2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Congress Center Hamburg
Sept 28 - Oct 2, 2015. Hamburg, Germany

Audio-Visual Beat Tracking Based on a State-Space Model for a Music Robot Dancing with Humans

Misato Ohkita, Yoshiaki Bando, Yukara Ikemiya, Katsutoshi Itoyama, and Kazuyoshi Yoshii

Abstract— **This paper presents an audio-visual beat-tracking method for an entertainment robot that can dance in synchronization with music and human dancers. Conventional music robots have focused on either music audio signals or dancing movements of humans for detecting and predicting beat times in real time. Since a robot needs to record music audio signals by using its own microphones, however, the signals are severely contaminated with loud environmental noise and reverberant sounds. Moreover, it is difficult to visually detect beat times from real complicated dancing movements that exhibit weaker repetitive characteristics than music audio signals do. To solve these problems, we propose a state-space model that integrates both audio and visual information in a probabilistic manner. At each frame, the method extracts acoustic features (audio tempos and onset likelihoods) from music audio signals and extracts skeleton features from movements of a human dancer. The current tempo and the next beat time are then estimated from those observed features by using a particle filter. Experimental results showed that the proposed multi-modal method using a depth sensor (Kinect) for extracting skeleton features outperformed conventional mono-modal methods by 0.20 (F measure) in terms of beat-tracking accuracy in a noisy and reverberant environment.**

I. Introduction

Development of entertainment robots that can interact with humans through music is one of the most attracting research directions in the field of robotics. Since various kinds of robots are expected to get into our daily lives in the future, not only task-oriented robots but also entertainment robots that people feel familiarity with have been developed. Among them are a violinist robot [1], a cheerleader robot that can move around while balancing on a ball [2], and a flutist robot that can play the flute in synchronization with a melody played by a human [3]. Since dancing is a form of expression seen in many cultures, in this paper we focus on music robots that can dance interactively with humans.

A robot that can dance synchronously with human dancers needs to adaptively control its movements while recognizing music and the movements of the dancers. Several dancing robots have already been developed. Murata *et al.* [4], for example, enabled a bipedal humanoid robot to step and sing in synchronization with musical beats, Kosuge *et al.* [5] devised a dancer robot that can predict the next step intended by a dance partner and move according to the movements of the partner, and Kaneko *et al.* [6] developed a humanoid robot that can generate natural dancing movements by using a complicated human-like dynamical system.

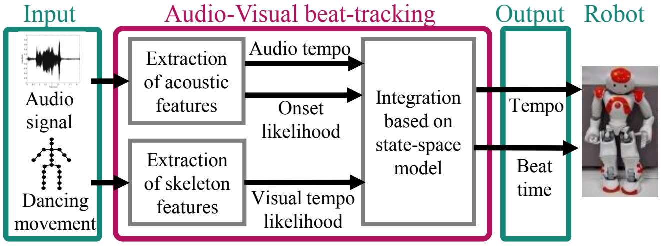

Input
Audio-Visual beat-tracking
Output
Robot
Audio signal
Dancing movement
Extraction of acoustic features
Extraction of skeleton features
Audio tempo
Onset likelihood
Visual tempo likelihood
Integration based on state-space model
Tempo
Beat time

Fig. 1. The proposed audio-visual beat-tracking method for music audio signals accompanied by dancing movements of a human

To synchronize its dancing movements with musical beats, the robot should be able to perform real-time beat tracking—*i.e.*, estimate a musical tempo and detect beat times (times that people are likely to clap)—in a noisy and reverberant environment. Many beat-tracking methods for music audio signals have been proposed. Murata *et al.* [4], for example, proposed an online method that can quickly follow tempo changes and is robust to environmental noise. This method, though, often fails to detect correct beat times for musical piece having many accented up-beats. Chu and Tsai [7], on the other hand, proposed an offline beat-tracking method that tries to detect tempos (periods) from dancing movements. It is, however, difficult to accurately analyze real musical pieces with complicated dancing movements because the accuracy of beat tracking using a single modality is limited.

In this paper, we propose a multi-modal beat tracking method that focuses on both music audio signals and dancing movements (Fig. 1)*. Audio-visual integration is widely studied in music information retrieval, and each achieved better performance than single-modal methods [8]–[13]. Music audio signals are recorded by a microphone, and skeleton information of dancing movements is obtained by using a depth sensor (*e.g.*, Microsoft Kinect) or a motion capture system. To extract acoustic features from music audio signals, we estimate audio tempos and onset likelihoods at each frame. To extract skeleton features, on the other hand, at each frame we calculate visual tempo likelihoods that indicates the likelihoods over possible tempos. We then formulate a unified state-space model that consists of latent variables (tempo and beat time) and observed variables (acoustic and skeleton features). A posterior distribution of latent variables can be estimated by using a particle filter.

