive as well.

We present the used music corpora in Section 2. Section 3 provides detail on the CNN structure and training, and Section 4 on the Bayesian model and its combination with the CNN activations. In both sections we aim at providing a concise presentation of both methods, emphasizing the novel elements compared to previously published approaches. Section 5 illustrates our findings, and Section 6 provides a summary and directions for future work.

## 2. MUSIC CORPORA

For the evaluation of meter tracking performance, we use two different music corpora. The first corpus consists of 697 monaural excerpts ( $f_s = 44.1$  kHz) of Ballroom dance music, with a duration of 30 s for each excerpt. The corpus was first presented in [5], and beat and bar annotations were compiled by [10]. Table 1 lists all the eight contained dance styles and their time signatures, and depicts the mean durations of the metrical cycles and their standard deviations in seconds. In general, the bar durations can be seen to have a range from about a second (Viennese Waltz) to 2.44 s (Rumba), with small standard deviations.

The second corpus unites two collections of Indian art music that are outcomes of the ERC project CompMusic. The first collection, the Carnatic music rhythm corpus contains 176 performance recordings of South Indian Carnatic music, with a total duration of more than 16 hours.<sup>2</sup> The second collection, the Hindustani music rhythm corpus,

<sup>2</sup> <http://compmusic.upf.edu/carnatic-rhythm-dataset>

Carnatic		
Tāḷa	#Pieces	Cycle Length: mean (std)
Adi (8/4)	50	5.34 (0.723)
Rūpaka (3/4)	50	2.13 (0.239)
Mīśra chāpu (7/4)	48	2.67 (0.358)
Khanda chāpu (5/4)	28	1.85 (0.284)
Hindustani		
Tāl	#Pieces	Cycle Length: mean (std)
Tintāl (16/4)	54	10.36 (9.875)
Ektāl (12/4)	58	30.20 (26.258)
Jhaptāl (10/4)	19	8.51 (3.149)
Rūpak tāl (7/4)	20	7.11 (3.360)

**Table 2:** The Indian music dataset. The columns depict time signature with the names of the Tāḷa/tāl cycles, the number of pieces/excerpts, and mean and standard deviation of the metrical cycle lengths in seconds.

contains 151 excerpts of 2 minutes length each, summing up to a total duration of a bit more than 5 hours.<sup>3</sup> All samples are monaural at  $f_s = 44.1$  kHz. Within this paper we unite these two datasets to one corpus, in order to obtain a sufficient amount of training data for the neural networks described in Section 3. This can be justified by the similar instrumental timbres that occur in these datasets. However, we carefully monitor the differences of tracking performance for the two musical styles. As illustrated in Table 2, metrical cycles in the Indian musics have longer durations with large standard deviations in most cases. This difference is in particular accentuated for Hindustani music, where, for instance, the Ektāl cycles range from 2.23 s up to a maximum of 69.73 s. This spans five tempo octaves and represents a challenge for meter tracking. The rhythmic elaboration of the pieces within a metrical class varies strongly depending on the tempo, which is likely to create difficulties when using the recordings in these classes for training one unified tracking model.

## 3. CNN FOR METER TRACKING

CNNs are feed-forward networks that include *convolutional layers*, computing a convolution of their input with small learned filter kernels of a given size. This allows processing large inputs with few trainable parameters, and retains the input’s spatial layout. When used for binary classification, the network usually ends in one or more dense layers integrating information over the full input at once, discarding the spatial layout. The architecture for this work is based on the one used by Ullrich et al. [18] on MLS (Mel-scaled log-magnitude spectrogram) features for their MIREX submission [16]. Therein, CNN-type networks have been employed for the task of musical structure segmentation. [7] have expanded on this approach by introducing two separate output units, yielding predictions for ‘fine’ and ‘coarse’ segment boundaries. For the research at hand, we can use this architecture to train and predict

