Paper:

Audio-Visual Beat Tracking Based on a State-Space Model for a Robot Dancer Performing with a Human Dancer

Misato Ohkita, Yoshiaki Bando, Eita Nakamura,
Katsutoshi Itoyama, and Kazuyoshi Yoshii

Graduate School of Informatics, Kyoto University
Yoshida-honmachi, Sakyo-ku, Kyoto 606-8501, Japan
E-mail: {ohkita, bando, enakamura, itoyama, yoshii}@sap.ist.i.kyoto-u.ac.jp

[Received August 5, 2016; accepted November 30, 2016]

This paper presents a real-time beat-tracking method that integrates audio and visual information in a probabilistic manner to enable a humanoid robot to dance in synchronization with music and human dancers. Most conventional music robots have focused on either music audio signals or movements of human dancers to detect and predict beat times in real time. Since a robot needs to record music audio signals with its own microphones, however, the signals are severely contaminated with loud environmental noise. To solve this problem, we propose a state-space model that encodes a pair of a tempo and a beat time in a state-space and represents how acoustic and visual features are generated from a given state. The acoustic features consist of tempo likelihoods and onset likelihoods obtained from music audio signals and the visual features are tempo likelihoods obtained from dance movements. The current tempo and the next beat time are estimated in an online manner from a history of observed features by using a particle filter. Experimental results show that the proposed multi-modal method using a depth sensor (Kinect) to extract skeleton features outperformed conventional mono-modal methods in terms of beat-tracking accuracy in a noisy and reverberant environment.

Keywords: robot dancer, real-time beat tracking, state-space model, audio-visual integration

1. Introduction

Intelligent entertainment robots that can adaptively interact with humans have actively been developed in the field of robotics. While one of the typical goals of robotics is to develop task-oriented industrial robots that can accurately perform routines, entertainment robots are assumed be used by people in their daily lives. To recognize dynamically-varying environments in real time, those robots should have both visual and auditory sensors, as humans do. The research topic of *robot audition* has thus gained a lot of attention [1, 2] for the detection, localization, separation, and recognition of various of sounds to help in computer vision and speech recognition.

Some entertainment robots are designed to interact with humans through music. Among them are a violin-playing robot that can play the violin according to a predefined sequence of movements [3], a cheerleader robot that can balance on a ball [a], and a flute-playing robot that can play the flute in synchronization with a melody played by a human being [4]. In this paper we aim to develop a music robot that can dance interactively with people using both auditory and visual sensors (microphones and depth sensors).

A robot dancer that performs synchronously with human dancers needs to adaptively and autonomously control its movements while recognizing both music and the movements of the people in real time. Murata et al. [5], for example, enabled a bipedal humanoid to step and sing in synchronization with musical beats. Kosuge et al. [6] devised a dancing robot that can predict the next step intended by a dance partner and move according to his or her movements. Nakaoka et al. [7] developed a humanoid that can generate natural dance movements by using a complicated human-like dynamic system.

The main technical challenge in synchronizing the dance movements of a robot with musical beats is to perform real-time beat tracking, i.e., estimate a musical tempo and detect beat times (temporal positions in which people are likely to clap their hands), in a noisy and reverberant environment. However, very few beat-tracking methods assume that they are used in an online manner and that music audio signals are contaminated. Murata et al. [5], for example, proposed an *online audio* beat-tracking method that can quickly follow tempo changes and is robust to environmental noise, but this method often fails for music that has many accented up-beats. Chu and Tsai [8] proposed an *offline visual* beat-tracking method that tries to detect tempos (periods) from dance movements, but this method often fails for real musical pieces with complicated dance movements that include irregular patterns. This means that the accuracy of beat tracking using a single modality is limited.

In this paper we propose a multi-modal beat-tracking method that analyzes both music audio signals recorded by a microphone and dance movements observed as a

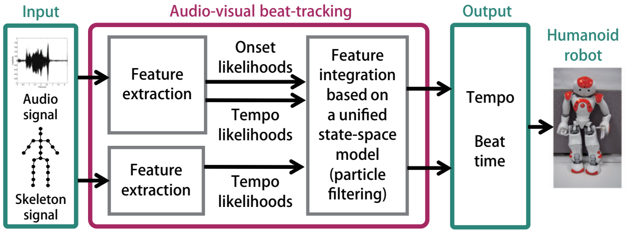


Fig. 1. An overview of real-time audio-visual beat tracking for music audio signals and human dance moves.

sequence of joint positions by a depth sensor (e.g., Microsoft Kinect) or a motion capture system (**Fig. 1**). Such audio-visual integration has often been studied in the music information retrieval (MIR) community, and it has been shown to achieve better performance than single-modal methods [9–14]. The proposed method is an improved version of our previous method [15]. To effectively integrate audio-visual information, it is necessary to extract *intermediate* features that represent the likelihood of a tempo and that of a beat time. Such integration has been known to be effective in the context of audio-visual speaker tracking [16]. In each frame, we estimate the likelihood of each tempo and the onset likelihood of the current frame from music audio signals. This method is more advantageous than the previous method [15], which directly and uniquely estimates an audio tempo without allowing for other possibilities. On the other hand, another likelihood of each tempo is also calculated from skeleton information. 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.