Misato Ohkita, Yoshiaki Bando, Yukara Ikemiya, Katsutoshi Itoyama, and Kazuyoshi Yoshii are with the Graduate School of Informatics, Kyoto University, Sakyo-ku, Kyoto 606−8501, Japan. ohkita@kuis.kyoto-u.ac.jp,{yoshiaki, ikemiya, itoyama, yoshii}@kuis.kyoto-u.ac.jp

*Demo page: http://winnie.kuis.kyoto-u.ac.jp/members/ohkita/demo/iros2015/

978-1-4799-9993-4/15/$31.00 ©2015 IEEE

II. Related work

This section describes the related work of beat tracking using audio and visual signals.

A. Beat tracking for music audio signals

Many beat-tracking methods have been proposed for music audio signals. Dixon *et al.* [14], for example, proposed an offline method based on a multi-agent architecture in which the agents independently estimate inter-onset intervals (IOIs) of music audio signals and estimate beat times by integrating the multiple interpretations. Goto *et al.* [15] proposed a similar online method using both IOIs and chord changes as useful clues for detecting beat times. Stark *et al.* [16] proposed an online method that combines a beat-tracking method based on dynamic programming [17] with another method using a state-space model for tempo estimation [18]. The performance of this method was shown to tie with those of offline systems. These methods, however, are not sufficiently robust against noise because clean music audio signals are assumed to be given. Murata *et al.* [4] proposed a real-time method that enables a robot to step and sing according to musical beats while recording music audio signals by using an embedded microphone. This method calculates an onset spectrum at each frame and detects beat times by calculating the auto-correlation of onset spectra. Oliveiral *et al.* [19] proposed an online multi-agent method using different multi-channel preprocessing strategies (*e.g.*, sound source localization and separation) to improve the robustness of environmental noise.

B. Beat tracking for dancing movements

Several studies have been conducted for analyzing rhythmic information of dancing movements. Guedes *et al.* [20] proposed a method that estimates an audio tempo of dancing movements in a dance movie. This method can be used for estimating a tempo from periodic movements (*e.g.*, periodically putting a hand up and down) under a condition that other moving objects do not exist in a dance movie. It is difficult to use this method for complicated movements seen in real dancing performances. Chu and Tsai [7] proposed an off-line method that extracts motion trajectories of a dancer's body from a dancing movie and then detects time frames in which a characteristic point stops or rotates. They proposed a system replacing background music of a dance video by using this method.

C. Audio-visual beat tracking

There are two main approaches that use both acoustic and skeleton features for multi-modal tempo estimation and/or beat tracking. One approach focuses on predefined visual cues that tell a tempo. Weinberg *et al.* [11] developed an interactive marimba playing robot called Shimon that performs beat tracking while recognizing a visual cue (nodding one's head to the beat). Petersen *et al.* [12] proposed a method that uses a visual cue (waving a hand to control parameters of vibrato or tempo). Lim *et al.* [13] developed a robot accompanist that follows a flutist. It starts and stops its performance when it sees a visual cue, and it estimates a tempo by seeing a visual beat cue (up and down movement of the flute to the tempo) and listening to the flutist's notes.

The other approach does not use predefined visual cues. Itohara *et al.* [9] proposed an audio-visual beat-tracking method using both guitar sounds and guitarist's arm motions. They formulated a simplified model that represents a guitarist's arm trajectory as a sine wave, and integrated acoustic and skeleton features by using a state-space model. Berman *et al.* [10] proposed a beat-tracking method for ensemble robots playing with a human guitarist. To visually estimate a tempo, a method similar to [20] was used. This method can estimate the tempo from a periodic behavior, such as the head's and foot's moving up and down to the music in playing a guitar.

III. Audio-visual beat tracking

This section describes the proposed method of audio-visual beat tracking that jointly deals with both music audio signals and skeleton information of dancing movements (Fig. 1). To extract acoustic features, we use the audio beat-tracking method [4] that is robust against environmental noise and change of tempo because in a dance there are various kinds of loud noise including the sound of footsteps and the voices of audiences. To extract skeleton features, we propose a visual tempo estimation method obtained by extending Chu's offline method [7] into an online one. Furthermore, our visual tempo estimation method uses skeleton information in order to deal with general dances with complex whole-body movements. To integrate acoustic features and skeleton features in a principled manner, we formulate a probabilistic state-space model that consists of latent variables (tempo and beat times) and observed variables (acoustic and skeleton features). A posterior distribution of the latent variables is estimated by using a particle filter.

A. Problem Specification

Our goal is to estimate online the current tempo ϕ_k and the next beat time θ_{k+1} by using the history of acoustic features $\{A_1, \cdots, A_k\}$ and that of skeleton features $\{S_1, \cdots, S_k\}$, where k indicates the index of the current beat time:

Input:	History of acoustic features: $\{A_1, A_2, \cdots, A_k\}$
	History of skeleton features: $\{S_1, S_2, \cdots, S_k\}$
Output:	Current tempo: ϕ_k
	Next beat time: θ_{k+1}

This estimation step is recursively executed each time the current time reaches the predicted next beat time. Note that the acoustic features are extracted from music audio signals (Section III-B) and the skeleton features are extracted from dancing movements (Section III-C).