<sup>3</sup> <http://compmusic.upf.edu/hindustani-rhythm-dataset>

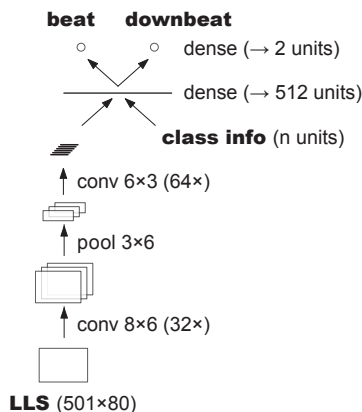


Figure 1: The CNN architecture in use.

beats and downbeats in the same manner with two output units, enabling the network to exploit information shared between these two temporal levels.

### 3.1 Data

For both datasets under examination, we use a train/validation/test split. The sizes are 488/70/140 for the ballroom data set and 228/33/66 for the combination of the two Indian data sets. From the audio files, we compute log-scaled logarithmic-magnitude spectrograms (LLS) of 80 bands (instead of mel-scaled MLS in [18]), ranging from 80 Hz to 16 kHz. We have found log-scaled features to work better in early stages of research, most probably because of their harmonic translational invariance, supporting the convolutional filters. The STFT size used is 2048, with a frame rate of 100 fps. In order to be able to train and predict on spectrogram excerpts near the beginning and ending of a music piece, we apply a simple padding strategy for the LLS features. If the first (or last, respectively) non-zero spectrogram frame has a mean volume of  $\geq -40$  dBFS, we assume an abrupt boundary and pad the spectrogram with a  $-100$  dBFS constant. Conversely, we pad with repeated copies of this first or last non-zero spectrogram frame. To either padding, we add  $\pm 3$  dB of uniform noise to avoid unnatural spectral clarity. Over the entire data sets, we normalize to zero mean and unit variance for each frequency band, yielding a suitable range of input values for the CNN.

### 3.2 Network Structure and training

Figure 1 shows the network architecture used for our experiments, unchanged from our previous experiments in [18]. On the input side, the CNN sees a temporal window of 501 frames with 80 frequency bands, equivalent to 5 seconds of spectral information. The LLS input is subjected to a convolutional layer of 32 parallel  $8 \times 6$  kernels (8 time frames and 6 frequency bands), a max-pooling layer with pooling factors of  $3 \times 6$ , and another convolution of 64 parallel  $6 \times 3$  kernels. Both convolutional layers employ linear rectifier units. While the first convolution emphasizes certain low-level aspects of the time-frequency patches it processes (for example the contrast

between patches), the subsequent pooling layer spatially condenses both dimensions. This effectively expands the scope with regard to the input features for the second convolution. The resulting learned features are fed into a dense layer of 512 sigmoid units encoding the relevance of individual feature components of the time-frequency window and the contribution of individual convolutional filters. Finally, the network ends in a dense output layer with two sigmoid units. Additionally, the class information (Indian tāḷa/tāl class or ballroom style class, which can generally be assumed to be known) is fed through one-hot coding directly to the first dense layer. Using this class information improves results in the range of 1–2%.

During training, the beat and downbeat units are tied to the target information from the ground-truth annotations using a binary cross-entropy loss function. The targets are set to one with a tolerance window of 5 frames, equivalent to 50 milliseconds, around the exact location of the beat or downbeat. Training weights decline according to a Gaussian window around this position (‘target smearing’). Training is done by mini-batch stochastic gradient descent, using the same hyper-parameters and tweaks as in [18]. The dense layers use dropout learning, updating only 50% of the weights per training step.