The remainder of this paper is organized as follows: Section 2 introduces related work on audio, visual, or audio-visual beat tracking methods. Section 3 explains the proposed method and Section 4 reports experimental results on beat tracking for two types of datasets. Section 5 describes the implementation of a robot dancer based on real-time beat tracking and Section 6 summarizes our results.

2. Related Work

This section describes the related work on beat tracking using audio and/or visual signals.

2.1. Beat Tracking for Music Audio Signals

Beat tracking for music audio signals has been studied extensively. Dixon et al. [17], 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. [18] proposed a similar online method using both IOIs and chord changes as useful clues for detecting beat times. Stark et al. [19] proposed an online method that combines a beat-tracking method based on dynamic programming [20] with another method using a state-space model for tempo estimation [21]. The performance of this method was shown to equal 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. [5] proposed a real-time method that enables a robot to step and sing to musical beats while recording music audio signals with an embedded microphone. This method calculates an onset spectrum at each frame and detects beat times by calculating the auto-correlation of onset spectra. Oliveira et al. [22] proposed an online multi-agent method using several kinds of multi-channel preprocessing (e.g., sound source localization and separation) to improve robustness against environmental noise.

Neural networks have recently gained a lot of attention for significantly improving the accuracy of beat tracking [23]. Böck et al. [24] and Krebs et al. [25], for example, used recurrent neural networks (RNNs) to model the periodic dynamics of beat times. Durand and Essid [26] proposed a method that uses acoustic features obtained by deep neural networks to train conditional random fields. However, the online application of these methods has scarcely been discussed.

2.2. Beat Tracking for Dance Movements

Several studies have been conducted to analyze the rhythms of dance movements. Guedes et al. [27] proposed a method that estimates an audio tempo of dance movements in a dance movie. This method can be used to estimate a tempo from periodic movements, e.g., periodically putting a hand up and down, provided that other moving objects do not exist in a dance movie. It is difficult to use this method with the complicated movements seen in real dance performances. Chu and Tsai [8] proposed an offline method that extracts the motion trajectories of a dancer's body from a dance movie and then detects time frames in which a characteristic point stops or rotates. They proposed a system that uses this method to replace the background music of a dance video.

2.3. 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 indicate a tempo. Weinberg et al. [12] developed an interactive marimba-playing robot called Shimon that performs beat tracking while recognizing the visual cue of a head nodding to the beat. Petersen et al. [13] proposed a method that uses the visual cue of a waving hand to control the parameters of vibrato or tempo. Lim et al. [14] 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 (the up and down movement of the flute to the tempo) and listening to the notes from the flute.

The other approach does not use predefined visual cues. Itohara et al. [10] proposed an audio-visual beat-tracking method using both guitar sounds and the guitarist's arm motions. They formulated a simplified model that represents a guitarist's arm trajectory as a sine wave and integrates acoustic and skeleton features by using a state-space model. Berman et al. [11] proposed a beat-tracking method for ensemble robots playing with a human guitarist. To visually estimate a tempo, a method similar to that in [27] was used. This method can estimate the tempo from a periodic behavior, such as a head and foot moving up and down to the music in playing a guitar.

3. Proposed Method

This section describes the proposed method of audio-visual beat tracking that jointly deals with both music audio signals and skeleton information of dance movements (**Fig. 1**). To effectively integrate acoustic and skeleton information so that they can serve as complementary sources of information to improve beat tracking, we extract *intermediate* information as acoustic and skeleton features that indicate the likelihoods of tempos and beat times. In this stage, the method does not uniquely determine the current tempo and the next beat time. Instead, the method keeps all the possibilities of tempos and beat times. If a unique tempo were extracted from music audio signals as in [15], tempo estimation failure would severely degrade the overall performance. We therefore formulate a nonlinear state-space model that has a tempo and a beat time as latent variables and acoustic and skeleton features as observed variables. The current tempo and the next beat time are updated at each beat time in an online manner by using a particle filter and referring to the history of observed and latent variables.

We specify the problem of audio-visual beat tracking in Section 3.1. We explain how to extract acoustic and skeleton features from music audio signals and dance movements in Sections 3.2 and 3.3, describe the state-space model integrating these features in Section 3.4, and provide an inference algorithm in Section 3.5.

3.1. Problem Specification

Our goal is to estimate incrementally, at each beat time k, the current tempo ϕ_k and the next beat time θ_{k+1} by using the history of acoustic features $\{A_1,\ldots,A_k\}$ and that of skeleton features $\{S_1,\ldots,S_k\}$:

Input:	history of acoustic features: $\{A_1,A_2,\ldots,A_k\}$
	history of skeleton features: $\{S_1,S_2,\ldots,S_k\}$
Output:	current tempo: ϕ_k
	next beat time: θ_{k+1}

where the tempo is defined in beats per minute (BPM). This estimation step is iteratively executed when the current time, denoted by t, exceeds the predicted next beat time ($t = \theta_{k+1}$).