B. Extraction of acoustic features

The acoustic features A_k of the current beat time θ_k is a pair of frame-based onset likelihoods $F_k(t)$ over time between the previous beat time θ_{k-1} and the current beat time θ_k and the instantaneous audio tempo M_k at the current beat time θ_k, where t is a frame index (a typical frame-shift interval is 10 msec). To extract these features from a music

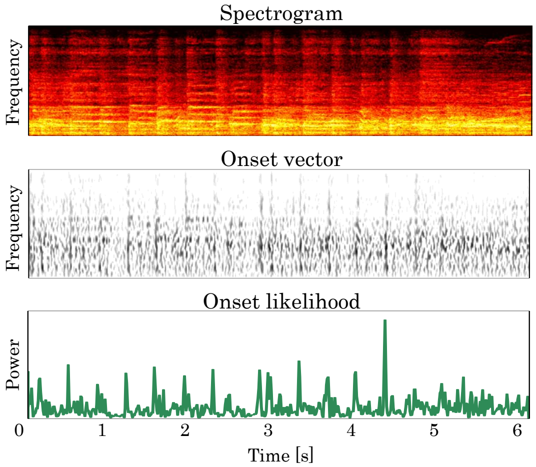


Fig. 2. From top to bottom: a spectrogram of a popular musical piece, onset vectors, and onset likelihoods.

audio signal $y(t)$, we use an audio beat-tracking method [4]. This method calculates an onset spectrum at each frame and detects beat times by calculating the autocorrelation of a sequence of onset spectra. We use this method that is robust against robot noise and the change of tempo.

1) Onset likelihood: The onset likelihood $F_k(t)$ at frame t indicates how likely the frame is to include an onset. This feature can be extracted by focusing on the power increase around that frame (Fig. 2). The short-time Fourier transform is first applied to the input audio signal $y(t)$ for obtaining frequency spectra. The Hanning window is used as a window function. The obtained spectra are sent to a mel-scale filter bank that changes the linear frequency scale to the mel-scale frequency scale, to reduce the computational cost. Let $mel(t,f)$ be a mel-scale spectra, where f $(1 \leq f \leq F_\omega)$ represents a mel-scale frequency.

A Sobel filter is then used for detecting frequency bins with rapid power increase from the spectra $mel(t,f)$. Since the Sobel filter has commonly been used for extracting edges from images, it can be applied to a music spectrogram by regarding it as a image (two-dimensional matrix). The onset vectors $d(t,f)$ are estimated by rectifying the output of the Sobel filter. The onset likelihood $F_k(t)$ is obtained by accumulating the values of the elements of the onset vector $d(t,f)$ over frequencies as follows:

$$F_k(t) = \sum_{f=1}^{F_\omega} d(t,f). \tag{1}$$

2) Audio tempo: The audio tempo M_k at the current beat time θ_k indicates an instantaneous tempo. This feature can be extracted via pattern matching in the time-frequency domain for the onset vectors $d(t,f)$. Normalized cross-correlation is used as a pattern matching function as follows:

$$R(t,s) = \frac{\sum\limits_{j=1}^{F_\omega} \sum\limits_{i=0}^{P_\omega - 1} d(t-i,j) d(t-s-i,j)}{\sqrt{\sum\limits_{j=1}^{F_\omega} \sum\limits_{i=0}^{P_\omega - 1} d(t-i,j)^2 \sum\limits_{j=1}^{F_\omega} \sum\limits_{i=0}^{P_\omega - 1} d(t-s-i,j)^2}}. \tag{2}$$

where P_ω is a window length for pattern matching and s is a shift parameter. Let I_1 and I_2 be shift parameters that take the top two largest local peaks of $R(t,s)$, respectively. The audio tempo $M(t)$ at frame t is given by

$$M(t) = \begin{cases} 2I_{n'} & \text{if } \max(|2I_{n'} - I_1|, |2I_{n'} - I_2|) < \delta \\ 3I_{n'} & \text{if } \max(|3I_{n'} - I_1|, |3I_{n'} - I_2|) < \delta \\ I_1 & \text{otherwise,} \end{cases} \tag{3}$$

where $I_{n'} = |I_1 - I_2|$ and δ is a tolerance parameter. To avoid the miss, tempo is limited from n' beats per minute (BPM) to $2n'$ (BPM) ($n' = 90$ in this paper). The audio tempo M_k is given by $M_k = M(\theta_k)$.

