Characterization, Detection, and Synchronization of Audio-Video Events in *Bharatanatyam Adavu*'s

Tanwi Mallick, Partha Pratim Das, Arun Kumar Majumdar

Department of Computer Science and Engineering, Indian Institute of Technology, Kharagpur 721302, India
tanwimallick@gmail.com, ppd@cse.iitkgp.ernet.in, akmj@cse.iitkgp.ernet.in

Abstract. *Bharatanatyam* is the most popular form of *Indian Classical Dance*. Its *Adavu*'s are basic choreographic units of a dance sequence. An *Adavu* is accompanied by percussion and vocal music and follows a specific rhythmic pattern (*Sollukattu*). In this paper we first characterize the audio, video, and sync events of *Adavu*'s to succinctly represent the *Adavu*'s. Then we present simple yet effective algorithms to detect audio and video events and measure their synchronization. The audio, video, and sync event detection achieve 94%, 84%, and 72% accuracy respectively. A comparison of our audio event detection against a well-known method by Ellis shows significant improvement. We also create an annotated repository of *Sollukattu*'s and *Adavu*'s for research. There are several applications of the characterization and beat detection including music / music video segmentation, synchronization of the postures with the beats, automatic tagging of rhythm metadata etc. Characterization of events or repository of *Bharatanatyam Adavu*'s has not been attempted before.

Keywords: Characterization of Dance Videos, Onset Detection, Beat Detection, Key Posture Detection, Audio-Visual Synchronization

1 Introduction

Bharatanatyam is a very popular form of Indian Classical Dance. *Adavu*'s are basic choreographic units of a dance sequence in *Bharatanatyam*. In an *Adavu* choreographic movements are accompanied by percussion instruments (*Tatta Palahai* (wooden stick) – *Tatta Kozhi* (wooden block), *Mridangam*, or *Tabla*) and rhythmic vocal sound (utterances). Optionally, vocal music, various woodwind (*Nagaswaram*, Flute) or string (Violin, or *Veena*) instruments also accompany *Adavu*'s. Hence, a performance of the an *Adavu* is recorded as a multimedia stream comprising audio and video streams (based on the sensor there may be other streams as well). This is, therefore, a combination of video events that are either postures or movements synchronized with audio events that are rhythmic pattern of beats or Taals. The rhythmic patterns (meter) used for *Adavu*'s are called *Sollukattu*'s. Every *Adavu* is performed in sync with a *Sollukattu*. There are[1] 50 *Adavu*'s each performed with one of the 23 *Sollukattu*'s.

In this paper we first present an in-depth characterization of *Bharatanatyam* performances for representation and processing of its audio as well as video streams. We characterize the *Sollukattu*'s (audio stream) in terms of audio-events comprising beats (and half beats), inter-beat silence, and their periodic structure. *Adavu*'s (video stream) are characterized in terms of Key Postures and their transitions, and movements together defining video-events. Finally, we characterize the synchronization between audio and video events and the associated issues in synchronization to understand the multimedia form of *Bharatanatyam* dance. These characterizations are severally used later to formulate algorithms, design tests and validations and create the basis for solving various choreographic problems.

Computationally we first present an algorithm to detect the beats of *Sollukattu*'s. These provide major clues to the audio events. Several work on beat detection, tempo estimation, and beat tracking have been reported in [2], [3], and [4]. These algorithms rely on a common scheme where the system extracts the onset locations from a time-frequency or sub-band analysis of the signal, traditionally using a filter bank or the discrete Fourier transform. Then, a periodicity estimation algorithm finds the rate at which these events occur.

Problem of estimating the meter of a musical piece has been addressed in [8], [7], [13], and [6]. The work by [13], [12] and [6] are based on *Indian Hindusthani & Carnatic Music*. Gulati et. al. [6] extended the two stage comb filter-based approach (originally proposed for double/ triple meter estimation) to septuple meter (such as 7/8 time-signature) and evaluated its performance on a sizable Indian music database. In [12], Sridhar et. al. propose a new algorithm to segment the instrumental and the vocal signals. The frequency components of the signal are determined on the voice signal and then these are mapped onto the *swara* sequence. Srinivasamurthy et. al. [13] present an algorithm that uses a beat similarity matrix and inter onset interval histogram to automatically extract the sub-beat structure and the long-term periodicity of a musical piece. On a manually annotated *Carnatic* music data set the recognition accuracy of the algorithm is shown to be 79.3%.

Here, we develop a simple yet effective onset based [4] algorithm to detect the beats for the polyphonic music signal of *Bharatanatyam Adavu*. The algorithm achieves over 94% accuracy for beat detection for the 23 *Sollukattu*'s for a set of annotated audio streams.

Next we analyze the video for the extent of motion between its sequences of consecutive frames to detect Key Frames (containing Key Postures), Transition Frames and Movements. These provide significant clues to video events. We achieve nearly 84% accuracy for key posture detection for the 50 *Adavu*'s for a set of annotated video streams.

---

[1] Depending on the school of *Bharatanatyam*, the exact set of *Adavu*'s and *Sollukattu*'s may vary.

Finally, to explore the synchronization aspects, we correlate the audio events from *Sollukattu*'s with the video events from *Adavu*'s. There has been variety of work in this area including – audio based video event detection [11], dance synthesis based on visual analysis of human motion and audio analysis of music tempo [9], detection of dance motion structure using motion capture and musical information [10], and audio and video tempo analysis for dance detection [5]. However, there has been no attempt to analyze synchronization in Indian Classical Dance forms. Here, we work on synchronization between beats (audio events) and key frames (video events) for the *Adavu*'s and achieve 72% accuracy of sync.

There has been no systematic research on multimedia streams of *Bharatanatyam Adavu*'s. Hence, there is no comprehensive and annotated data set for it. So we also create an annotated repository of *Sollukattu*'s and *Adavu*'s for research. The data set is created using Kinect XBox (Kinect 1.0). Hence it has depth and skeleton data streams synchronized with RGB stream that can be further used for analysis of specific postures and movements. The data set is captured for all 23 *Sollukattu*'s performed independently by 4 trained music accomplices of dancers. All 50 *Adavu*'s are also recorded using 7 different professionally trained dancers. A part of the data has been annotated by *Bharatanatyam* experts. These have been used for validation of our algorithms and comparison with others in some cases. A selective subset of the data has been published[2] for public use.

There are several applications of the characterization and beat detection including music / music video segmentation, synchronization of the postures with the beats, automatic tagging of rhythm metadata etc. Characterization, beat detection, synchronization, segmentation or repository of *Bharatanatyam Adavu*'s has not been attempted before.

The paper makes three major contributions - characterization of audio and video of *Adavu*'s, algorithms for detection of audio events, video events and their synchronization, and creation of an annotated repository of *Bharatanatyam* data.

The paper is organized as follows. We characterize the multi-modal structure of *Bharatanatyam Adavu*'s in terms of audio, video and sync events in Section 2. Audio event (beat) detection is presented in Section 3 where we first outline the pre-processing, followed by onset detection and subsequent pruning, and beat detection. Video event (motion) detection is presented in Section 4. Estimation of sync is then discussed in Section 5. We conclude in Section 6.

2 Characterization of *Bharatanatyam Adavu*'s

A *Bharatanatyam Adavu* consists of:

1. **Audio Stream**: *Sollukattu* or rhythmic music as generated by percussion instrument and vocal sound (utterances).
2. **Video Stream**: Stream of frames each capturing the combination of (a) Position of the legs (*Sthanakam*), (b) Posture of standing (*Mandalam*), (c)

[2] Data Repository: http://cse.iitkgp.ac.in/resgrp/hci/

Walking Movement (*Chari*), and (d) Hand Gestures (*Nritta Hastas*) as assumed by the dancer.
3. **Synchronization**: Position, Posture, Movement and Gesture of an *Adavu* are performed in synchronization among themselves and in synchronization with the rhythm of the music.

To characterize the above and represent an *Adavu* in a succinct manner, we define a set of events.

2.1 Events of *Adavu*'s

An *Event* denotes the occurrence of an activity (called *Causal Activity*) in the audio or the video stream of an *Adavu*. Further, synchronization (sync) events are defined between multiple events based on temporal constraints. Sync events may be defined jointly between audio and video streams. An event is described by:

1. *Category*: The nature of the event based on its origin (source) is called Category. It can be *audio*, *video* or *sync*.
2. *Type*: Type relates to the causal activity of an event in a given category. Event types are listed in Table 1 with brief description of respective causal activities.
3. *Time-stamp / range*: The time of occurrence of the causal activity of the event. This is elapsed time from the beginning of the stream and is marked by a function $\tau(.)$. Often a causal activity may spread over an interval $[\tau_s, \tau_e]$ which will be associated with the event. Time-stamp and time range are interchangeably denoted by the τ function of the event.
For video events, we use range of video frame numbers $[\eta_s, \eta_e]$ as the temporal interval. The Kinect video has a fixed rate of 30 fps. Hence, for any event we interchangeably use τ or η as is appropriate in a context.
4. *Label*: One or more optional labels may be attached to an event annotating details for the causal activity.
5. *ID*: Every instance of an event in a stream is distinguishable. These are sequentially numbered (within a specific type of an event) in the temporal order of their occurrence.