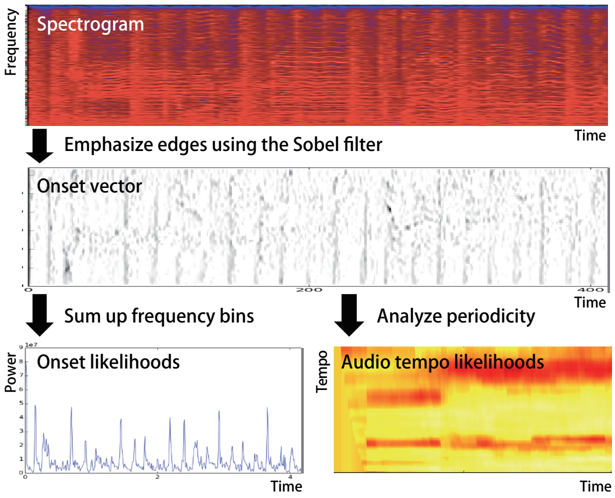


Fig. 2. Acoustic features consisting of an onset likelihood and audio tempo likelihoods are extracted at each frame.

3.2. Extraction of Acoustic Features

The acoustic feature A_k at the current beat time θ_k consists of frame-based onset likelihoods $\{F_k(t)|\theta_{k-1} < t \leq \theta_k + \varepsilon_f\}$ and audio tempo likelihoods $\{R_k(u)\}$ over possible tempo u at the current beat time θ_k. Here, t is a frame index (the frame-shift interval is 10 ms in our study), u is a tempo parameter, and ε_f is a few frames. Our requirement for these features is that they be robust against environmental noise and quick tempo change since audio signals involve various kinds of loud noises, including the sounds of footsteps and the voices of the audience. In the following we describe a method for obtaining these likelihoods based on an audio beat-tracking method in [5].

3.2.1. Onset Likelihoods

The onset likelihood $F_k(t)$ in 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 $y(t)$ to obtain frequency spectra. The Hanning window is used as a window function. The obtained spectra are sent to a mel-scale filter bank, which changes the linear frequency scale to the mel-scale frequency scale, to reduce the computational cost. Let $mel(t,f)$ be a mel-scale spectrum, where f $(1 \leq f \leq F_\omega)$ represents a mel-scale frequency.

A Sobel filter is then used to detect frequency bins with rapid power increase from the spectra $mel(t,f)$. Since the Sobel filter has been commonly used for extracting edges from images, it can be applied to a music spectrogram by regarding it as an 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

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

3.2.2. Audio Tempo Likelihoods

The audio tempo likelihood $R_k(u)$ indicates a distribution of instantaneous tempo u at the current beat time θ_k. Murata et al. [5] proposed a method of estimating the most likely instantaneous tempo by calculating the autocorrelation of the onset vector and extracting its peaks. To obtain the likelihood of tempo instead of the most likely value, we extend this method as follows:

Let us first define the normalized cross-correlation (NCC) of the onset vector as follows:

$$R(t,s) = \frac{\sum_{j=1}^{F_\omega}\sum_{i=0}^{P_\omega-1} d(t-i,j)d(t-s-i,j)}{\sqrt{\sum_{j=1}^{F_\omega}\sum_{i=0}^{P_\omega-1} d(t-i,j)^2 \sum_{j=1}^{F_\omega}\sum_{i=0}^{P_\omega-1} d(t-s-i,j)^2}}, \quad \ldots\ldots (2)$$

where s is a shift parameter and P_ω is a window length. The NCC has the property of being able to be calculated with a shorter window length than the conventional autocorrelation. For real-time processing, we used the fast NCC, a computationally efficient algorithm, to calculate the NCC. $R(t,s)$ tends to take larger values when s is close to the time interval of a beat.

If $R(t,s)$ is used, the audio tempo likelihood $R_k(u)$ for possible tempo u is given by this equation:

$$R_k(u) = \exp(R(\theta_k, s_u)), \quad \ldots\ldots (3)$$

where $s_u = (60/u)$ is a time shift corresponding to tempo u. Because $R(t,s)$ can have negative values with the fast NCC, we take the exponentials. In order to avoid the problem of double/halved tempos, the tempo value is restricted to the range from m BPM to $2m$ BPM, as in [5].

3.3. Extraction of Skeleton Features

The skeleton feature S_k of the current beat time θ_k is a vector of visual tempo likelihoods $\{S_k(u)\}$ over possible tempo u. To extract this feature, we use an online version of a visual tempo estimation method proposed by Chu and Tsai [8]. Although the original method is assumed to analyze the movements of characteristic points detected from a dance movie, we develop a method that can deal with the movements of the joints of a human dancer. Let $\{\boldsymbol{b}_1(t),\ldots,\boldsymbol{b}_J(t)\}$ be a set of the 3D coordinates of joints, e.g., neck and hip, where J is the number of joints ($\boldsymbol{b}_j(t) \in \mathbb{R}^3$). The value of J depends on the device, e.g., Kinect or a motion capture system, used to analyze the movements of a human dancer.

The skeleton information $\{\boldsymbol{b}_1(t),\ldots,\boldsymbol{b}_J(t)\}$ is obtained by following these three steps (**Fig. 3**). First, we detect time frames in which some joints stop and turn (*stopping frames* and *turning frames*). This step is considered to be important because dancers tend to stop or turn their joints at beat times. Second, we make a continuous signal from a discrete set of the detected stopping and turning 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.