C. Extraction of skeleton features

The skeleton feature S_k of the current beat time θ_k is a set of visual tempo likelihoods $S_k(u)$ over possible tempo u. To extract this feature, we used an online version of a visual tempo estimation method proposed by Chu and Tsai [7]. Although the original method aims to analyze the movements of characteristic points detected from a dance movie, our method can deal with the movements of joints of a human dancer. Let $\{\boldsymbol{b}_1(t), \cdots, \boldsymbol{b}_J(t)\}$ be a set of the 3D coordinates of joints (*e.g.*, neck and hip), where J is the number of joints. The value of J depends on a device (*e.g.* Kinect or a motion capture system) used for analyzing the movements of a human dancer.

The skeleton information $\{\boldsymbol{b}_1(t), \cdots, \boldsymbol{b}_J(t)\}$ is obtained according to three steps (Fig. 3). Firstly, we estimate time frames in which some joints stop and turn. This is justified by the fact that dancers tend to stop or turn their joints at beat times. Secondly, we make a signal from a discrete set of the detected frames for each joint. Finally, we obtain the likelihood of each possible tempo by applying the Fourier transform to the signals of all joints independently and accumulating the obtained spectra over all joints.

1) Detection of stopping and turning frames: Stopping and turning frames of each joint j are estimated from the latest movements of the joint $\{\boldsymbol{b}_j(t-N+1), \cdots, \boldsymbol{b}_j(t)\}$, where N is the number of the latest frames considered.

Stopping frames are defined as frames at which the moving distance of the joint takes a local minimum. The moving distance $g_j(i)$ at frame i is given by

$$g_j(i) = ||\boldsymbol{b}_j(i+1) - \boldsymbol{b}_j(i)||. \tag{4}$$

A set of stopping frames $\mathcal{I}_j^{\mathrm{st}}$ is obtained as follows:

$$\mathcal{I}_j^{\mathrm{st}} = \left\{ \underset{i \leq m \leq i+n}{\operatorname{argmin}}\, g_j(m) \;\middle|\; t-N+1 \leq i < t-n \right\}, \tag{5}$$

where n is a shift length.

Turning frames, on the other hand, are defined as frames at which the inner product of the moving distances takes a local maximum. The inner product $h_j(i)$ is given by

$$h_j(i) = \boldsymbol{o}_{j,i}^T \boldsymbol{o}_{j,i+1}, \tag{6}$$

$$\boldsymbol{o}_{j,i} = \frac{\boldsymbol{b}_j(i+1) - \boldsymbol{b}_j(i)}{g_j(i)}. \tag{7}$$

A set of turning frames $\mathcal{I}_j^{\mathrm{tr}}$ is then obtained as follows:

$$\mathcal{I}_j^{\mathrm{tr}} = \left\{ \underset{i \leq m \leq i+n}{\operatorname{argmin}}\, h_j(m) \;\middle|\; t-N+1 \leq i < t-n \right\}. \tag{8}$$

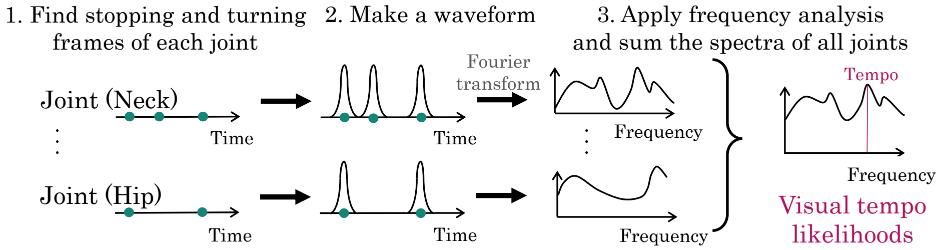


Fig. 3. Extraction of skeleton features: Visual tempo likelihoods is obtained by detecting characteristic points of all joint, and generating continuous signals, and performing frequency analysis.

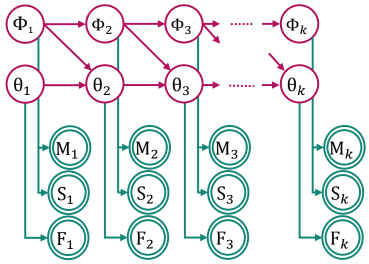


Fig. 4. The graphical representation for the proposed state-space model.

2) Signal generation from stopping and turning frames: Since $\mathcal{I}_j^{\mathrm{st}}$ and $\mathcal{I}_j^{\mathrm{tr}}$ are discrete sets of time points, it is difficult to directly analyze the periodicities of those sequences. To make periodicity analysis easy, we instead generate continuous signals by convoluting a Gaussian function with $\mathcal{I}_j^{\mathrm{st}}$ and $\mathcal{I}_j^{\mathrm{tr}}$. This enables us to use the Fourier transform.