2.2 Characterization of Audio

The musical *meter*[3] of an *Adavu* is called a *Sollukattu*. Traditionally, a *Tatta Palahai* (wooden stick) is periodically struck on a *Tatta Kozhi* (wooden block) in the rhythmic pattern of *Adi* or *Rupak Taal*'s[4] to produce the periodic beats (or α^{fb} events). Usually beats repeat in a *bar*[5] of $\lambda = 6$ or 8. The *tempo* of a

[3] The *meter* of music is its rhythmic structure.
[4] *Taal* is the Indian system for organizing and playing metrical music.
[5] A *bar* (or *measure*) is a segment of time corresponding to a specific λ number of beats. *Sollukattu*'s also use longer bars (12, 16, 24, or 32).

Table 1. List of Events in *Bharatanatyam Adavu*'s

Event Category	Event Type	Event Description	Event Label
Audio	α^{fb}	Full beat with *bol*	*bol*[1], downbeat[2], upbeat[3]
Audio	α^{hb}	Half beat with *bol*	*bol*
Audio	α^{fn}	Full beat having no *bol*	upbeat
Audio	α^{hn}	Half beat having no *bol*	
Audio	α^{qn}	Quarter[4] beat having no *bol*	
Audio	α^{sl}	Silence – No beat or *bol*	upbeat
Audio	α^{f}	$\alpha^{fb} \mid \alpha^{fn}$	*bol*[1], downbeat[2], upbeat[3]
Audio	α^{h}	$\alpha^{hb} \mid \alpha^{hn}$	*bol*
Audio	α	$\alpha^{f} \mid \alpha^{h} \mid \alpha^{qn} \mid \alpha^{sl}$	
Video	ν^{nm}	No motion[5]	Range of Frames[6], Key Posture[7]
Video	ν^{tr}	Transition Motion[8]	Range of Frames
Video	ν^{tj}	Trajectory Motion[9]	Range of Frames, Trajectory
Video	ν^{t}	$\nu^{tr} \mid \nu^{tj}$	Range of Frames, Trajectory
Video	ν	$\nu^{t} \mid \nu^{nm}$	
Sync	ψ^{fb}	No motion @ Full beat[10]	Key Posture
Sync	ψ^{hb}	No motion @ Half beat	Key Posture
Sync	ψ	$\psi^{fb} \mid \psi^{hb}$	

1: Vocalized *bol*'s accompany some beats
2: The first beat of a bar
3: The last beat in the previous bar which immediately precedes, and hence anticipates, the downbeat
4: *Sollukattu*'s do not use quarter beats to define a meter. However, often the beat player would produce one that needs to be ignored
5: Frames over which the dancer does not move (assumes a Key Posture)
6: Sequence of consecutive frames over which the events spreads
7: A Key Posture is a well-defined and stationery posture
8: Transitory motion to change from one Key Posture to the next. This has no well-defined trajectory of movement for limbs
9: Motion that follows a well-defined trajectory of movement for limbs
10: α^{fb} and ν^{nm} in sync. That is, $\tau(\alpha^{fb}) \cap \tau(\nu^{nm}) \neq \phi$

meter is measured by beats per minute (*bpm*). We use Period $T = (60/bpm)$ or the time interval between two consecutive beats in secs as the temporal measure for a meter.

Consider two consecutive beats α_i^{fb} and α_{i+1}^{fb} in a bar of λ, where i denotes the i^{th} $(1 \leq i < \lambda)$ period. The time-stamps of the respective events are then related as $\tau(\alpha_{i+1}^{fb}) - \tau(\alpha_i^{fb}) \approx T$. Further the bar repeats after an equal time interval of T. That is, $\tau(\alpha_{\lambda*i+1}^{fb}) - \tau(\alpha_{\lambda*i}^{fb}) \approx T$, $i \geq 1$. We refer to such beats as *full beats* and hence the superscript *fb* in α^{fb} events. The first beat α_1^{fb} (last beat α_λ^{fb}) of a bar is referred to as a *downbeat* (*upbeat*). We mark these on the events as labels.

In many *Sollukattu*'s beating is also performed at the middle of a period. These are called *half beats* and produce the α_i^{hb} events in the i^{th} period. Naturally, $\tau(\alpha_i^{hb}) - \tau(\alpha_i^{fb}) \approx \tau(\alpha_{i+1}^{fb}) - \tau(\alpha_i^{hb}) \approx T/2$.

A *Sollukattu* uses one of the 3 different speeds or *Tempo* (*Laya*) – *Vilambit Laya* (Slow), *Madhya Laya* (Medium), and *Drut Laya* (High). The *Period* (T) depends on the *Tempo* (shorter for faster tempo) and remains more or less uniform across *Sollukattu*'s.

Often in a *Sollukattu* an accomplice of the dancer also speaks out a distinct vocalization of rhythm with words like *tat*, *tei*, *ta* etc., called *Bol*'s. These are done in sync with a full beat or a half beat. We represent *bol*'s as labels of the respective α^{fb} or α^{hb} events. A *bol* is optional for an event.

There are 23 *Sollukattu*'s. We illustrate a few here to understand various meters. All *Sollukattu*'s are shown in *Vilambit Laya*.

1. *Kuditta Mettu* ($T \approx 1.2$ secs, $\lambda = 8$): We show two meters of it in Table 2 and Figure 1 (a). Note that it has only α^{fb} events.
2. *Tatta_C Sollukattu* ($T \approx 1.6$ secs, $\lambda = 8$): It has α^{fb} as well as α^{hb} events (Table 3 and Figure 1 (b)).
3. *Kuditta Nattal_A* & *Tatta_E* ($T \approx 1.0$ secs, $\lambda = 8$): In addition to α^{fb}, α^{fn} and α^{hn} events are also found (Table 4) where there is only beating and no *bol*.
4. *Joining_B* ($T \approx 1.5$ secs, $\lambda = 8$): As such it uses only α^{fb}'s (Table 4). But the 4^{th} and 8^{th} beats are silent (α^{sl}) with neither any *bol* nor any beating. So the upbeat in this case is guessed from T.

2.3 Characterization of Video

While performing an *Adavu* the dancer closely follows the beats of the accompanying[6] *Sollukattu* and synchronizes her movements with the beats. At a beat, the dancer assumes a *Key Posture*[7] and holds it for a little while before quickly changing to the next *Key Posture* at the next beat. Consequently, while the dancer holds the key posture, she stays almost stationary and there is no or very slow motion in the video. This leads to ν^{nm} (no-motion) events. Further, while the dancer changes to the next key posture, we observe the ν^{tr} (transition) or ν^{tj} (trajectory) motion events. Since a frame is an atomic observable unit in a video, we can classify the frames of the video of an *Adavu* into 2 classes:

[6] Every *Adavu* is performed with a specific *Sollukattu*. In this paper, we use 50 *Adavu*'s each performed with one of 23 *Sollukattu*'s.

[7] A *Key Posture* is defined in terms of Position of the legs (*Sthanakam*) and Posture of standing (*Mandalam*). Some are laterally symmetric ((c)–(h) in Figure 2), while rest have *left* and *right* sided variants ((a)–(b)).

Table 2. Pattern of *Kuditta Mettu Sollukattu* (Figure 1 (a)) annotated with time-stamps τ_i (start-time of the full beat event α^{fb}). $Gap_i = \tau_i - \tau_{i-1}$ is computed from consecutive time-stamps and provides the distribution for tempo period T

Event	**Time** (τ_i)	**Gap** ($\tau_i - \tau_{i-1}$)	**Event**	**Time** (τ_i)	**Gap** ($\tau_i - \tau_{i-1}$)
α_1^{fb}(tei)	2.681		α_9^{fb}(tei)	12.271	1.207
α_2^{fb}(hat)	3.912	1.231	α_{10}^{fb}(hat)	13.386	1.115
α_3^{fb}(tei)	5.108	1.196	α_{11}^{fb}(tei)	14.512	1.126
α_4^{fb}(hi)	6.269	1.161	α_{12}^{fb}(hi)	15.603	1.091
α_5^{fb}(tei)	7.523	1.254	α_{13}^{fb}(tei)	16.764	1.161
α_6^{fb}(hat)	8.742	1.219	α_{14}^{fb}(hat)	17.902	1.138
α_7^{fb}(tei)	9.891	1.149	α_{15}^{fb}(tei)	19.028	1.126
α_8^{fb}(hi)	11.064	1.173	α_{16}^{fb}(hi)	20.178	1.150

Table 3. Pattern of *Tatta_C Sollukattu* (Figure 1 (b)) annotated with time-stamps τ_i (start-time of the full-beat event α^{fb}). $Gap_i = \tau_i - \tau_{i-1}$ is computed from consecutive time-stamps and provides the distribution for tempo period T. Half-beat offsets happen roughly at $T/2$.

Event	**Time** (τ_i)	**Gap** ($\tau_i - \tau_{i-1}$)	**1/2–Beat Offset**	**Event**	**Time** (τ_i)	**Gap** ($\tau_i - \tau_{i-1}$)	**1/2–Beat Offset**
α_1^{fb}(tei)	6.571			α_5^{fb}(tei)	13.003	1.64	
α_1^{hb}(ya)	7.395		0.82	α_5^{hb}(ya)	13.815		0.81
α_2^{fb}(tei)	8.185	1.61		α_6^{fb}(tei)	14.628	1.63	
α_2^{hb}(ya)	8.962		0.78	α_6^{hb}(ya)	15.441		0.81
α_3^{fb}(tei)	9.752	1.57		α_7^{fb}(tei)	16.184	1.56	
α_3^{hb}(ya)	10.565		0.81	α_7^{hb}(ya)	17.031		0.85
α_4^{fb}(tei)	11.366	1.61		α_8^{fb}(tei)	17.809	1.63	

1. ***K-frame*'s or Key Frames**: These frames contain key postures where the dancer *holds* the Posture. Evidently, a ν^{nm} has the sequence of *K-frames* as labels. All *K-frames* of an ν^{nm} contain the same key posture.
2. ***T-frame*'s of Transition Frame**: These are transition frames between two *K-frames* while the dancer is rapidly changing posture to assume the next key posture from the previous one. A ν^{tr} or ν^{tj} event has a sequence of *T-frames* as labels.