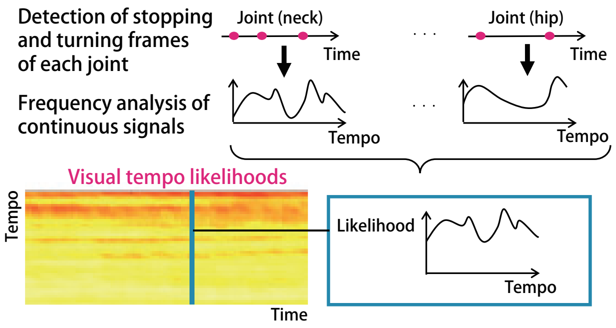

Detection of stopping and turning frames of each joint
Joint (neck)
Time
Joint (hip)
Time
Frequency analysis of continuous signals
Tempo
Tempo
Visual tempo likelihoods
Tempo
Time
Likelihood
Tempo

Fig. 3. Skeleton features (i.e., visual tempo likelihoods) are extracted in each frame by detecting the characteristic points of all joints.

3.3.1. Detection of Stopping and Turning Frames

Stopping and turning frames of each joint j are detected using the latest movements of the joint $\{\boldsymbol{b}_j(t-N+1),\ldots,\boldsymbol{b}_j(t)\}$, where N is the number of frames considered. The moving distance $g_j(i)$ at frame i is given by

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

Stopping frames are defined as frames in which the distance the joint moves takes a local minimum. A set of stopping frames $\mathscr{I}_j^{\mathrm{st}}$ is obtained as follows:

$$\mathscr{I}_j^{\mathrm{st}} = \left\{ \operatorname*{argmin}_{i \le m \le i+n} g_j(m) \,\middle|\, t-N+1 \le i < t-n \right\}, \quad (5)$$

where n is a shift length.

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

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

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

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

$$\mathscr{I}_j^{\mathrm{tr}} = \left\{ \operatorname*{argmin}_{i \le m \le i+n} h_j(m) \,\middle|\, t-N+1 \le i < t-n \right\}, \quad (8)$$

where n is a shift length.

3.3.2. Frequency Analysis of Continuous Signals Converted from Stopping and Turning Frames

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

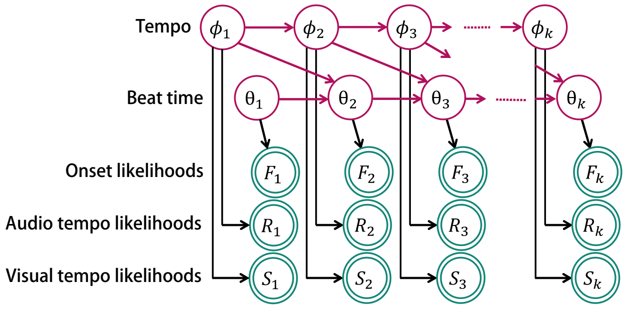


Fig. 4. The graphical representation of the proposed state-space model that represents how acoustic features F_k and R_k and skeleton features S_k are stochastically generated from a beat time θ_k with a tempo ϕ_k.

given by

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

where $\mathscr{N}(x|\mu,\sigma^2)$ represents a Gaussian function with mean μ and standard deviation σ. This enables us to use the Fourier transform.

Let $\hat{y}_j^{\mathrm{st}}(f)$ and $\hat{y}_j^{\mathrm{tr}}(f)$ be the Fourier transform of $y_j^{\mathrm{st}}(t)$ and $y_j^{\mathrm{tr}}(t)$. In each frame t, the visual tempo likelihood $S(t,f)$ that indicates the likelihood over possible tempos is calculated by accumulating the amplitude spectra of all joints as follows:

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

The visual tempo likelihood $S_k(u)$ of the current beat time θ_k is given by $S_k(u) = S(\theta_k, f_u)$, where $f_u = 2\pi u/60$ (1/s) is a frequency corresponding to tempo u.

3.4. State-Space Modeling for Feature Integration

We formulate a state-space model that integrates the acoustic and skeleton features (**Fig. 4**). A state vector $\boldsymbol{z}_k$ is defined as a pair made up of the tempo ϕ_k and the beat time θ_k:

$$\boldsymbol{z}_k = [\phi_k, \theta_k]^T. \quad \dots\dots\dots\dots (11)$$

An observation vector $\boldsymbol{x}_k$, is constructed from the audio tempo likelihood $R_k(u)$, the onset likelihood $F_k(t)$ (acoustic features) and the visual tempo likelihood $S_k(u)$ (skeleton features) as follows:

$$\boldsymbol{x}_k = [F_k^T, R_k^T, S_k^T]^T. \quad \dots\dots\dots\dots (12)$$

We then explain the two key components of the proposed state-space model: an observation model $p(\boldsymbol{x}_k|\boldsymbol{z}_k)$ and a state transition model $p(\boldsymbol{z}_{k+1}|\boldsymbol{z}_k)$.

3.4.1. Observation Model

We assume the components of an observation vector to follow independent distributions. Each distribution is assumed to be proportional to the likelihood function. Consequently, the observation model is defined as follows:

$$p(\boldsymbol{x}_k|\boldsymbol{z}_k) = p(F_k|\boldsymbol{z}_k)p(R_k|\boldsymbol{z}_k)p(S_k|\boldsymbol{z}_k), \quad \dots (13)$$

$$p(F_k|\boldsymbol{z}_k) \propto F_k(t=\theta_k), \quad \dots\dots\dots (14)$$

$$p(R_k|\boldsymbol{z}_k) \propto R_k(u=\phi_k), \quad \dots\dots\dots (15)$$

$$p(S_k|\boldsymbol{z}_k) \propto S_k(u=\phi_k) + \varepsilon, \quad \dots\dots\dots (16)$$

where a small constant ε governs the smoothness of the distribution.