More specifically, the two signals $y_j^{\mathrm{st}}(t)$ and $y_j^{\mathrm{tr}}(t)$ corresponding to $\mathcal{I}_j^{\mathrm{st}}$ and $\mathcal{I}_j^{\mathrm{tr}}$ are given by

$$y_j^{\mathrm{st}}(t) = \sum_{i \in \mathcal{I}_j^{\mathrm{st}}} \mathcal{N}(t|i, \sigma_y^2),\ y_j^{\mathrm{tr}}(t) = \sum_{i \in \mathcal{I}_j^{\mathrm{tr}}} \mathcal{N}(t|i, \sigma_y^2), \tag{9}$$

where $\mathcal{N}(x|\mu, \sigma)$ represents a Gaussian function where x is a variable and parameters μ and σ correspond to the mean and standard deviation.

3) Frequency analysis of generated signals: The spectra $\hat{y}_j^{\mathrm{st}}(f)$ and $\hat{y}_j^{\mathrm{tr}}(f)$ are obtained by applying the Fourier transform to the corresponding signals $y_j^{\mathrm{st}}(t)$ and $y_j^{\mathrm{tr}}(t)$. At each frame t, the visual tempo likelihoods $S(t, f)$ that indicates the likelihoods over possible tempos is calculated by accumulating the amplitude spectra of all joints as follows:

$$S(t, f) = \sum_{j=1}^{J} (|\hat{y}_j^{\mathrm{st}}(f)| + |\hat{y}_j^{\mathrm{tr}}(f)|). \tag{10}$$

The visual tempo likelihoods $S_k(u)$ of the current beat time θ_k are given by $S_k(u) = S(\theta_k, f_u)$, where f_u is a frequency corresponding to tempo u.

D. Unified state-space modeling for audio-visual integration

We formulate a state-space model that integrates acoustic and skeleton features (Fig. 4). A state vector $\boldsymbol{z}_k$ is given by using a tempo ϕ_k and a beat time θ_k as follows:

$$\boldsymbol{z}_k = [\phi_k, \theta_k]^T. \tag{11}$$

An observation vector $\boldsymbol{x}_k$, is given by using an audio tempo M_k, onset likelihoods F_k (acoustic features) and visual tempo likelihoods S_k (skeleton features) as follows:

$$\boldsymbol{x}_k = [M_k, S_k^T, F_k^T]^T \tag{12}$$

1) Observation model: We assume the audio tempo M_k to follow a Gaussian distribution. The visual tempo likelihoods S_k is considered as a probability distribution by being normalized. Similarly, the onset likelihoods F_k is considered as the probability distribution by being normalized. Consequently, the observation model is defined as:

$$p(\boldsymbol{x}_k|\boldsymbol{z}_k) = p(M_k|\boldsymbol{z}_k)p(S_k|\boldsymbol{z}_k)p(F_k|\boldsymbol{z}_k), \tag{13}$$

$$p(M_k|\boldsymbol{z}_k) \propto \mathcal{N}(M_k|\phi_k, \sigma_M^2) + \varepsilon, \tag{14}$$

$$p(S_k(u = \phi_k)|\boldsymbol{z}_k) \propto S_k(u = \phi_k), \tag{15}$$

$$p(F_k(t = \theta_k)|\boldsymbol{z}_k) \propto F_k(t = \theta_k), \tag{16}$$

where σ_M is the standard deviation of M_k and ε is a constant.

2) State transition model: We assume the state vector to follow a random walk as follows:

$$p(\boldsymbol{z}_k|\boldsymbol{z}_{k-1}) = \mathcal{N}(\boldsymbol{z}_k|[\phi_{k-1}, \theta_{k-1} + b/\phi_{k-1}]^T, \boldsymbol{Q}), \tag{17}$$

where $\boldsymbol{Q}$ is the covariance matrix of the process noise and b is a constant representing the ratio between the inverse of a tempo and the frame-shift interval.

E. Posterior estimation based on a particle filter

The tempo ϕ_k and the beat time θ_k are estimated by using a particle filter because the visual tempo likelihoods $S_k(u)$ and the onset likelihoods $F_k(t)$ are not Gaussian distributed. Here we use sequential importance resampling (SIR) [21] for efficient particle filtering. The posterior distribution of the state vector $p(\boldsymbol{z}_k|\boldsymbol{x}_{1:k})$ is approximated by L particles:

$$p(\boldsymbol{z}_k^{(l)}|\boldsymbol{x}_{1:k}) \approx w_k^{(l)}, \tag{18}$$

where $w_k^{(l)}$ is the weight of particle l $(1 \le l \le L)$.