   For an *Aadvu* the transition can either be performed according to a well-defined trajectory[8] for the hands and legs or may just be undefined and arbitrary. Former is defined as ν^{tj} events and the latter is marked as ν^{tr}

[8] In *Bharatanatyam*, these could be various forms of *Nritta (rhythmical and repetitive elements)* like *Chari*, *Karana*, *Angahara* or *Mandala*.

Table 4. Variations in the patterns of *Sollukattu*'s with *Adavu*'s

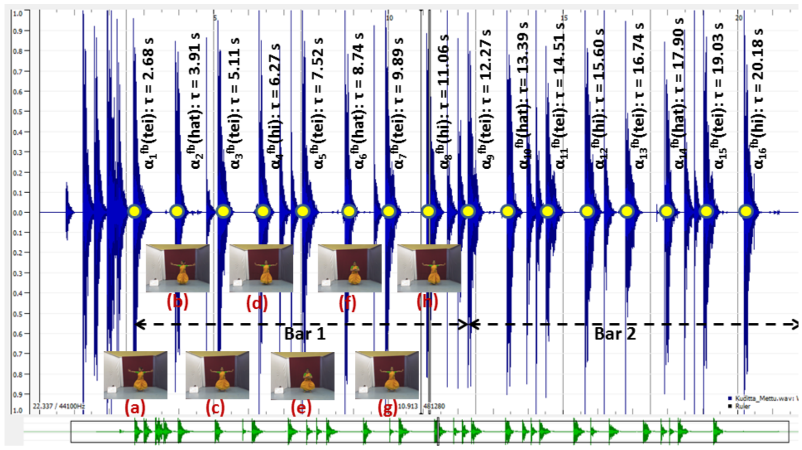
Sollukattu	**Description of Bol / Adavus**
Kuditta Mettu	α_1^{fb}(tei) α_2^{fb}(hat) α_3^{fb}(tei) α_4^{fb}(hi) α_5^{fb}(tei) α_6^{fb}(hat) α_7^{fb}(tei) α_8^{fb}(hi)
	Adavu: Kuditta_Mettu 1, 2, 3, 4
Kuditta Nattal A	α_1^{fb}(tat) α_2^{fb}(tei) α_2^{hn} α_3^{fb}(tam) α_4^{fn} α_4^{hn} α_5^{fb}(dhit) α_6^{fb}(tei) α_6^{hn} α_7^{fb}(tam) α_8^{fn} α_8^{hn}
	Adavu: Kuditta_Nattal 1, 2, 3, 6
Tatta E	α_1^{fb}(tei) α_2^{fb}(tei) α_3^{fb}(tam) α_4^{fn} α_4^{hn} α_5^{fb}(tei) α_6^{fb}(tei) α_7^{fb}(tam) α_8^{fn} α_8^{hn}
	Adavu: Tatta 6
Joining B	α_1^{fb}(dhit) α_2^{fb}(dhit) α_3^{fb}(tei) α_4^{sl} α_5^{fb}(dhit) α_6^{fb}(dhit) α_7^{fb}(tei) α_8^{sl}
	Adavu: Joining 2

event. In this paper we do not deal with trajectory-based motion and hence do not distinguish between ν^{tj} and ν^{tr} events.

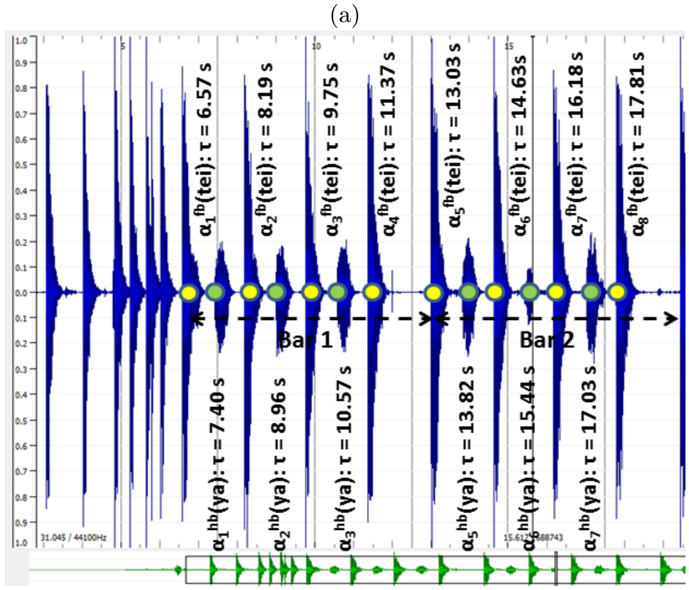
In Figure 2 we show the key postures of *Kuditta Mettu Adavu* at every beat of the first bar of *Kuditta Mettu Sollukattu*. The corresponding video and audio events are marked in Table 5 with *K-/T-Frames*. These are also marked on the *Sollukattu* in Figure 1(a). Note that only the right-sided half of the postures are shown.

Table 5. Patterns of *Kuditta Mettu Adavu* (Figure 2)

Events	***K-/T-Frames***		**Events**	***K-/T-Frames***	
	Range	**# of**		**Range**	**# of**
ν_1^{nm} [α_1^{fb}(tei)]	70–99	30	ν_9^{nm} [α_9^{fb}(tei)]	359–386	28
ν_1^{tr}	100–103	4	ν_9^{tr}	387–390	4
ν_2^{nm} [α_2^{fb}(hat)]	104–124	21	ν_{10}^{nm} [α_{10}^{fb}(hat)]	391–410	20
ν_2^{tr}	125–145	21	ν_{10}^{tr}	411–429	19
ν_3^{nm} [α_3^{fb}(tei)]	146–172	27	ν_{11}^{nm} [α_{11}^{fb}(tei)]	430–451	22
ν_3^{tr}	173–176	4	ν_{11}^{tr}	452–455	4
ν_4^{nm} [α_4^{fb}(hi)]	177–191	15	ν_{12}^{nm} [α_{12}^{fb}(hi)]	456–470	15
ν_4^{tr}	192–214	23	ν_{12}^{tr}	471–492	22
ν_5^{nm} [α_5^{fb}(tei)]	215–245	31	ν_{13}^{nm} [α_{13}^{fb}(tei)]	493–521	29
ν_5^{tr}	246–249	4	ν_{13}^{tr}	522–525	4
ν_6^{nm} [α_6^{fb}(hat)]	250–262	13	ν_{14}^{nm} [α_{14}^{fb}(hat)]	526–542	17
ν_6^{tr}	263–287	25	ν_{14}^{tr}	543–564	22
ν_7^{nm} [α_7^{fb}(tei)]	288–314	27	ν_{15}^{nm} [α_{15}^{fb}(tei)]	565–587	23
ν_7^{tr}	315–317	3	ν_{15}^{tr}	588–590	3
ν_8^{nm} [α_8^{fb}(hi)]	318–345	28	ν_{16}^{nm} [α_{16}^{fb}(hi)]	591–620	30
ν_8^{tr}	346–358	13	ν_{16}^{tr}	621–	–

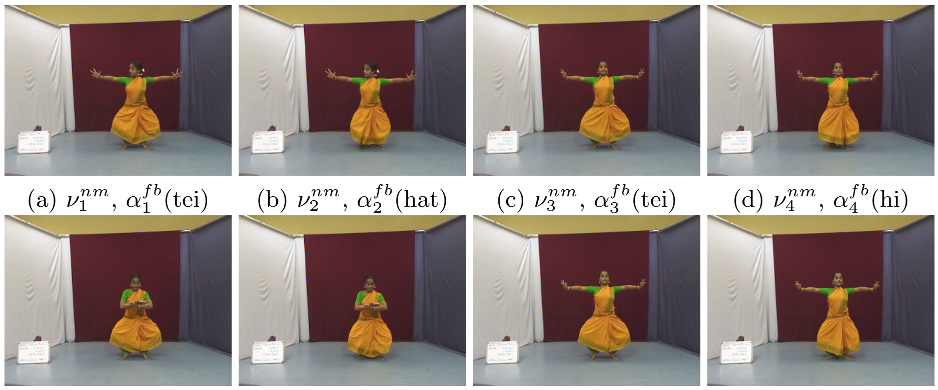


(a)



(b)

Fig. 1. Marking of beats and annotations of *bol*'s for 2 bars and $\lambda = 8$. Full beat (α^{fb}) and half beat (α^{hb}) event positions are highlighted and corresponding *bol*'s and time-stamps are shown (Tables 2 & 3). Note that several α^{hn} and α^{qn} events are visible in the signals. These are rather incidental and not intended in the *Sollukattu*. Also, the beatings before the downbeat (α_1^{fb}) are ignored. (a) *Kuditta Mettu Sollukattu* ($T = 1.16$ sec.). Right-sided *Key Postures* (Figure 2) are also shown for the first 8 beats. Left-sided *Key Postures* are performed for the next 8 beats. (b) *Tatta_C Sollukattu* ($T = 1.56$ sec.).