### 3.3 Beat and downbeat prediction

In order to obtain beat and downbeat estimations from a trained CNN, we follow the basic peak-picking strategy described in [18] to retrieve likely boundary locations from the network output. Note that the class information is provided in the same way as in the training, which means that we assume the meter type (e.g., 7/4) known, and target the tracking of the given metrical hierarchy. The adjustable parameters for peak picking have been optimized on the validation set. Several individual network models have been trained individually from random initializations, yielding slightly different predictions. Differently than in [18] we did not ‘bag’ (that is, average) multiple models, but rather selected the model with the best results as evaluated on the validation set. Although the results directly after peak picking are inferior to bagged models by up to 3%, the Bayesian post-processing works better on non-averaged network outputs, as also tested on the validation set. The CNN output vectors that represents the beat probability will be referred to as  $P(b)$ , and the vector representing the downbeat probabilities as  $P(d)$ , respectively. The results obtained from the peak picking on these vectors will be denoted as **CNN-PP**.

## 4. METER TRACKING USING BAYESIAN NETWORKS

The Bayesian network used for meter tracking is an extension of the model presented in [11]. Within the model in [11], activations from RNN were used as observations in a Bayesian network for beat tracking in music, whereas in this paper we extend the approach to the tracking of a metrical cycle. We will shortly summarize the principle of the

algorithm presented in [11] in Section 4.1. In Section 4.2, we present the extension of the existing approach to meter tracking using activations from a CNN.

#### 4.1 Summary: A Bayesian meter tracking model

The underlying concept of the approach presented in [11] is an improvement of [8], and was first described by [19] as the bar pointer model. In [11], given a series of observations/features  $\mathbf{y}_k$ , with  $k \in \{1, \dots, K\}$ , computed from a music signal, a set of hidden variables  $\mathbf{x}_k$  is estimated. The hidden variables describe at each analysis frame  $k$  the position  $\Phi_k$  within a beat (in the case of beat tracking) or within a bar (in the case of meter tracking), and the tempo in positions per frame ( $\dot{\Phi}_k$ ). The goal is to estimate the hidden state sequence that maximizes the posterior (MAP) probability  $P(\mathbf{x}_{1:K}|\mathbf{y}_{1:K})$ . If we express the temporal dynamics as a Hidden Markov Model (HMM), the posterior is proportional to

$$P(\mathbf{x}_{1:K}|\mathbf{y}_{1:K}) \propto P(\mathbf{x}_1) \prod_{k=2}^K P(\mathbf{x}_k|\mathbf{x}_{k-1})P(\mathbf{y}_k|\mathbf{x}_k) \quad (1)$$

In (1),  $P(\mathbf{x}_1)$  is the *initial state distribution*,  $P(\mathbf{x}_k|\mathbf{x}_{k-1})$  is the *transition model*, and  $P(\mathbf{y}_k|\mathbf{x}_k)$  is the *observation model*. When discretizing the hidden variable  $\mathbf{x}_k = [\Phi_k, \dot{\Phi}_k]$ , the inference in this model can be performed using the Viterbi algorithm. In this paper, for the sake of simplicity of representation we do not apply approximate inference, as for instance in [17], but strictly follow the approach in [11].

In [11], efficiency of the inference was improved by a flexible sampling of the hidden variables. The position variable  $\Phi_k$  takes  $M(T)$  values  $1, 2, \dots, M(T)$ , with

$$M(T) = \text{round}\left(\frac{N_{beats} * 60}{T * \Delta}\right) \quad (2)$$

where  $T$  denotes the tempo in beats per minute (bpm), and  $\Delta$  the analysis frame duration in seconds. In the case of meter tracking,  $N_{beats}$  denotes the number of beats in a measure (e.g., nine beats in a 9/8), and is set to 1 in the case of beat tracking. This sampling results in one position state per analysis frame. The discretized tempo states  $\dot{\Phi}_k$  were distributed logarithmically between a minimum tempo  $T_{min}$  and a maximum tempo  $T_{max}$ .