This estimation consists of the following three stages: state transition, weight calculation, and state estimation. The proposal distribution is based on the state transition model. Here, L' particles selected randomly transit independently from the state transition model. It prevents significant concentrations of particles and enables adaptation to tempo changes. The proposal distribution is defined as

$$\boldsymbol{z}_k^{(l)} \sim q(\boldsymbol{z}_k|\boldsymbol{z}_{k-1}^{(l)}) \tag{19}$$

$$\propto \mathcal{N}\left(\boldsymbol{z}_k \middle| [\phi_{k-1}, \theta_{k-1} + b/\phi_{k-1}]^T, \boldsymbol{Q}\right) + \frac{L'}{L}. \tag{20}$$

The weight $w_k^{(l)}$ for each particle l is given by

$$w_k^{(l)} = w_{k-1}^{(l)} \frac{p(\boldsymbol{z}_k^{(l)}|\boldsymbol{z}_{k-1}^{(l)})p(\boldsymbol{x}_k|\boldsymbol{z}_k^{(l)})}{q(\boldsymbol{z}_k|\boldsymbol{z}_{k-1}^{(l)})}. \tag{21}$$

The observation and state transition probabilities are in Eqs. (13) and (17). The proposal distribution is in Eqs. (20).

The expected value of the the state vector $\overline{\boldsymbol{z}}_k = [\overline{\phi}_k, \overline{\theta}_k]^T$ is obtained by using the weights of particles as follows:

$$\overline{\phi}_k = \sum_{l=1}^{L} w_k^{(l)} \phi_k^{(l)},\ \overline{\theta}_k = \sum_{l=1}^{L} w_k^{(l)} \theta_k^{(l)}. \tag{22}$$

In resampling, the particles with large weights are replaced by many new similar particles, whereas those with small weights are discarded because they are unreliable.

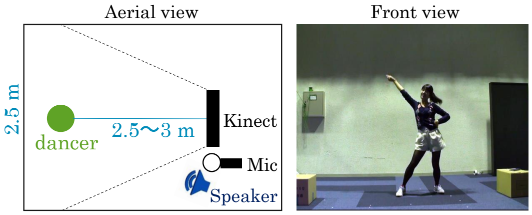


Fig. 5. Kinect data acquisition

TABLE I

COMPARED METHODS AND PARAMETER VALUES

Methods		Audio tempo (acoustic feature)	Visual tempo likelihoods (skeleton feature)	Onset likelihoods (acoustic feature)	σ_ϕ	σ_θ	σ_M
Proposed	(MotionCapture data)	✓	✓	✓	2.5	0.2	3.0
	(Kinect data)				1.0	0.1	2.0
Without visual tempo likelihoods		✓		✓	2.0	0.1	0.5
Without audio tempo			✓	✓	2.5	0.1	–

IV. EXPERIMENTAL EVALUATION

This section reports a comparative experiment conducted to evaluate the performance improvement of the proposed audio-visual beat-tracking method over mono-modal (audio-only or visual-only) methods [1].

A. Experimental conditions

The five sessions were obtained from the dance motion capture database released by the university of Cyprus (54 joints, about 30 frames per second (FPS)) [22]. In addition, we recorded the dancing movements of a female dancer by using a Kinect Xbox 360 depth sensor (15 joints, about 20 FPS). There were eight sessions of dances to popular music. The distance between the Kinect sensor and the dancer was about 2.5 meters. The whole body of the dancer was captured by the Kinect sensor (Fig. 5). Audio signals of dance music (noisy live recordings) were played back and captured by a microphone with a sampling rate of 16 kHz and a quantization of 16 bits. The experiment was conducted in a room with a reverberation time (RT_{60}) of 800 msec.

We compared the proposed method with the conventional audio beat-tracking method [4] and two methods that use a subset of the proposed features. Since our method uses three types of observations: an audio tempo, visual tempo likelihoods, and onset likelihoods, we compared with a method without the observation of the audio tempo M_k and a method without that of the visual tempo likelihoods $S_k(u)$ (Tab. I). We ran the audio beat-tracking algorithm implemented in the robot audition software called HARK [23]. The parameters were the default settings of HARK. With the frame rate of the data defined as t_{fps}, the parameters of our visual tempo estimation method were set as follows: $N = 20t_{\mathrm{fps}}, n = 60t_{\mathrm{fps}}/180$. The parameters of the particle filter were set as follows: $L = 1000, \varepsilon = 0.02$, and $b = 60$. The other parameters were determined experimentally (Tab. I), here $\boldsymbol{Q} = \begin{bmatrix} \sigma_\phi^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix}$.

The error tolerance between an estimated beat time and a ground-truth beat time was 100 msec, because we feel that the two sounds whose onset times differ by less than 100 msec are played at the same time [24]. Based on this, we calculated the precision ($r_p = N_e/N_d$), the recall ($r_r = N_e/N_c$) and the F-measure ($2r_pr_r/(r_p + r_r)$). Here, N_e, N_d, and N_c correspond to the numbers of correct estimates, whole estimates, and correct beats. We estimated thirty times for each data and evaluated the average of them because the estimation results depend on random initialization of a particle filter.

B. Experimental results

The experimental results showed that the proposed method always outperformed the audio beat-tracking method (Fig. 6). In addition, the proposed method was more accurate than the other methods on average. Consequently, the effectiveness of the proposed method was verified.