(a) ν_1^{nm}, α_1^{fb}(tei) (b) ν_2^{nm}, α_2^{fb}(hat) (c) ν_3^{nm}, α_3^{fb}(tei) (d) ν_4^{nm}, α_4^{fb}(hi)

(e) ν_5^{nm}, α_5^{fb}(tei) (f) ν_6^{nm}, α_6^{fb}(hat) (g) ν_7^{nm}, α_7^{fb}(tei) (h) ν_8^{nm}, α_8^{fb}(hi)

Fig. 2. Right-sided Key Postures of *Kuditta Mettu Adavu* (Variant = 2, *Sollukattu* = *Kuditta Mettu*) with *Bol*'s for Bar 1. From a *tei* to the next *hat* or *hi* the dancer sharply lowers her raised feet. Further, 8 left-sided Key Postures are performed for the next 8 beats in Bar 2.

2.4 Characterization of Synchronization

A *Bharatanatyam* dancer intends to perform the key postures of an *Adavu* in synchronization with the beats. Hence audio events like α^{fb} and corresponding video events like ν^{nm} should be in sync. Every *Adavu* has a well-defined set of rules that specifies this synchronization based on its associated *Sollukattu*. For example, in Figure 2, we show how different key postures should be assumed in the *Kuditta Mettu Avadu* at every beat of the *Kuditta Mettu Sollukattu*. That is, how the α^{fb}'s of a bar in the audio should sync with the ν^{nm}'s of the video. Other *Adavu*'s require several other forms of synchronization between the audio-video events including sync between beats and trajectory-based body movements ν^{tj}.

We assert a sync event ψ^{fb} if a key posture (ν^{nm}) sync with a corresponding (full) beat (α^{fb}). In simple terms, a ψ^{fb} occurs if the time intervals of α^{fb} and ν^{nm} events overlap. That is, $\tau(\psi^{fb}) = \tau(\alpha^{fb}) \cap \tau(\nu^{nm}) \neq \phi$. Similar sync events may be defined between other audio and video events according to the rules of *Adavu*'s.

Perfect synchronization is always intended and desirable for a performance. However, we often observe the lack of it due to various reasons. The beating instrument, vocal *bol*'s, and body postures each has a different latency. If a posture is assumed *after hearing* the beat, ν^{nm} will lag α^{fb}. If the dancer assumes the posture in *anticipation*, ν^{nm} may lead α^{fb}. Lack of sync may also arise due to imperfect performance of the dancer, the beater, the vocalist, or a combination of them. Hence, analysis and estimation of sync is critical for processing *Adavu*.

While sync between the audio and video streams is fundamental to the choreography, there are a variety of other synchronization issues that need to be explored. These include sync between beats of beating (instrumental) and (vocalized) *bol*'s, uniformity of time gap between consecutive beats, sync between different body limbs while changing from one key posture to the next, and so on.

Based on the characterizations, we next present algorithms for detection of select audio, video and sync events. In the rest of the paper, we focus only on α^{fb}, ν^{nm}, ν^{tr} and ψ^{fb} events.

3 Audio Event Detection

We detect the beats in *Sollukattu*'s in four steps as follows:

3.1 Pre-processing of the Audio Signal *Sollukattu*

A *Sollukattu* is a mixture of two sources of sound – percussion and vocal – that are synchronized by generation. It has dominant frequencies and is periodic. But it is cluttered with a lot of harmonics. So to eliminate the harmonics and noise to estimate the periodicity, we analyze it in frequency domain.

Considering N samples in the signal $S(t)$, we compute its FFT as $S^*(f)$. The frequency components in $S^*(f)$ ranges from 0 to 8 KHz with up to 800Hz contributing to vocal sound (*Bol*'s) and 1kHz to 2.6kHz to percussion sound (beating stick). Rest are harmonics.

Hence, we filter $S^*(f)$ restricting between 1Hz to 2.6kHz to eliminate the vocal sound and the harmonics and get $S^*_{filt}(f)$. Inverse FFT of $S^*_{filt}(f)$ gives $S_{filt}(t)$. Usually, the beats have high amplitude. So we discard the low amplitude components in $S_{filt}(t)$ by a threshold $Th = 0.5$ to get $S_{Th}(t)$. This is used for onset detection.

3.2 Detection of Onsets

From $S_{Th}(t)$ we compute the *Onset Strength Envelope* using [4]. $S_{Th}(t)$ is resampled at 8kHz, and STFT[9] (spectrogram) is calculated using 32ms windows and 4ms advance between frames. It is first mapped to *40 Mel bands* via a weighted sum of the spectrogram values and then the Mel spectrogram is converted to dB. The first order difference along time is calculated in each band. Negative values are set to zero (half wave rectification) and, positive differences are summed up across all frequency bands. Finally, the signal is passed through a high-pass filter with a cut-off around 0.4Hz to make it locally zero-mean, and then is smoothed by convolving with a Gaussian envelope of about 20ms width. The output is the *OSE* as a function of time that responds to proportional increase in energy summed across approximately auditory frequency bands. The algorithm also outputs the onset time in the audio stream.

[9] Short-Time Fourier Transform

3.3 Detection of Local Maxima

Naturally, every beat has an onset in the OSE, but every onset in OSE is not necessarily a beat. An onset is associated with a beat only if it is a local maxima in the OSE. To model the locality we use a window of time interval T_w, slide it over the OSE and compute the set of local maxima L_{max} at every time position in OSE. This is given in Algorithm 1. L_{max} may have more than one local maxima in a window. So in Algorithm 2 we prune the set of onsets in L_{max} to ensure that only one onset can be present in a window T_w. Pruned L_{max} contains the candidates for detected beats.

Algorithm 1 : Local Maxima Detection

Inputs:
O_t = Vector of detected onset times, $nOnset = length(O_t)$;
Val_t = Strength of onsets in O_t;
T_w = Window of time interval for local maxima, a threshold parameter;
Output:
L_{max} = Vector containing the indices of the locally maximal onsets
for $i = 1 : nOnset$ **do**
    $L_{max}(i) = 0$;
end for
for $i = 1 : nOnset$ **do**
    $max = i$;
    for do $j = i + 1 : nOnset$
        if $O_t(i) - O_t(j) < T_w$ **then**
            if $Val_t(j) > Val_t(max)$ **then**
                $max = j$;
            end if
        else
            break;
        end if
    end for
    $L_{max}(max) = 1$;
end for

3.4 Beat Detection

Using L_{max} and the periodicity of the *Sollukattu*'s we detect and mark the beats in Algorithm 3. The first candidate beat is detected as the downbeat[10]. For every detected beat $beat_d$, we search for the next beat from L_{max} that lie within $period_{low}$ and $period_{high}$ from $beat_d$, where $period_{low}$ and $period_{high}$ are global bounds on the tempo period of the *Sollukattu*'s at given speed (*laya*) and are considered invariant. We also use a threshold period $period_{th}$ which is slightly

[10] The first beat of the *Sollukattu*.

Algorithm 2 : Prunning of Local Maxima

Inputs: O_t, Val_t, T_w, L_{max} = Vector containing the indices of the locally maximal onsets
Output:
L_{max} = Vector containing the pruned indices of the locally maximal onsets
for $i = 1 : length(L_{max}) - 1$ **do**
  if $L_{max}(i) == 1$ **then**
    for $j = i + 1 : length(L_{max})$ **do**
      if $L_{max}(j) == 1$ **then**
        if $O_t(i) - O_t(j) < T_w$ **then**
          if $Val_t(i) > Val_t(j)$ **then**
            $L_{max}(j) = 0;$
          else
            $L_{max}(i) = 0;$
          end if
        end if
      end if
    end for
  end if
end for

Algorithm 3 : Beat Detection

Inputs:
L_{max} = Vector containing the pruned indices of the locally maximal onsets
$period_{max}$ = Maximum tempo period for any *Sollukattu*
$period_{min}$ = Minimum tempo period for any *Sollukattu*
$period_{th}$ = Threshold tempo period, $period_{th} > period_{max}$. Typically $period_{th} = 2$.
Output:
$Beats$ = Vector containing the indices of the detected beats
$Beats(1) = L_{max}(1);$
$i = 1;$
for $ind = 2 : length(L_{max})$ **do**
  if $L_{max}(ind) - Beats(i) > period_{min}$ **then**
    if $L_{max}(ind) - Beats(i) < period_{max}$ **then**
      $i = i + 1;$
      $Beats(i) = L_{max}(ind);$
    else $L_{max}(ind) - Beats(i) > period_{th}$
      $i = i + 1;$
      $Beats(i) = L_{max}(ind);$
    end if
  end if
end for

more than $period_{high}$. If no beat is found in L_{max} within $period_{high}$ of $beat_d$ then the next beat in L_{max} that is away by $period_{th}$ or more is detected. This is done to avoid missing a beat.

We illustrate the working of the algorithm in Table 6 for *Kuditta Mettu* by striking out onsets in successive stages.

Table 6. Illustration of steps for beat detection in *Kuditta Mettu Sollukattu*. We use $T_w = 0.6$ sec., $period_{max} = 1.6$ sec., $period_{min} = 1.2$ sec., $period_{th} = 2.0$ sec. T_{anno} shows the set of time-stamps in annotation. These are used as reference for validation.