As in [11], a uniform initial state distribution  $P(\mathbf{x}_1)$  was chosen in this paper. The transition model factorizes into two components according to

$$P(\mathbf{x}_k|\mathbf{x}_{k-1}) = P(\Phi_k|\Phi_{k-1}, \dot{\Phi}_{k-1})P(\dot{\Phi}_k|\dot{\Phi}_{k-1}) \quad (3)$$

with the two components describing the transitions of position and tempo states, respectively. The position transition model increments the value of  $\Phi_k$  deterministically by values depending on the tempo  $\dot{\Phi}_{k-1}$ , starting from a value of 1 (at the beginning of a metrical cycle) to a value of  $M(T)$ . The tempo transition model allows for tempo transitions according to an exponential distribution in exactly the same way as described in [11].

We incorporated the *GMM-BarTracker (GMM-BT)* as described in [11] as a baseline in our paper. The observation model in the *GMM-BarTracker* divides a whole note into 64 discrete bins, using the beat and downbeat annotations that are available for the data. For instance, a 5/4 meter would be divided into 80 metrical bins, and we denote this number of bins within a specific meter as  $N_{bins}$ . Spectral-flux features obtained from two frequency bands, computed as described in [12], are assigned to one of these metrical bins. Then, the parameters of a two-component Gaussian Mixture Model (GMM) are determined in exactly the same way as documented in [12], using the same training data as for the training of the CNN in Section 3.1. Furthermore, the fastest and the slowest pieces were used to determine the tempo range  $T_{min}$  to  $T_{max}$ . A constant number of 30 tempo states were used, a denser sampling did not improve tracking on any of the validation sets.

#### 4.2 Extension of the Bayesian network: CNN observations

The proposed extensions of the GMM-BT approach affect the observation model  $P(\mathbf{y}_k|\mathbf{x}_k)$ , as well as the parametrization of the state space. We will refer to this novel model as **CNN-BT**.

Regarding the observation model, we incorporate the beat and downbeat probabilities  $P(b)$  and  $P(d)$ , respectively, obtained from the CNN as described in Section 3. Network activations were incorporated in [11] on the beat level only, and in this paper our goal is to determine in how far the downbeat probabilities can help to obtain an accurate tracking not only of the beat, but the entire metrical cycle. Let us denote the metrical bins that are beat instances by  $\mathcal{B}$  (excluding the downbeat), and the downbeat position as  $\mathcal{D}$ . Then we calculate the observation model  $P(\mathbf{y}_k|\mathbf{x}_k)$  as follows

$$P(\mathbf{y}_k|\mathbf{x}_k) = \begin{cases} P_k(d)*P_k(b), & \Phi_k \in \mathcal{D}, \mathcal{D}+1; \\ P_k(b)*(1-P_k(d)) & \Phi_k \in \mathcal{B}, \mathcal{B}+1; \\ (1-P_k(b))*(1-P_k(d)) & \text{else;} \end{cases} \quad (4)$$

Including the bin that follows a beat and downbeat was found to slightly improve the performance on the evaluation data. In simple terms, the network outputs  $P(b)$  and  $P(d)$  are directly plugged into the observation model. The two separate probabilities for beats and downbeats combined according to the metrical bin. For instance, downbeats are also instances of the beat layer, and at these positions the activities are multiplied in the first row of (4). The columns of the obtained observation matrix of size  $N_{bins} \times K$  are then normalized to sum to one.

The CNN activations  $P(b)$  and  $P(d)$  are characterized by clearly accentuated peaks in the vicinity of beats and downbeats, as will be illustrated in Section 5. We take advantage of this property in order to restrict the number of possible tempo hypotheses  $\dot{\Phi}_k$  in the state space of the model. To this end, the autocorrelation function (ACF) of the beat activation function  $P(b)$  is computed, and the highest peak at tempi smaller than 500 bpm is determined. This peak serves as an initial tempo hypothesis