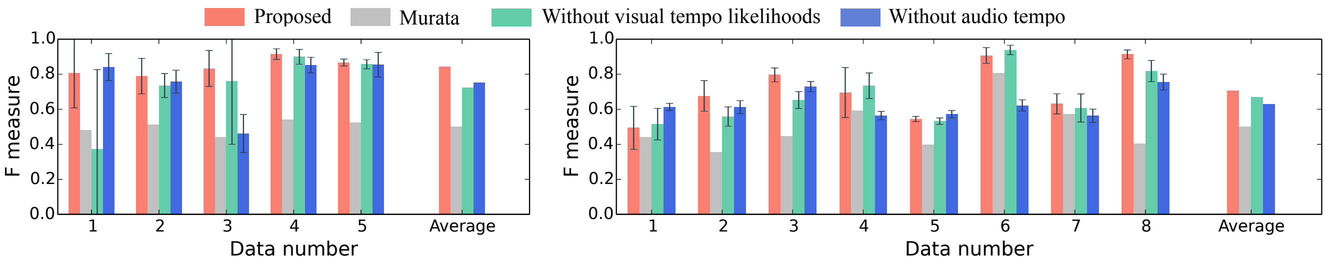
We discuss cases in which the results of the method without an audio tempo M_k of acoustic features had considerably lower scores than those of the method without visual tempo likelihoods $S_k(u)$ of skeleton features (Kinect data No. 4 and No. 6). In these cases, the results obtained by the proposed method had lower scores than the method without visual tempo likelihoods $S_k(u)$. The visual tempo estimation method failed in these cases because it was difficult to detect the stopping and turning frames of joints from dances in which the hands and feet moved very little. The average results for the Kinect data had considerably lower scores than those for the motion capture data. This is because the number of joints used for the Kinect data was less than that used for the motion capture data and because the Kinect data had a lot of noise.

Fig. 7 shows three examples of the experimental results. In Fig. 7-(a) and (b), the proposed method estimated a correct tempo using an audio tempo M_k and visual tempo likelihoods $S_k(u)$. In Fig. 7-(c), as was mentioned earlier, although a visual tempo estimation method failed, the proposed method estimated a correct tempo using acoustic features. To solve this problem, we will introduce a high-pass filter to reduce noise, and we have to integrate error handling for occlusions into our system.

V. CONCLUSIONS AND FUTURE WORK

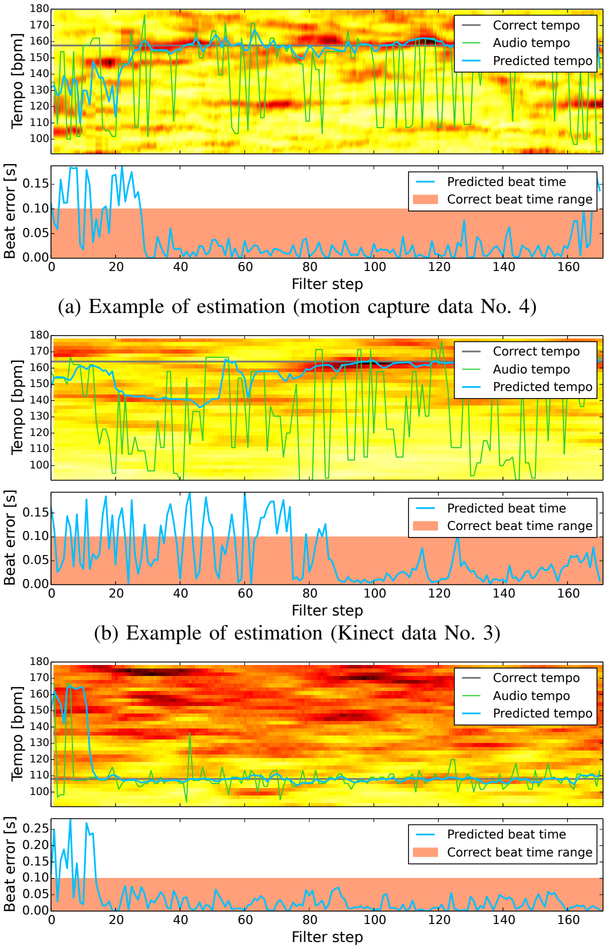
We developed an audio-visual beat-tracking method for an entertainment robot that can dance in synchronization with music and human dancers. The proposed method, which deals with both music audio signals and skeleton information of dancers, is designed to be robust to noise and reverberation. To extract acoustic features from music audio signals, we estimate an audio tempo and onset likelihoods at each frame. To extract skeleton features, on the other hand, we calculate visual tempo likelihoods. We then formulate a state-space model that consists of latent variables (tempo and beat times) and observed variables (acoustic and skeleton features). The posterior distribution of the latent variables is estimated by using a particle filter.

[1] Although a robot danced with estimated beats in the demo video, this experiment evaluated the proposed method without any enbodied robots.



(a) Motion capture data (b) Kinect data

Fig. 6. Experimental results.