3.4.2. State Transition Model

Music performance and dancing inevitably have timing fluctuations due to tempo variations and the noise of human movements. The current beat time, the next beat time, and the tempo are expected to meet $\theta_{k+1} = \theta_k + 60/\phi_k$ in theory. By modeling the tempo variations and the noise with Gaussians, the state transition probability is given as follows:

$$\begin{aligned} p(\boldsymbol{z}_{k+1}|\boldsymbol{z}_k) &\propto \mathscr{N}(\phi_{k+1}|\phi_k,\sigma_\phi^2)\mathscr{N}\left(\theta_{k+1}|\theta_k + \frac{60}{\phi_k}, \sigma_\theta^2\right) \\ &= \mathscr{N}\left(\boldsymbol{z}_{k+1}\middle|\left[\phi_k, \theta_k + \frac{60}{\phi_k}\right]^T, \boldsymbol{Q}\right), \quad \dots (17) \end{aligned}$$

where σ_ϕ and σ_θ are standard deviations of tempo variation and the noise of human movements, and $\boldsymbol{Q} = \mathrm{diag}[\sigma_\phi^2, \sigma_\theta^2]$ is a covariance matrix.

3.5. 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 likelihood $S_k(u)$ and the onset likelihood $F_k(t)$ are not Gaussian distributed and ϕ_k and θ_k should be updated in an online manner. Here we use sequential importance resampling (SIR) [28] for efficient particle filtering. The posterior distribution of the state vector $p(\boldsymbol{z}_k|\boldsymbol{x}_{1:k})$ is approximated as a distribution of L particles:

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

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

$$\begin{aligned} \boldsymbol{z}_k^{(l)} &\sim q(\boldsymbol{z}_k|\boldsymbol{z}_{k-1}^{(l)}) \\ &\propto \mathscr{N}\left(\boldsymbol{z}_k\middle|\left[\phi_{k-1}, \theta_{k-1} + \frac{b}{\phi_{k-1}}\right]^T, \boldsymbol{Q}\right) + \frac{L'}{L}. \quad (19) \end{aligned}$$

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

$$w_k^{(l)} = w_{k-1}^{(l)} \frac{p(\boldsymbol{z}_k^{(l)}|\boldsymbol{z}_{k-1}^{(l)})p(\boldsymbol{x}_k|\boldsymbol{z}_k^{(l)})}{q(\boldsymbol{z}_k|\boldsymbol{z}_{k-1}^{(l)})}. \quad \dots (20)$$

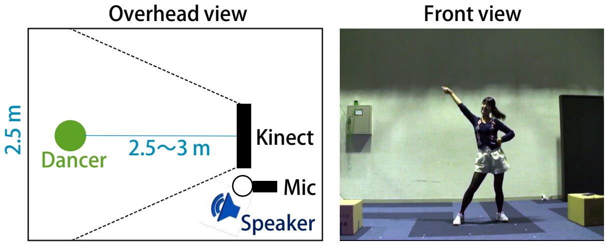


Fig. 5. Analysis of dance movements using Kinect.

The observation and state transition probabilities are given by Eqs. (13) and (17). The proposal distribution is given by Eq. (19).

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

$$\overline{\phi}_k = \sum_{l=1}^{L} w_k^{(l)} \phi_k^{(l)}, \qquad (21)$$

$$\overline{\theta}_k = \sum_{l=1}^{L} w_k^{(l)} \theta_k^{(l)}. \qquad (22)$$

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.

4. Evaluation

This section reports on experiments conducted to evaluate the performance improvement of the audio-visual beat-tracking method over mono-modal methods that use either audio tempo likelihoods or visual tempo likelihoods. Note that onset likelihoods obtained from music audio signals are always required for beat times to be estimated; they cannot be estimated if only skeleton features (visual tempo likelihoods) are used.

4.1. Experimental Conditions

The five sessions were obtained from a dance motion capture database released by the University of Cyprus ($J = 54$ joints, about 30 frames per second (FPS)) [b]. In addition, using a Kinect Xbox 360 depth sensor ($J = 15$ joints, about 20 FPS), we recorded the dance movements of a female dancer. 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 audio-visual beat-tracking method with two conventional audio beat-tracking methods [5, 15]. The method [5] is implemented in HARK [29] robot audition software, and its parameters are set to the default values except for $m = 90$. The method [15] is similar to our method except that an audio tempo is uniquely determined in each frame as an acoustic feature. To evaluate the effectiveness of integrating the three kinds of features: onset likelihoods F_k, audio tempo likelihoods R_k (acoustic features), and visual tempo likelihoods S_k (visual features), we tested an audio-based method using only F_k and R_k as well as a visual-based method using only F_k and S_k (**Table 1**). Given a frame rate t_{fps} of the skeleton data, the parameters of visual feature extraction were set as follows: $N = 20t_{\mathrm{fps}}$, $n = 60t_{\mathrm{fps}}/180$. The parameters of the particle filter were set as follows: $L = 1000$, $\varepsilon = \{0.0, 0.02\}$, and $b = 60$. σ_ϕ and σ_θ were experimentally chosen from $\{1.0, 3.0, 5.0\}$ and $\{0.01, 0.02, 0.03, 0.04\}$, respectively, for each method such that the average performance over all sessions was maximized. σ_M of the conventional method [15] was experimentally chosen from $\{0.25, 4.0, 9.0\}$ such that the average performance was maximized. Note that $\boldsymbol{Q} = \mathrm{diag}[\sigma_\phi^2, \sigma_\theta^2]$. All the methods were implemented as single-threaded codes and executed in an online manner on a standard desktop computer with Intel Core i7-4790 (3.6 GHz).