Bol	**tei**	**hat**	**tei**	**hi**	**tei**	**hat**	**tei**	**hi**	**tei**	**hat**	**tei**	**hi**	**tei**	**hat**	**tei**	**hi**
T_{anno}	2.68	3.91	5.11	6.27	7.52	8.74	9.89	11.06	12.27	13.39	14.51	15.60	16.76	17.90	19.03	20.18
OSE	2.69	4.00	5.15	6.28	7.53	8.75	9.90	11.08	12.34	13.49	14.52	15.62	16.77	17.99	19.03	20.19
	2.76	4.80		~~6.35~~		~~8.83~~		~~11.15~~		13.95		~~15.69~~		18.47	19.09	~~20.26~~
				6.88		9.60		11.68		14.22		16.17		~~18.75~~		
				~~7.21~~				11.98								
L_{max}	2.69	4.00	5.15	6.28	7.53	8.75	9.90	11.08	12.34	13.49	14.52	15.62	16.77	17.99	19.03	20.19
	2.76	4.80		6.88		9.60		11.68		~~13.95~~		16.17		~~18.47~~	19.09	
								~~11.98~~		~~14.22~~						
L_{max}	2.69	4.00	5.15	6.28	7.53	8.75	9.90	11.08	12.34	13.49	14.52	15.62	16.77	17.99	19.03	20.19
(pruned)	~~2.76~~	~~4.80~~		~~6.88~~		~~9.60~~		~~11.68~~				~~16.17~~			~~19.09~~	
Beats	2.69	4.00	5.15	6.28	7.53	8.75	9.90	11.08	12.34	13.49	14.52	15.62	16.77	17.99	19.03	20.19

3.5 Results of Audio Event Detection

Now we present the beat detection results and compare our algorithm with the well-known algorithm of Ellis [4] using our recorded data set.

Audio Data Set: Recorded audio data of *Sollukattu*'s are not available for research. Hence we have created a benchmark data set with the help of performers from a dance school[11].

Sollukattu's have been recorded by *Zoom H2n Portable Handy Recorder*. For each of the 23 *Sollukattu*'s we have recorded 6 sets performed by 4 (3 female and 1 male) accomplices. Of these, two sets have so far been annotated (sample annotations are shown in Tables 2, 3, 6 and 8) by experts by marking every beat in the audio file as a range of time-stamp of its occurrence. The accompanying *bol* for every beat is also annotated. One of the annotated sets[12] is taken as the golden audio and used for the recording of the videos.

Result Analysis: We now present the results of beat detection in Table 7 for all *Sollukattu*'s using the annotated set. For the i^{th} annotated beat event[13] α_a^i

[11] Natanam Kalakshetra, Kolkata, India
[12] This data set is available at: http://cse.iitkgp.ac.in/resgrp/hci/
[13] We consider only $\alpha^f \mid \alpha^h$

in *Sollukattu* s, let the time range be $[\tau_b(\alpha_a^i), \tau_e(\alpha_a^i)]$ and let the corresponding detected beat be α_d^i with time-stamp $\tau(\alpha_d^i)$. The error in detected time is defined as $\epsilon_i = \tau(\alpha_d^i) - \tau_b(\alpha_a^i)$. The *Absolute Error* is defined as $E_{abs}^i = |\epsilon_i|$ and the *Relative Error* is defined as $E_{rel}^i = E_{abs}^i / T$, where T is the tempo period of s. If s has n beats in its bar, then we define the following error metrics for accuracy:

1. $Max(s) = \max_{i=1}^{n} E_i$
2. $85_{ptl}(s) = 85\ percentile\ in\ E_i, 1 \leq i \leq n$. That is, 85% of the errors are less than $85_{ptl}(s)$.
3. $Median(s) = \text{median}_{i=1}^{n} E_i$. That is, half of the errors are less than $Median(s)$.

where $E_i = E_{abs}^i$ or E_{rel}^i.

Table 7. Result of beat detection for all *Sollukattu*'s using $T_w = 0.6$ sec., $period_{max} = 1.6$ sec., $period_{min} = 1.2$ sec., $period_{th} = 2.0$ sec.. We compute several statistics for E_{abs} and E_{rel} for analysis. The absolute error E_{abs} as the difference between the annotated and detected time of a beat. Relative error E_{rel} is computed as a percentage of the period of the *Sollukattu*.

			E_{abs}			E_{rel}			
Sr. No.	***Sollukattu***	**Tempo Period**	**Max**	$\mathbf{85_{ptl}}$	**Med-ian**	**Max**	$\mathbf{85_{ptl}}$	**Med-ian**	**Remarks**
1	Joining A	1.18	0.13	0.11	0.02	11	9	2	
2	Joining B	1.52	0.12	0.11	0.01	8	7	1	
3	Joining C	1.17	0.12	0.01	0.01	10	1	1	
4	Kartari Utsanga	1.07	0.15	0.11	0.05	14	10	5	
5	Kuditta Mettu	1.16	0.11	0.08	0.01	9	7	1	
6	Kuditta Nattal A	0.99	**0.28**	0.06	0.01	**29**	6	1	2 outliers
7	Kuditta Nattal B	1.30	0.08	0.07	0.05	6	5	4	
8	Kuditta Tattal	1.21	0.22	0.05	0.01	18	4	1	
9	Natta	1.39	0.08	0.07	0.01	6	5	1	
10	Paikkal	1.58	0.12	0.10	0.07	8	6	4	
11	Pakka	1.21	**0.50**	0.13	**0.10**	**41**	11	8	1 outlier
12	Sarika	0.93	0.15	0.05	0.03	16	6	3	
13	Tatta A	1.51	**0.39**	**0.38**	0.10	**26**	**25**	6	2 outliers
14	Tatta B	1.36	0.06	0.05	0.03	5	4	2	
15	Tatta C	1.56	0.13	0.13	0.07	9	8	4	
16	Tatta D	1.35	0.16	0.14	**0.11**	12	10	8	7 outliers
17	Tatta E	1.17	**0.53**	0.14	0.04	**45**	12	3	1 outlier
18	Tatta F	1.21	0.15	0.13	0.05	13	10	4	
19	Tatta G	1.32	0.24	**0.20**	**0.13**	18	**15**	10	6 outliers
20	Tei Tei Dhatta	1.41	0.12	0.11	0.06	8	8	4	
21	Tirmana A	1.23	0.04	0.04	0.01	4	3	1	
22	Tirmana B	1.22	0.10	0.09	0.04	8	8	3	
23	Tirmana C	1.46	**0.41**	**0.33**	0.02	**28**	**22**	1	2 outliers

We compute the above error metrics for E_{abs} and E_{rel} in Table 7. Using 0.25 sec., 0.15 sec., and 0.10 sec. as cutoffs respectively for Max, 85_{ptl} and $Median$, we have marked outlier measures in the table with underline. On the detected beats also we have computed the outliers for these values and summarized their number under the *Remarks* column. There are 21 outliers in detection of 377 beats in total. Hence, 356 beats are detected correctly. So we achieve an accuracy of 94%.

It may be noted that 13 of the 21 outliers come from *Tatta D* and *Tatta G*. This is due to higher variation of inter-beat time in these cases. As expected, more outliers are observed when the inter-beat times vary more widely.

Next we compare the accuracy of our results against the algorithm by Ellis [4].

Comparison with Ellis' [4] Algorithm: In Table 8, we compare the results of beat detection for *Pakka Sollukattu* by our method against [4] by computing the recall and precision in each case. Ellis' method achieves 100% recall at only 25% precision, while our method achieves 97% recall at 97% precision. However, this comparison is not exactly apple-to-apple because Ellis' method estimates the tempo period from the signal (during the dynamic programming stage) while we use a preset range of tempo periods and a tempo threshold (Algorithm 3).

So in Table 9 we study the accuracy of the estimation of tempo period that Ellis' method performs internally. The method makes two guesses for *Slower* and *Faster* tempo (in terms of *bpm*) and uses a *Strength* parameter for the final choice. If $Strength < 0.5$, it chooses the *Faster* tempo, else it chooses the *Slower*. Out of 23 cases, it gets the tempo period right in only 5 cases and hence the beat detection results degrade.

Finally, we tweak the algorithm of Ellis by inputting the correct tempo period for detecting the beats. We then compare the recall and precision of Ellis' method (with estimated as well as given tempo period) and our method (given a global range of tempo periods) in Table 10. We find that given the tempo period, the precision of Ellis' method improves (or remains same) in 22 cases (96%) while the recall degrades in 15 cases (65%). Our method has a better (or equal) precision in 18 cases (78%) and a better (or equal) recall in 19 cases (83%). Overall we achieve more than 80% precision for over 80% recall in 19 cases (83%). So we do better in terms of our pruning and detection strategies (Algorithms 2 and 3). We use the beats detected by our method in the synchronization with key video frames.

Next we discuss the video event detection and event synchronization.

4 Video Event Detection

We primarily detect no motion[14] (ν^{nm} events) in the video. Given that ν^{nm} and ν^{tr} must alternate in the video, we then deduce the ν^{tr} events. We detect no motion from the co-occurrence of the no motion in the RGB and Skeleton data of Kinect by (1) Frame Differences in RGB data and (2) Velocity acceleration of skeleton Joints.

[14] Actually, slow or low motion in the video as cutoff by a threshold

Table 8. Comparison of beat detection results between Ellis' method [4] and our method for *Pakka Sollukattu* (data file = Pakka_14_HB1). For every *beat / bol* (col. 1) the range of estimated time as manually marked is shown under *Annotated Beat Range* (cols. 2-3). While Ellis' method detects all beats correctly (col. 4), it spuriously detects almost 100% (col. 5) and 200% (cols. 6-8) beats respectively within and outside the annotated time range. Hence it achieves 100% recall at 25% precision (127 beats detected for 32 correct beats). In contrast, our method detects 31 out of 32 beats correctly (col. 9) for 97% recall but detects only one spurious beat (col. 10) for 97% precision.