(a) Example of estimation (motion capture data No. 4)

(b) Example of estimation (Kinect data No. 3)

(c) Example of estimation (Kinect data No. 6)

Fig. 7. Examples of beat estimation (blue: estimation results). The figures above show the results of tempo estimation (green: audio tempo, gray: ground truth, color depth: visual tempo likelihood). The figures below show the errors of beat time estimation.

We plan to improve the accuracy of visual tempo estimation for the Kinect data. Furthermore, when microphones attached to a robot are used, the music audio signals contain self-generated noise. Semi-blind independent component analysis [25] is a promising solution to cancel the noise and it was actually used by Murata *et al.* [4]. To develop a robot that can dance with humans, we plan to conduct subjective experiments using a real dancing robot.

ACKNOWLEDGMENTS This study was supported in part by KAKENHI 24220006, 26700020, 24700168, and 26280089 and the OngaCREST project.

References

[1] Y. Kusuda, "Toyota's Violin-playing Robot," *Ind. Robot*, vol. 35, no. 6, pp. 504–506, 2008.

[2] Murata Manufacturing Co., Ltd, "Cheerleaders Debut," http://www.murata.co.jp/cheerleaders/, 2015.

[3] K. Petersen *et al.*, "Development of a Aural Real-Time Rhythmical and Harmonic Tracking to Enable the Musical Interaction with the Waseda Flutist Robot," *IROS*, 2009.

[4] K. Murata *et al.*, "A Beat-Tracking Robot for Human-Robot Interaction and Its Evaluation," *Humanoids*, 2008.

[5] K. Kosuge *et al.*, "Partner Ballroom Dance Robot-PBDR-," *SICE Journal of Control, Measurement, ans System Integration*, vol. 1, no. 1, pp. 74–80, 2008.

[6] K. Kaneko *et al.*, "Cybernetic Human HRP-4C," *Humanoids*, 2009.

[7] W. Chu *et al.*, "Rhythm of Motion Extraction and Rhythm-Based Cross-Media Alignment for Dance Videos," *ACM Multimedia*, 2012.

[8] T. Shiratori *et al.*, "Rhythmic Motion Analysis using Motion Capture and Musical Information," *MFI*, 2003.

[9] T. Itohara *et al.*, "Particle-filter Based Audio-visual Beat-tracking for Music Robot Ensemble with Human Guitarist," *IROS*, 2011.

[10] D. R. Berman, "Avisarme: Audio visual synchronization algorithm for a robotic musician ensemble," Master's thesis, Maryland, 2012.

[11] G. Weinberg *et al.*, "Interactive Jamming with Shimon: A Social Robotic Musician," *Human robot interaction*, 2009.

[12] K. Petersen *et al.*, "Development of a real-time instrument tracking system for enabling the musical interaction with the Waseda Flutist Robot," *IROS*, 2008.

[13] A. Lim *et al.*, "Robot Musical Accompaniment: Integrating Audio and Visual Cues for Real-time Synchronization with a Human Flutist," *IROS*, 2010.

[14] S. Dixon, "Evaluation of the Audio Beat Tracking System BeatRoot," *J.New Music Res.*, vol. 36, no. 1, pp. 39–50, 2007.

[15] M. Goto, "An Audio-based Real-time Beat Tracking System for Music With or Without Drum-sounds," *J.New Music Res.*, vol. 30, no. 2, pp. 159–171, 2001.

[16] A. M. Stark *et al.*, "Real-time beat-synchronous analysis of musical audio," *DAFx*, 2009.

[17] D. Ellis *et al.*, "Beat Tracking by Dynamic Programming," *J.New Music Res.*, vol. 1, pp. 51–60, 2007.

[18] M. Davies *et al.*, "Context-Dependent Beat Tracking of Musical Audio," *IEEE Trans. on Audio, Speech, and Lang. Process.*, vol. 15, no. 3, pp. 1009–1020, 2007.

[19] J. L. Oliveira *et al.*, "Live Assessment of Beat Tracking for Robot Audition," *IROS*, 2012.

[20] C. Guedes *et al.*, "Extracting Musically-Relevant Rhythmic Information from Dance Movemen by Applying Pitch-Tracking Techniques to a Video Signal," *SMC*, 2006.

[21] M. Sanjeev *et al.*, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *Signal Processing*, 2002.

[22] the University of Cyprus, "Dance Motion Capture Database," http://dancedb.cs.ucy.ac.cy/, 2014.

[23] K. Nakadai *et al.*, "Design and Implementation of Robot Audition System'HARK'-Open Source Software for Listening to Three Simultaneous Speakers," *Advanced Robotics*, 2010.

[24] R. A. Rasch, "Synchronization in Performed Ensemble Music," *J Acta Acustica united with Acustica*, 1979.

[25] R. Takeda *et al.*, "Exploiting Known Sound Source Signals to Improve ICA-based Robot Audition in Speech Separation and Recognition," *IROS*, 2007.