The error tolerance between an estimated beat time and a ground-truth beat time was 100 msec, because we consider two sounds with onset times that differ by less than 100 msec to be played at the same time [30]. We calculated the precision rate ($r_p = N_e/N_d$), recall rate ($r_r = N_e/N_c$), and F-measure ($2r_p r_r/(r_p + r_r)$), where N_e, N_d, and N_c correspond to the numbers of correct estimates, total estimates, and correct beats. Each method was executed thirty times for each dataset and the average performance over the thirty trials was calculated because the results depend on random initialization of a particle filter.

4.2. Experimental Results

The experimental results in **Fig. 6** show that the average F-measures (88.2% and 82.0%) obtained by the proposed model ($\varepsilon = 0.2$) were significantly better than those obtained by the other methods for both the motion capture data and Kinect data. The average F-measures obtained by the audio-based method were 85.9% and 79.0% and those obtained by the visual-based method were 84.1% and 70.5%. This indicates that the proposed method of integrating acoustic and visual features indeed serves to improve the beat-tracking performance and the use of audio tempo likelihoods brings improvements (85.7% and 72.5%) to our previous method that extracts a unique audio tempo before probabilistic integration [15]. The average F-measures for the Kinect data were considerably lower than those for the motion capture data. This is because the number of joints used for the Kinect data was lower than that used for the motion capture data and because the Kinect data had a lot of noise and fluctuations.

For the proposed model, the F-measure for $\varepsilon = 0.2$ was larger than that for $\varepsilon = 0$ in all cases. In particular, let us discuss cases in which the F-measure for the visual-based method was considerably worse than that for the audio-based method, e.g., Kinect data Nos.1, 4, and 6.

Table 1. Compared methods and parameter values.

Methods	Onset likelihoods (acoustic feature)	Audio tempo likelihoods (acoustic feature)	Visual tempo likelihoods (skeleton feature)
Proposed	✓	✓	✓
Audio-based	✓	✓	
Visual-based	✓		✓

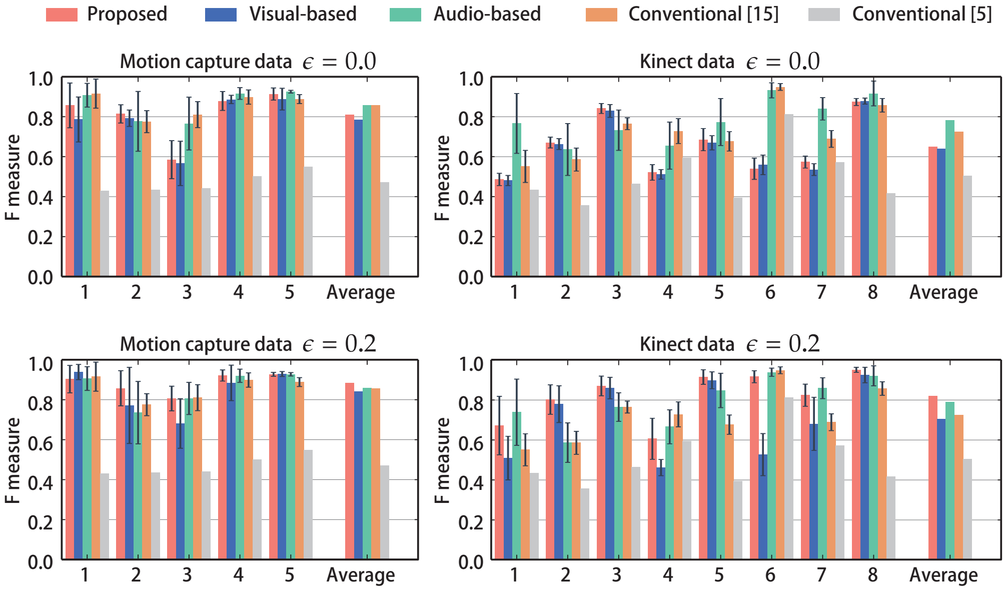


Fig. 6. Experimental results for two datasets with $\varepsilon = 0$ and $\varepsilon = 0.2$.

The visual-based 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. In these cases, we see that whereas the proposed method with $\varepsilon = 0$ had F-measures close to those for the visual-based method, the case with $\varepsilon = 0.2$ had F-measures closer to those for the audio-based method. This is probably because the smoothing by nonzero ε can avoid excessive concentration of particles when the visual likelihoods are unreliable and thus the complementary information of acoustic features can be more effectively used. This confirms that it is effective to smooth visual likelihoods for integration with acoustic features in the state-space model.

Figure 7 shows four examples of the experimental results. In **Figs. 7(a)** and **(c)**, both the visual and audio likelihoods had peaks near the ground-truth tempos, and we see that the estimated tempo gradually converged to the ground-truth tempo in real-time beat tracking. On the other hand, **Figs. 7(b)** and **(d)** show cases in which the visual likelihoods were unreliable. Such cases may happen when there are occlusions due to frequent rotations of the body or the dance motion involves only small movements of the hands and feet. Even in such situations, the estimated tempo gradually converged to the correct one in both examples. The convergence time is much faster in **Fig. 7(d)** than in **Fig. 7(b)** since the audio tempo likelihoods had more peaks near the true tempo values.