$T_0$ , and we define  $T_{min}=0.4*T_0$  and  $T_{max}=2.2*T_0$ , in order to include half and double tempo as potential tempo hypotheses into the search space. Then we determine the peaks of the ACF in that range, and if their number is higher than 5, we choose the 5 highest peaks only. This way we obtain  $N_{hyp}$  tempo hypotheses, covering  $T_0$ , its half and double value (in case the ACF has peaks at these values), as well as possible secondary tempo hypotheses. These peaks are then used to determine the number of position variables at these tempi according to (2). In order to allow for tempo changes around these modes, we include for a mode  $T_n$ ,  $n \in \{1, \dots, N_{hyp}\}$ , all tempi related to  $M(T_n)-3, M(T_n)-2, \dots, M(T_n)+3$ . This means that for each of the  $N_{hyp}$  tempo modes we use seven tempo samples with the maximum possible accuracy at a given analysis frame rate  $\Delta$ , resulting in a total of at most 35 tempo states (for  $N_{hyp}=5$ ). Using more modes or more tempo samples per mode did not result in higher accuracy on the validation data. While this focused tempo space has not been observed to lead to large improvements over a logarithmic tempo distribution between  $T_{min}$  and  $T_{max}$ , the more important consequence is a more efficient inference. As will be shown in Section 5, metrically simple pieces are characterized by only 2 peaks in the ACF between  $T_{min}$  and  $T_{max}$ , which leads to a reduction of the state space size by more than 50% over the GMM-BT.

## 5. SYSTEM EVALUATION

### 5.1 Evaluation measures

We use three evaluation measures in this paper [4]. For **F-measure** (0% to 100%), estimations are considered accurate if they fall within a  $\pm 70$  ms tolerance window around annotations. Its value is measured as a function of the number of true and false positives and false negatives. **AMLt** (0% to 100%) is a continuity-based method, where beats are accurate when consecutive beats fall within tempo-dependent tolerance windows around successive annotations. Beat sequences are also accurate if the beats occur on the off-beat, or are at double or half the annotated tempo. Finally, **Information Gain (InfG)** (0 bits to approximately 5.3 bits) is determined by calculating the timing errors between an annotation and all beat estimations within a one-beat length window around the annotation. Then, a beat error histogram is formed from the resulting timing error sequence. A numerical score is derived by measuring the K-L divergence between the observed error histogram and the uniform case. This method gives a measure of how much information the beats provide about the annotations.

Whereas the F-measure does not evaluate the continuity of an estimation, the AMLt and especially the InfG measure penalize random deviations from a more or less regular underlying beat pulse. Because it is not straight-forward to apply such regularity constraints on the downbeat level, downbeat evaluation is done using the F-measure only, denoting the F-measure at the downbeat and beat levels as  $F(d)$  and  $F(b)$ , respectively.

Evaluation Measure	$F(d)$	$F(b)$	AMLt	InfG
CNN-PP	54.29	75.15	60.80	1.820
GMM-BT	63.84	77.00	74.92	1.942
CNN-BT	<b>69.93</b>	80.75	<b>87.46</b>	<b>2.314</b>
CNN-BT ( $T_{ann}$ )	73.63	85.27	89.22	2.499