				Ellis' Method					**Our Method**	
				Within Range		*Outside Range*			*Within Range*	
Beat No.	**Bol**	**Annotated Beat Range**		Correct Beat	Spurious Beat	Spurious Beat			Correct Beat	Spurious Beat
(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
1	ta	2.160	2.642	2.182	2.490	2.798	3.082	3.366	2.180	
2	tei	3.481	4.088	3.654	3.938	4.226	4.558		3.634	
3	tei	4.855	5.426	4.894	5.226	5.558	5.906		4.988	
4	tat	6.194	6.747	6.242	6.578	6.910	7.234		6.312	
5	dhit	7.479	8.032	7.554	7.870	8.190	8.514		7.541	
6	tei	8.764	9.336	8.822	9.138	9.450	9.778		8.808	
7	tei	10.067	10.639	10.114	10.434	10.750	11.086		10.199	
8	tat	11.353	11.817	11.390	11.706	12.022	12.346		11.455	
9	ta	12.602	13.155	12.670	12.982	13.298	13.602		12.785	
10	tei	13.905	14.423	13.926	14.230	14.534	14.842		14.037	
11	tei	15.101	15.619	15.154	15.466	15.778	16.082		15.260	
12	tat	16.351	16.886	16.390	16.690	16.986	17.298		16.483	
13	dhit	17.564	18.100	17.610	17.898	18.186	18.490		17.669	
14	tei	18.760	19.296	18.790	19.102	19.410	19.734		18.778	
15	tei	19.974	20.545	20.030	20.326	20.622	20.918		20.149	
16	tat	21.170	21.670	21.218	21.506	21.798	22.102		21.208	
17	ta	22.312	22.884	22.402	22.686	22.974	23.278		22.499	
18	tei	23.562	24.097	23.610	23.906	24.206	24.506		23.602	
19	tei	24.740	25.347	24.806	25.106	25.410	25.718		24.868	
20	tat	25.990	26.507	26.018	26.318	26.614	26.906		26.110	
21	dhit	27.114	27.632	27.202	27.498	27.794	28.082		27.268	
22	tei	28.364	28.917	28.402	28.694	28.990	29.294		28.391	
23	tei	29.524	30.095	29.594	29.882	30.170	30.478		29.654	
24	tat	30.773	31.345	30.814	31.106	31.402	31.706		30.804	
25	ta	31.934	32.523	32.010	32.298	32.590	32.894		32.099	
26	tei	33.165	33.683	33.194	33.482	33.770	34.066		33.271	
27	tei	34.343	34.843	34.366	34.658	34.946	35.238		34.352	35.514
28	tat	35.521	36.021	35.526	35.814	36.102	36.390			
29	dhit	36.610	37.128	36.678	36.958	37.242	37.534		36.743	
30	tei	37.806	38.324	37.834	38.118	38.406	38.698		37.945	
31	tei	38.966	39.466	38.998	39.282	39.570	39.866		39.112	
32	tat	40.145	40.644	40.190	40.510				40.263	

(*All times are in sec*)

Table 9. Estimation of tempo period by Ellis' method [4]

Sollukattu	Actual Tempo Period	Slower Estimate		Faster Estimate		Strength	Estimated Tempo Period	Remarks
		bpm	Period	bpm	Period			
(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Joining A	1.18	55.147	1.09	110.294	0.54	0.05	0.54	Wrong
Joining B	1.52	32.189	1.86	64.378	0.93	0.08	0.93	Right
Joining C	1.17	52.083	1.15	104.167	0.58	0.63	1.15	Wrong
Kartari Utsanga	1.07	59.055	1.02	118.110	0.51	0.41	0.51	Wrong
Kuditta Mettu	1.16	100.000	0.60	200.000	0.30	0.65	0.60	Wrong
Kuditta Nattal A	0.99	63.559	0.94	127.119	0.47	0.16	0.47	Wrong
Kuditta Nattal B	1.30	46.296	1.30	92.593	0.65	0.22	0.65	Wrong
Kuditta Tattal	1.21	101.351	0.59	202.703	0.30	0.74	0.59	Wrong
Natta	1.39	43.860	1.37	87.719	0.68	0.28	0.68	Wrong
Paikkal	1.58	38.660	1.55	77.320	0.78	0.11	0.78	Wrong
Pakka	1.21	100.000	0.60	200.000	0.30	0.66	0.60	Wrong
Sarika	0.93	61.983	0.97	123.967	0.48	0.68	0.97	Right
Tatta A	1.51	41.899	1.43	83.799	0.72	0.14	0.72	Wrong
Tatta B	1.36	22.189	2.70	44.379	1.35	0.01	1.35	Right
Tatta C	1.56	39.063	1.54	78.125	0.77	0.31	0.77	Wrong
Tatta D	1.35	45.455	1.32	90.909	0.66	0.19	0.66	Wrong
Tatta E	1.17	36.765	1.63	110.294	0.54	0.14	0.54	Wrong
Tatta F	1.21	48.387	1.24	96.774	0.62	0.65	1.24	Right
Tatta G	1.32	45.455	1.32	90.909	0.66	0.51	1.32	Right
Tei Tei Dhatta	1.41	66.964	0.90	133.929	0.45	0.32	0.45	Wrong
Tirmana A	1.23	47.468	1.26	94.937	0.63	0.06	0.63	Wrong
Tirmana B	1.22	50.000	1.20	100.000	0.60	0.13	0.60	Wrong
Tirmana C	1.46	90.361	0.66	180.723	0.33	0.87	0.66	Wrong

(*All times are in sec*)
(*bpm ≡ beats per minute. Period = 60/bpm*)

4.1 Frame Differences in RGB Stream

Frame difference or image sequence difference method refers to a very small time intervals of the two images before and after the pixel based on the time difference, and then using a threshold to extract the image regions of the movement. The image is then binarized based on motion (marked as 1) and no motion (marked as 0). We sum the the non-zero pixels (having motion) present in the image and then label it as motion or no motion frame based on a threshold.

4.2 Velocity-Acceleration in Skeleton Stream

We compute the velocity and acceleration for 4 joint points (Wrist, Elbow, Knee and Ankle) of the Kinect skeleton corresponding to every RGB frame. If the $StartPoint$ is (x_1, y_1, z_1) and the $EndPoint$ is (x_2, y_2, z_2) then the instantaneous velocity is $v = (v_x, v_y, v_z) = velocity(x_2 - x_1, y_2 - y_1, z_2 - z_1)$ and the instanta-

Table 10. Comparison of Precision and Recall between Ellis' [4] and our methods

Sollukattu	Ellis' Method using Estimated[a] Tempo Period		Ellis' Method using Given[b] Tempo Period		Our Method using Given Ranges of[c] Tempo Periods	
	Precision	**Recall**	**Precision**	**Recall**	**Precision**	**Recall**
Joining A	38	83	71	83	86	100
Joining B	73	92	54	58	100	100
Joining C	63	95	100	70	100	100
Kartari Utsanga	96	98	100	52	100	100
Kuditta Mettu	25	100	50	100	81	81
Kuditta Nattal A	37	96	40	25	71	92
Kuditta Nattal B	74	96	93	58	100	100
Kuditta Tattal	25	94	48	63	88	88
Natta	50	100	100	94	81	81
Paikkal	100	75	100	75	100	100
Pakka	25	100	97	97	97	97
Sarika	50	100	97	94	97	97
Tatta A	39	100	88	58	100	75
Tatta B	48	92	100	83	100	100
Tatta C	68	100	86	57	75	100
Tatta D	65	100	100	75	94	100
Tatta E	18	100	75	100	65	92
Tatta F	22	100	88	100	88	100
Tatta G	30	100	91	71	100	100
Tei Tei Dhatta	65	100	96	72	100	100
Tirmana A	68	100	68	100	91	82
Tirmana B	87	98	87	98	100	100
Tirmana C	41	100	100	58	95	75

[a]: *Original dynamic programming method of Ellis*
[b]: *Ellis' method where the actual tempo period has been set for each Sollukattu*
[c]: *Our method where a common range of tempo periods are set for all*

neous acceleration is $a = (a_x, a_y, a_z) = acceleration(v_{x_2} - v_{x_1}, v_{y_2} - v_{y_1}, v_{z_2} - v_{z_1})$. If acceleration $|a|$ is less than a threshold then no motion is inferred.

Finally, a frame is marked with no motion (ν^{nm}) if it does not show symptoms of motion from frame difference as well as velocity-acceleration. The range of consecutive no motion frames forms $\eta_{est}(\nu^{nm})$ (the frames preceding and following this range must have motion).

4.3 Results of Video Event Detection

Now we present the results for video event detection using our data set.

Video Data Set: *Adavu*'s are captured at 30 fps by *Microsoft Kinect XBox (Kinect 1.0)* using a special purpose capture software *nuiCapture* [1]. Every recorded file comprises RGB, depth, skeleton, and audio streams. For each of 50 variants of 15 *Adavu*'s, we have recorded over 20 sessions each as performed by 7 dancers (4 female and 3 male) giving over 1000 performance videos to analyze. 10% of the data has so far been annotated[15] by experts at frame level. An example for annotated Audio-Visual Data of *Kuditta Mettu Adavu* is shown in Table 12.