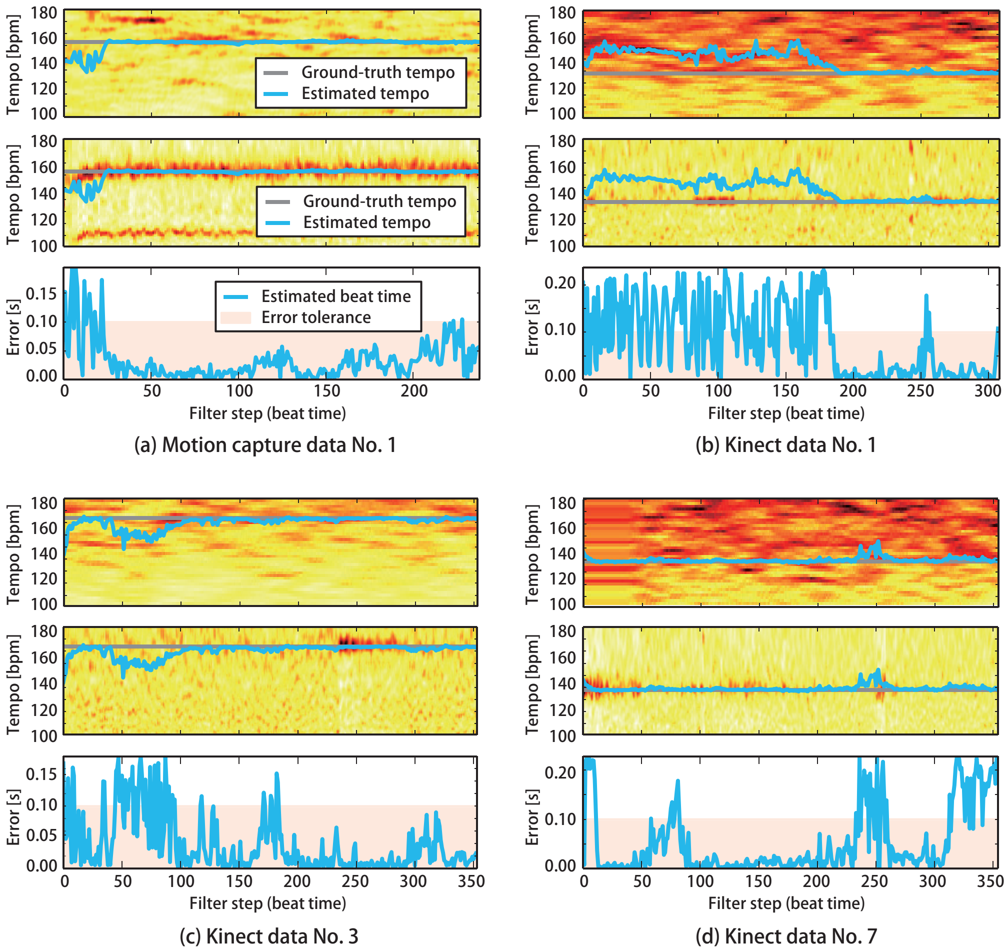
4.3. Evaluation on Noise Robustness

To evaluate the effectiveness of audio-visual integration in terms of noise robustness, we conducted an additional experiment using noise-contaminated audio signals. In this comparative experiment, crowd noise was added to each song of the dance motion capture database [b] with a different signal-to-noise (SNR) ratio of 20, 10, 0, −20, or −10 dB. The proposed method of audio-visual integration was compared with the audio-based and visual-based methods (see **Table 1**).

As shown in **Fig. 8**, the proposed method attained the best performances in almost all SNR conditions except for the SNR of −10 dB. In the SNRs of 20 and 10 dB, the audio-based method worked better than the visual-based method. In the SNRs of 0, −10, and −20 dB, on the other hand, in which audio signals were severely contaminated, the visual-based method worked slightly better than the audio-based method did, in which the proposed method was better than or comparable to the visual-based method. A reason why the performance was significantly degraded in a low SNR condition is that the proposed and visual-based methods need to use onset likelihoods obtained from audio signals to determine beat times because only tempos can be estimated from visual data.

4.4. Discussion

To realize a humanoid robot that can adaptively and autonomously dance like humans, it will be necessary to



(a) Motion capture data No. 1

(b) Kinect data No. 1

(c) Kinect data No. 3

(d) Kinect data No. 7

Fig. 7. Examples of audio-visual beat tracking for four musical pieces. The boxes show, from top to bottom, the visual tempo likelihoods, audio tempo likelihoods, and estimation errors of beat times.