**Table 3:** Results on Indian music.

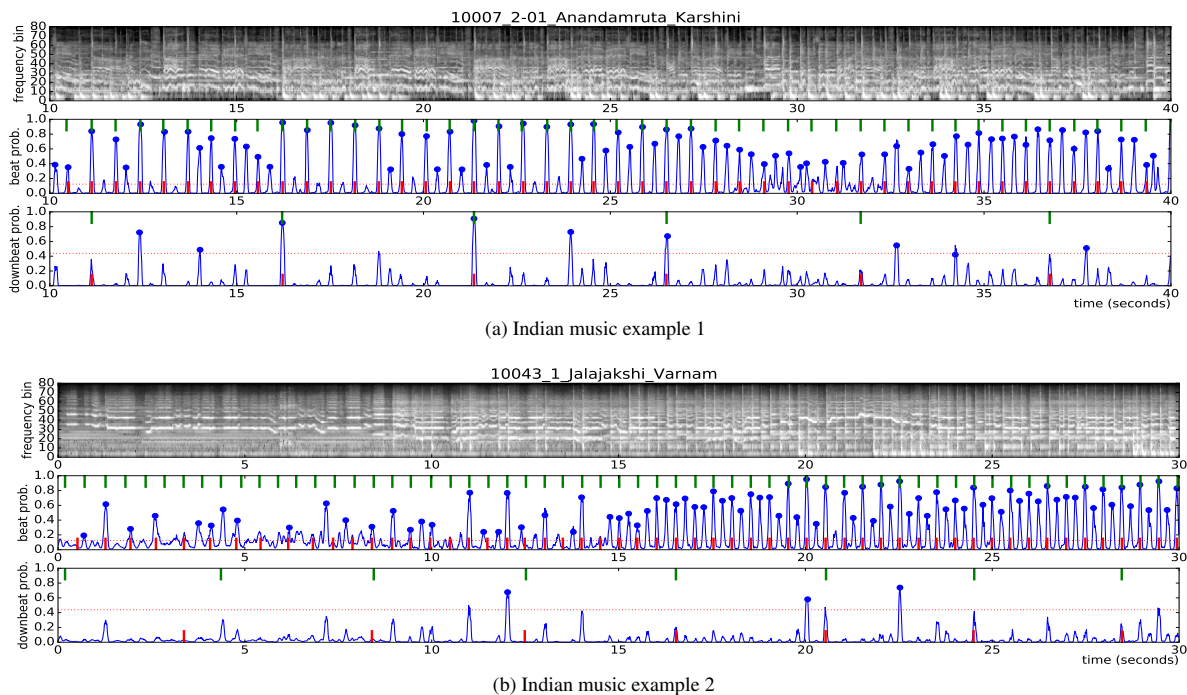
Evaluation Measure	$F(d)$	$F(b)$	AMLt	InfG
CNN-PP	79.30	93.59	88.98	3.216
GMM-BT	77.51	90.67	91.45	2.961
CNN-BT	<b>89.63</b>	93.74	<b>93.89</b>	<b>3.244</b>
CNN-BT ( $T_{ann}$ )	90.67	94.81	94.25	3.240

**Table 4:** Results on Ballroom music.

### 5.2 Results

Results are presented separately for the Indian and the Ballroom datasets in Tables 3 and 4, respectively. The first two columns represent F-scores for downbeats ( $F(d)$ ) and beats ( $F(b)$ ), followed by AMLt and InfG. We evaluated CNNs with subsequent peak-picking on the network activations (CNN-PP) as explained in Section 3, the Bayesian network from [11] using Spectral Flux in its observation model (GMM-BT), and the Bayesian network that incorporates the novel observation model obtained from CNN activations (CNN-BT). Bold numbers indicate significant improvement of CNN-BT over CNN-PP, underlining indicates significant improvement of CNN-BT over GMM-BT. Paired-sample t-tests were performed with a 5% significance level. Performing a statistical test over both corpora reveals a significant improvement by CNN-BT over CNN-PP for all measures, and for  $F(d)$  and AMLt over GMM-BT. These results demonstrate that beat and downbeat estimations obtained from a CNN can be further improved using a Bayesian model that incorporates hypotheses about metrical regularity and the dynamic development of tempo. On the other hand, employing CNN activations yields significant improvements over the Bayesian model that incorporates hand-crafted features (Spectral Flux).