Result Analysis: We compare the video events by using the above algorithms with the manually annotated video events. First, we get a sequence of no-motion frame ranges from the detection algorithm (as in the manual video annotation given in Table 5). Next, we determine the number of overlapped ranges between detected video (DV) and annotated video (AV) events and compute precision and recall of the detection as:

$$Precision = \frac{Number\ of\ overlapped\ ranges\ between\ DV\ and\ AV}{Number\ of\ DV\ events} * 100$$

$$Recall = \frac{Number\ of\ overlapped\ ranges\ between\ DV\ and\ AV}{Number\ of\ AV\ events} * 100$$

The results are given Table 11. If the precision and recall both are $\geq 75\%$ then we mark it as *Good*, if their minimum is within 74-50% then we mark it as *Moderate*, otherwise mark the result as *Poor*. We achieve 84% accuracy for *Good* and *Moderate* quality detection of video events.

As expected, we achieve *Good* results where the distinction between key postures and transitions is clear in the dance sequence. In a few *Adavu*'s like *Kuditta Nattal 1*, *Kuditta Nattal 5*, and *Kuditta Tattal 1* the dancer holds the key postures in over only a few of frames (generally it is 15-20 frames, but in these cases it is down to 2-3 frames). Such key postures are missed out in detection especially because the estimated skeletons are not stable and well-formed. Thus the detection performance goes down from *Good* to *Poor* depending on the clarity of the key posture in the sequence itself.

5 Estimation of Event Synchronization

For a detected beat α^{fp}, we have the estimated time-stamp $\tau(\alpha^{fp})$ from Section 3.4. We convert this to frame number $\eta(\alpha^{fp})$ of the video (using 30 fps). We use a buffer threshold of ± 5 frames to get the frame interval $\eta_{est}(\alpha^{fp})$ of α^{fp} as $[\eta(\alpha^{fp}) - 5, \eta(\alpha^{fp}) + 5]$. Similarly, for a detected no motion event ν^{nm}, we have the estimated frame range as $\eta_{est}(\nu^{nm})$ from Section 4.

Finally, the sync event ψ^{fb} is inferred as

$$\eta_{est}(\alpha^{fb}) \cap \eta_{est}(\nu^{nm}) \neq \phi$$

Synchronization in annotated audio and video events are shown in Table 12.

[15] Part of this data set is available at: http://cse.iitkgp.ac.in/resgrp/hci/

Table 11. Results of Video Event Detection

Sr.	Adavu	Precision	Recall	Remarks	Sr.	Adavu	Precision	Recall	Remarks
1	Tatta 1	100.00	100.00	Good	26	Kuditta Nattal 6	57.14	100.00	Moderate
2	Tatta 2	88.89	100.00	Good	27	Kuditta Tattal 1	85.00	35.42	Poor
3	Tatta 3	80.00	100.00	Good	28	Paikkal 1	50.00	75.00	Moderate
4	Tatta 4	94.12	100.00	Good	29	Paikkal 2	80.00	100.00	Good
5	Tatta 5	90.48	95.00	Good	30	Paikkal 3	70.00	87.50	Moderate
6	Tatta 6	81.82	75.00	Good	31	Tei Tei Dhatta 1	71.43	62.50	Moderate
7	Tatta 7	100.00	92.86	Good	32	Tei Tei Dhatta 2	50.00	87.50	Moderate
8	Tatta 8	100.00	100.00	Good	33	Tei Tei Dhatta 3	50.00	12.50	Poor
9	Natta 1	77.78	87.50	Good	34	Katti or Kartari 1	61.54	100.00	Moderate
10	Natta 2	80.00	100.00	Good	35	Utsanga 1	100.00	75.00	Good
11	Natta 3	94.12	100.00	Good	36	Mandi 1	51.11	71.88	Moderate
12	Natta 4	37.84	87.50	Poor	37	Mandi 2	86.36	59.38	Moderate
13	Natta 5	82.35	87.50	Good	38	Sarrikkal 1	60.53	71.88	Moderate
14	Natta 6	93.75	93.75	Good	39	Sarrikkal 2	80.00	66.67	Moderate
15	Natta 7	100.00	50.00	Moderate	40	Sarrikkal 3	54.55	56.25	Moderate
16	Natta 8	100.00	58.33	Moderate	41	Tirmana 1	62.50	50.00	Moderate
17	Pakka 1	77.78	87.50	Good	42	Tirmana 2	47.37	50.00	Poor
18	Kuditta Mettu 1	80.00	50.00	Moderate	43	Tirmana 3	72.22	72.22	Moderate
19	Kuditta Mettu 2	100.00	50.00	Moderate	44	Sarika 1	90.91	62.50	Moderate
20	Kuditta Mettu 3	87.50	82.35	Good	45	Sarika 2	92.31	75.00	Good
21	Kuditta Nattal 1	85.71	75.00	Good	46	Sarika 3	100.00	100.00	Good
22	Kuditta Nattal 2	91.67	78.57	Good	47	Sarika 4	57.14	50.00	Moderate
23	Kuditta Nattal 3	72.73	66.67	Moderate	48	Joining 1	75.00	100.00	Good
24	Kuditta Nattal 4	50.00	36.36	Poor	49	Joining 2	33.33	33.33	Poor
25	Kuditta Nattal 5	80.00	28.57	Poor	50	Joining 3	33.33	40.00	Poor

Table 12. Annotation of Audio-Visual Data of *Kuditta Mettu Adavu*

	Audio Annotation				Video Annotation	
	(In Time (Sec))		(In Frame #)		(In Frame #)	
Events	**Start**	**End**	**Start**	**End**	**Start**	**End**
ν_1^{nm} $[\alpha_1^{fb}(\text{tei})]$	2.681	3.218	80	97	70	99
ν_2^{nm} $[\alpha_2^{fb}(\text{hat})]$	3.912	4.247	117	127	104	124
ν_3^{nm} $[\alpha_3^{fb}(\text{tei})]$	5.108	5.541	153	166	146	172
ν_4^{nm} $[\alpha_4^{fb}(\text{hi})]$	6.269	6.681	188	200	177	191
ν_5^{nm} $[\alpha_5^{fb}(\text{tei})]$	7.523	7.975	226	239	215	245
ν_6^{nm} $[\alpha_6^{fb}(\text{hat})]$	8.742	9.125	262	274	250	262
ν_7^{nm} $[\alpha_7^{fb}(\text{tei})]$	9.891	10.375	297	311	288	314
ν_8^{nm} $[\alpha_8^{fb}(\text{hi})]$	11.064	11.563	332	347	318	345
ν_9^{nm} $[\alpha_9^{fb}(\text{tei})]$	12.271	12.698	368	381	359	386
ν_{10}^{nm} $[\alpha_{10}^{fb}(\text{hat})]$	13.386	13.819	402	415	391	410
ν_{11}^{nm} $[\alpha_{11}^{fb}(\text{tei})]$	14.512	14.969	435	449	430	451
ν_{12}^{nm} $[\alpha_{12}^{fb}(\text{hi})]$	15.603	16.109	468	483	456	470
ν_{13}^{nm} $[\alpha_{13}^{fb}(\text{tei})]$	16.764	17.201	503	516	493	520
ν_{14}^{nm} $[\alpha_{14}^{fb}(\text{hat})]$	17.902	18.302	537	549	526	542
ν_{15}^{nm} $[\alpha_{15}^{fb}(\text{tei})]$	19.028	19.476	571	584	565	587
ν_{16}^{nm} $[\alpha_{16}^{fb}(\text{hi})]$	20.178	20.630	605	619	591	620

5.1 Results of Event Synchronization

After audio and video event detection we get time-stamp of beats from the audio signal and range of Key Posture from the video stream. Next we compute the quality of the match using the following measures:

Matching Detected Video (DV) events against Annotated Audio (AA) events:

$$Measure\,of\,Match\,(DV{-}AA) = \frac{Number\ of\ matched\ DV\ and\ AA\ events}{Number\ of\ AA\ Events} * 100$$

Matching Detected Video (DV) events against Detected Audio (DA) events:

$$Measure\,of\,Match\,(DV{-}DA) = \frac{Number\ of\ matched\ DV\ and\ DA\ events}{Number\ of\ DA\ Events} * 100$$

Detected audio and video events of *Kuditta Mettu 3 Adavu* are shown in Table 13. In 2 out of the 16 events, there is no overlap. Hence, we achieve 87.5% sync between the DA and DV events.