Figure 2 visualizes the improvement of CNN-BT over CNN-PP by depicting the network outputs along with reference annotations, and beat and downbeat estimations from CNN-BT and CNN-PP. It is apparent that the Bayesian network finds a consistent path through the pieces that is supported by the network activations as well as by the underlying regular metrical structure. Both figures depict examples of Carnatic Adi tāla, which has a symmetric structure that caused tempo halving/doubling errors when using spectral flux features as in GMM-BT [8]. In Figure 2a, the spectrogram, especially in the first two depicted cycles, is characterized by a similar melodic progression that marks the cycle. The CNN is able to capture such regularities, leading to an improved performance. In Figure 2b, the music provides no clear metrical cues in the beginning, but the output of the CNN-BT can be seen to be nicely synchronized from the third cycle on (at about 8 s), demonstrating the advantage of the regularity imposed by the Bayesian network.



**Figure 2:** Input LLS features and network outputs for beat (upper curve) and downbeat (lower curve) predictions for two music examples. Ground-truth positions as green vertical marks on top, peak-picking thresholds as red dotted lines, picked peaks from the CNN-PP as blue circle markers, and final predictions by the Bayesian tracking (CNN-BT) as red vertical marks on the bottom.

Corpus	Ballroom	Carnatic	Hindustani
Correct tempo (%)	97.1	100	81.8
ACF-peaks	2.67	3.60	4.15

**Table 5:** Some characteristics of the focused state space in CNN-BT. The first row depicts the percentage of pieces for which the true tempo was between  $T_{min}=0.4*T_0$  to  $T_{max}=2.2*T_0$  that was selected using the autocorrelation function (ACF) of  $P(b)$ . The second row depicts the number of peaks in the ACF in the selected tempo range.

In Table 5, we depict some characteristics of the tempo states that are chosen in the CNN-BT, as described in Section 4.2. We depict the Carnatic and Hindustani musics separately in order to illustrate differences. It can be seen that the true tempo is almost always in the chosen range from  $T_{min}$  to  $T_{max}$  for Ballroom and Carnatic music, but drops to 81.8% for Hindustani music. Furthermore, the number of peaks in the ACF of  $P(b)$  is lowest for the Ballroom corpus, while the increased number for the Hindustani music indicates an increased metrical complexity for this style. Indeed, the performance values are generally lower for Hindustani musics than for Carnatic musics, with, for instance, the downbeat F-measure  $F(d)$  being 0.76 for Carnatic, and 0.64 for Hindustani musics. This is to some extent related to the extremely low tempi that occur in Hindustani music, which cause the incorrect tempo ranges for Hindustani music depicted in Table 5.

The last rows in Tables 3 and 4 depict the performance that is achieved when the correct tempo  $T_{ann}$  is given in CNN-BT. To do this evaluation, we use 30 logarithmically-spaced tempo coefficients in a range of  $\pm 20\%$  around

$T_{ann}$ , in order to allow for gradual tempo changes, excluding, however, double and half tempo. For the Ballroom corpus, only marginal improvement can be observed, with none of the changes compared to the non-informed CNN-BT case being significant. For the Indian data the improvement is larger, however, again not significantly. This illustrates that even a perfect tempo estimation cannot further improve the results. The reasons for this might be, especially for Hindustani music, the large variability within the data due to the huge tempo ranges. The CNNs are not able to track pieces at extreme slow tempi, due to their limited temporal horizon of 5 seconds – slightly shorter than the beat period in the slowest pieces. However, further increasing this horizon was found to generally deteriorate the results, due to more network weights to learn with the same, limited amount of training data.

## 6. DISCUSSION

In this paper, we have combined CNNs and Bayesian networks for the first time in the context of meter tracking. Results clearly indicate the advantage of this combination that results from the flexible signal representations obtained from CNNs with the knowledge of metrical progression incorporated into a Bayesian model. Furthermore, the clearly accentuated peaks in the CNN activations enable us to restrict the state space in the Bayesian model to certain tempi, thus reducing computational complexity depending on the metrical complexity of the musical signal. Limitations of the approach can be seen in the ability to track very long metrical structures in Hindustani music. To this end, the incorporation of RNN will be evaluated in the future.

## 7. REFERENCES

- [1] Sebastian Böck, Florian Krebs, and Gerhard Widmer. A multi-model approach to beat tracking considering heterogeneous music styles. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 602–607, Taipei, Taiwan, 2014.
- [2] Baris Bozkurt, Ruhi Ayangil, and Andre Holzapfel. Computational analysis of makam music in Turkey: Review of state-of-the-art and challenges. *Journal for New Music Research*, 43(1):3–23, 2014.
- [3] Martin Clayton. *Time in Indian Music : Rhythm , Metre and Form in North Indian Rag Performance*. Oxford University Press, 2000.
- [4] M. E. P. Davies, N. Degara, and M. D. Plumbley. Evaluation methods for musical audio beat tracking algorithms. Technical Report C4DM-TR-09-06, Queen Mary University of London, Centre for Digital Music, 2009.
- [5] S. Dixon, F. Gouyon, and G. Widmer. Towards characterisation of music via rhythmic patterns. In *Proceedings of International Conference on Music Information Retrieval*, pages 509–516, 2004.
- [6] Simon Durand, Juan Pablo Bello, Bertrand David, and Gaël Richard. Feature adapted convolutional neural networks for downbeat tracking. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing. ICASSP*, 2016.
- [7] Thomas Grill and Jan Schlüter. Music Boundary Detection Using Neural Networks on Combined Features and Two-Level Annotations. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Malaga, Spain, 2015.
- [8] Andre Holzapfel, Florian Krebs, and Ajay Srinivasamurthy. Tracking the “odd”: Meter inference in a culturally diverse music corpus. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 425–430, Taipei, Taiwan, 2014.
- [9] A. P. Klapuri, A. J. Eronen, and J. T. Astola. Analysis of the meter of acoustic musical signals. *IEEE Transactions on Audio, Speech and Language Processing*, 14(1):342–355, 2006.
- [10] Florian Krebs, Sebastian Böck, and Gerhard Widmer. Rhythmic pattern modeling for beat- and downbeat tracking in musical audio. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Curitiba, Brazil, 2013.
- [11] Florian Krebs, Sebastian Böck, and Gerhard Widmer. An efficient state-space model for joint tempo and meter tracking. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Malaga, Spain, 2015.
- [12] Florian Krebs, Andre Holzapfel, Ali Taylan Cemgil, and Gerhard Widmer. Inferring metrical structure in music using particle filters. *IEEE Transactions on Audio, Speech and Language Processing*, 23(5):817–827, 2015.
- [13] Donna Lee Kwon. *Music in Korea : experiencing music, expressing culture*. Oxford University Press, 2011.
- [14] Meinard Müller, Daniel P. W. Ellis, Anssi Klapuri, and Gaël Richard. Signal processing for music analysis. *J. Sel. Topics Signal Processing*, 5(6):1088–1110, 2011.
- [15] Geoffroy Peeters and Helene Papadopoulos. Simultaneous beat and downbeat-tracking using a probabilistic framework: Theory and large-scale evaluation. *IEEE Transactions on Audio, Speech and Language Processing*, 19(6):1754–1769, 2011.
- [16] Jan Schlüter, Karen Ullrich, and Thomas Grill. Structural segmentation with convolutional neural networks mirex submission. In *Tenth running of the Music Information Retrieval Evaluation eXchange (MIREX 2014)*, 2014.
- [17] Ajay Srinivasamurthy, Andre Holzapfel, Ali Taylan Cemgil, and Xavier Serra. Particle filters for efficient meter tracking with dynamic bayesian networks. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Malaga, Spain, 2015.
- [18] Karen Ullrich, Jan Schlüter, and Thomas Grill. Boundary Detection in Music Structure Analysis using Convolutional Neural Networks. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, Taipei, Taiwan, 2014.
- [19] N. Whiteley, A. T. Cemgil, and S. J. Godsill. Bayesian modelling of temporal structure in musical audio. In *Proceedings of International Conference on Music Information Retrieval*, Victoria, Canada, 2006.