Table 13. Detected Audio & Video Events of *Kuditta Mettu*

		Audio Time to	**Video Frames**	
Events	**Detected Beats**	**Video Frame**	Start	End
ν_1^{nm} [α_1^{fb}(tei)]	2.742	82	78	83
ν_2^{nm} [α_2^{fb}(hat)]	3.964	119	95	119
ν_3^{nm} [α_3^{fb}(tei)]	4.798	144	143	150
ν_4^{nm} [α_4^{fb}(hi)]	6.280	188	157	198
ν_5^{nm} [α_5^{fb}(tei)]	7.215	216	215	247
ν_6^{nm} [α_6^{fb}(hat)]	8.753	263	252	265
ν_7^{nm} [α_7^{fb}(tei)]	9.600	288	289	299
ν_8^{nm} [α_8^{fb}(hi)]	11.156	**335**	**303**	**330**
ν_9^{nm} [α_9^{fb}(tei)]	12.333	370	364	389
ν_{10}^{nm} [α_{10}^{fb}(hat)]	13.485	405	392	405
ν_{11}^{nm} [α_{11}^{fb}(tei)]	14.566	437	428	437
ν_{12}^{nm} [α_{12}^{fb}(hi)]	15.624	469	442	481
ν_{13}^{nm} [α_{13}^{fb}(tei)]	16.776	503	500	539
ν_{14}^{nm} [α_{14}^{fb}(hat)]	17.973	**539**		
ν_{15}^{nm} [α_{15}^{fb}(tei)]	19.030	571	565	572
ν_{16}^{nm} [α_{16}^{fb}(hi)]	20.189	606	575	621

Result Analysis: In Table 14 we present the summary of sync results and analyze the quality of sync. We also achieve 72% accuracy of *Good* (DV-DA > 75%) or *Moderate* (50% < DV-DA < 75%) synchronization. We explain the reasons behind the poor results below:

1. We detect motion or no-motion of a frame from the change in the current frame with respect to the previous frame. If the change in the consecutive frames are very low then very slow motion gets falsely detected as no-motion. Hence, number of detected Key Posture is becomes more than number of annotated key postures. This is happening in *Paikkal 3*.
2. In some *Adavu*'s like *Kuditta Nattal 4*, *Tei Tei Dhatta 3*, *Kuditta Nattal 5*, *Natta 7* and *Natta 8* the dancer holds the key posture for very small span of time. Hence, the Key Posture detection fails for the reasons explained in Section 4.3 and less Key Postures are detected than the actual annotated.

Table 14. Results of Sync Events in percentage of Match

Sr.	**Adavu**	**DV-AA**	**DV-DA**	**Remark**	**Sr.**	**Adavu**	**DV-AA**	**DV-DA**	**Remark**
1	Tatta 1	100.00	100.00	Good	26	Kuditta Nattal 6	100.00	41.94	Poor
2	Tatta 2	100.00	100.00	Good	27	Kuditta Tattal 1	35.42	31.25	Poor
3	Tatta 3	100.00	100.00	Good	28	Paikkal 1	75.00	56.25	Moderate
4	Tatta 4	100.00	100.00	Good	29	Paikkal 2	56.25	56.25	Moderate
5	Tatta 5	93.75	94.12	Good	30	Paikkal 3	31.25	56.25	Moderate
6	Tatta 6	75.00	58.82	Moderate	31	Tei Tei Dhatta 1	62.50	62.50	Moderate
7	Tatta 7	92.86	81.25	Good	32	Tei Tei Dhatta 2	87.50	68.75	Moderate
8	Tatta 8	100.00	100.00	Good	33	Tei Tei Dhatta 3	12.50	12.50	Poor
9	Natta 1	87.50	81.25	Good	34	Katti or Kartari 1	100.00	54.17	Moderate
10	Natta 2	100.00	100.00	Good	35	Utsanga 1	50.00	29.17	Poor
11	Natta 3	100.00	93.75	Good	36	Mandi 1	64.58	87.23	Good
12	Natta 4	90.63	81.25	Good	37	Mandi 2	39.58	40.43	Poor
13	Natta 5	87.50	75.00	Good	38	Sarrikkal 1	52.08	68.09	Moderate
14	Natta 6	93.75	87.50	Good	39	Sarrikkal 2	37.00	38.31	Poor
15	Natta 7	50.00	50.00	Moderate	40	Sarrikkal 3	56.25	65.96	Moderate
16	Natta 8	58.33	43.75	Poor	41	Tirmana 1	50.00	72.73	Moderate
17	Pakka 1	50.00	65.63	Moderate	42	Tirmana 2	45.83	62.50	Moderate
18	Kuditta Mettu 1	50.00	56.25	Moderate	43	Tirmana 3	58.33	59.09	Moderate
19	Kuditta Mettu 2	81.25	50.00	Moderate	44	Sarika 1	62.50	68.75	Moderate
20	Kuditta Mettu 3	75.00	87.50	Good	45	Sarika 2	75.00	37.50	Poor
21	Kuditta Nattal 1	54.55	45.16	Poor	46	Sarika 3	53.13	56.25	Moderate
22	Kuditta Nattal 2	55.00	38.71	Poor	47	Sarika 4	50.00	28.13	Poor
23	Kuditta Nattal 3	50.00	41.94	Poor	48	Joining 1	100.00	85.71	Good
24	Kuditta Nattal 4	26.67	31.25	Poor	49	Joining 2	100.00	87.50	Good
25	Kuditta Nattal 5	26.67	18.75	Poor	50	Joining 3	37.50	56.25	Moderate

6 Conclusions

This paper is the maiden approach to characterize the *Bharatanatyam* dance form and attempt multimedia analytics for Kinect data of *Bharatanatyam Adavu*'s. In the process we make the following contributions:

1. We characterize the events of *Bharatanatyam Adavu*'s for automated analysis. First we analyze and document the structure of its music – understanding

the pattern of beats and *bol*'s in depth. Next we outline the characterization of its video in terms of key postures. Finally, we identify core synchronization issues in an *Adavu*.

2. We present a simple yet effective algorithm to detect beats in *Sollukattu*'s. We validate the results against annotated data. Overall we achieve 94% accuracy.
   We compare our results against the Ellis' algorithm [4]. Under similar conditions, our algorithm performs better. We show that the correct estimation of tempo period is crucial for accurate beat detection and the same remains elusive for now.
3. We present algorithms to detect no-motion video events and achieve 84% accuracy for it.
4. In terms of audio-video sync, we achieve 72% accuracy.
5. No annotated data of *Sollukattu*'s and *Adavu*'s is available for research. We have recorded 6 six sets of all 23 *Sollukattu*'s and 20 sessions of all 50 variants of 15 *Adavu*'s. 30% of audio and 10% of video data have already been annotated by experts.

The paper also raises several questions including:

1. *Beat Detection and Marking*: From the characterization we know that most beats are accompanied by a *bol*. Since the current approach is based on onsets, it ignores the *bol*'s. We can create a vocab of *bol*'s, detect these as utterances, and correspond with the onsets to achieve near 100% accuracy. Once *bol*'s are known, the same can be marked on the stream. Half beats also need to be detected.
   Information of *bol*'s can also be used to estimate the tempo period accurately which, as discussed, is a critical factor in beat detection.
   Estimating lead / lag between instrumental and vocal sound and the uniformity of beat-to-beat time gaps would be key problems for a *Sollukattu*.
2. *Detection of Key Frames and Audio-guided Segmentation of* Adavu*'s*: The paper presents an important characterization of the video of *Adavu* in terms of *K–* and *T–Frames*. These can be further characterized in terms of motion parameters. Based on the marked beats and *bol*'s, the video may be segmented at approximate *K–Frames* and then refined with motion estimates.
3. *Synchronization Issues*: Based on the solution of the above problems, several synchronization issues as discussed in Section 2.4 may be attempted.

It may be reiterated that the characterization of *Adavu*'s and detection of beats in *Sollukattu*'s have several applications including music segmentation, music video segmentation, estimating the synchronization of the postures with the musical beats, automatic tagging of rhythm metadata of music, synchronization correction, and the like. These can be attempted in future.

References

1. Cadavid Concepts. nuiCapture Analyze. http://nuicapture.com/, 2014. Accessed: 2016-10-15. 19

2. Matthew EP Davies and Mark D Plumbley. Context-dependent beat tracking of musical audio. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(3):1009–1020, 2007. 2
3. Simon Dixon. Evaluation of the audio beat tracking system beatroot. *Journal of New Music Research*, 36(1):39–50, 2007. 2
4. Daniel PW Ellis. Beat Tracking by Dynamic Programming. *Journal of New Music Research*, 36(1):51–60, 2007. 2, 11, 14, 16, 17, 18, 19, 24
5. Ryan Matthew Faircloth. *Combining Audio and Video Tempo Analysis for Dance Detection*. PhD thesis, University of Central Florida Orlando, Florida, 2008. 3
6. Sankalp Gulati, Vishweshwara Rao, and Preeti Rao. Meter detection from audio for indian music. In *Speech, Sound and Music Processing: Embracing Research in India*, pages 34–43. Springer, 2012. 2
7. Anssi Klapuri et al. Musical meter estimation and music transcription. In *Cambridge Music Processing Colloquium*, pages 40–45. Citeseer, 2003. 2
8. Anssi P Klapuri, Antti J Eronen, and Jaakko T Astola. Analysis of the meter of acoustic musical signals. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(1):342–355, 2006. 2
9. Costas Panagiotakis, Andre Holzapfel, Damien Michel, and Antonis A Argyros. Beat synchronous dance animation based on visual analysis of human motion and audio analysis of music tempo. In *International Symposium on Visual Computing*, pages 118–127. Springer, 2013. 3
10. Takaaki Shiratori, Atsushi Nakazawa, and Katsushi Ikeuchi. Detecting dance motion structure through music analysis. In *Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on*, pages 857–862. IEEE, 2004. 3
11. Takaaki Shiratori, Atsushi Nakazawa, and Katsushi Ikeuchi. Detecting dance motion structure using motion capture and musical information. In *Proc. International Conference on Virtual Systems and Multimedia (VSMM)*, volume 3, 2004. 3
12. Rajeswari Sridhar and TV Geetha. Raga identification of carnatic music for music information retrieval. *International Journal of recent trends in Engineering*, 1(1), 2009. 2
13. Ajay Srinivasamurthy, Sidharth Subramanian, Gregoire Tronel, and Parag Chordia. A beat tracking approach to complete description of rhythm in indian classical music. In *Proc. of the 2nd CompMusic Workshop*, pages 72–78. Citeseer, 2012. 2