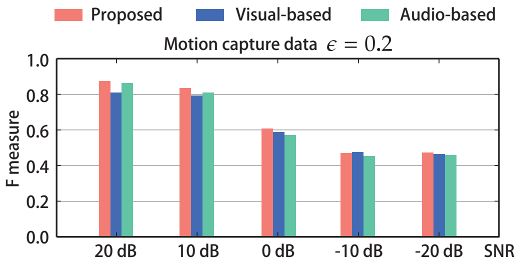


Fig. 8. Experimental results for noise-contaminated audio signals with motion capture data.

solve several problems in the future. First, real-time beat tracking often fails for music audio signals with complicated rhythms such as syncopation, and dance movements, such as slowly-varying movements. In addition, the response of the proposed beat-tracking method is not fast enough because correct beat times cannot be estimated stably before several tens of beat times have passed from the beginning of a musical piece, as seen in **Fig. 7**. Second, it is difficult to perform real-time beat tracking for music audio signals recorded by a microphone attached to the robot. One way to suppress self-generated motor noise originating from the robot's own dance movements would be to extend a semi-blind source separation method [31] such that noise sounds to be suppressed can be predicted from the dancing movements.

5. Application to Robot Dancer

This section presents a entertainment humanoid robot capable of singing and dancing to a song in an improvisational manner while recognizing the beats and chords of the song in real time. Among various kinds of entertainment robots that are expected to live with humans in the future, music robots, such as robot dancers and singers, are considered to be one of the most attractive applications of music analysis techniques. Our robot mainly consists of listening, dancing, and singing functions. The listening function captures music audio signals and recognizes the beats and chords in real time.

5.1. Internal Architecture

The listening, dancing, and singing functions are communicated among themselves in an asynchronous manner through data streams managed by the Robot Operating System (ROS) (**Fig. 9**). The listening function, which

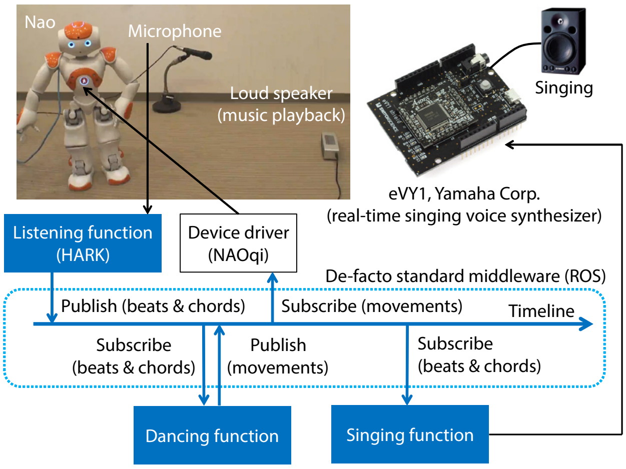


Fig. 9. System architecture of a singing robot dancer.

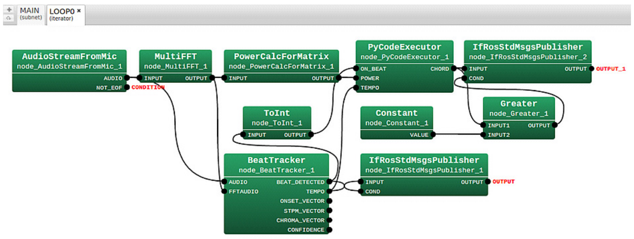


Fig. 10. Visual programming interface of HARK.

is implemented with HARK, an open-source robot audition software, takes music audio signals captured by a microphone and recognizes the beats and chords of those signals in real time. The dancing function then receives the recognition results and then determines dance movements. The singing function also receives the recognition results, determines vocal pitches and onsets, and synthesizes singing voices by using a singing-voice synthesizer called eVY1, Yamaha Corp. (MIDI device).

5.2. Listening Function

The listening function mainly consists of two modules: the beat tracking proposed in this paper and chord estimation, which are performed in real time on the HARK dataflow-type visual programming interface (**Fig. 10**). The latter module classifies 12-dimensional beat-synchronous chroma vectors extracted from music spectra into 24 chords (12 root notes × 2 types (major/minor)). To enhance the accuracy of chord estimation, we used von Mises-Fisher mixture models rather than standard Gaussian mixture models as classifiers [32].

5.3. Dancing and Singing Functions

The dancing function concatenates dance movements according to the chord progression of a target musical piece. We defined 24 different dance movements corresponding to the 24 chords (**Fig. 11**). A proprietary device driver called NAOqi should be linked to the ROS to send control commands to the robot.

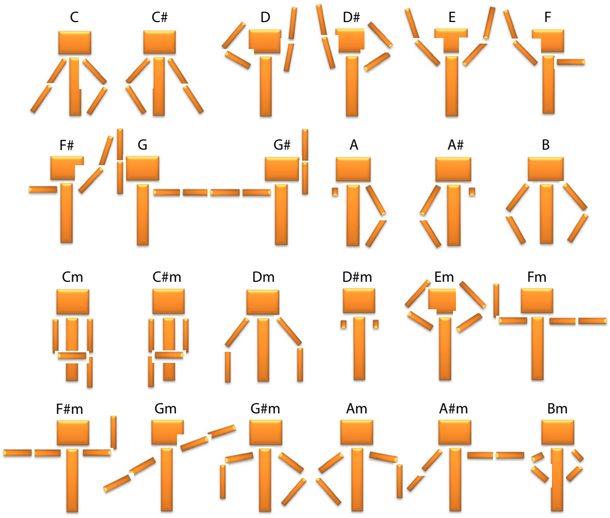


Fig. 11. Predefined dance movements.

The singing function controls the eVY1 device to generate beat-synchronous singing voices, the pitches of which match the root notes of the estimated chords. eVY1 can be controlled in real time as a standard MIDI device.

5.4. Discussion

We conducted an experiment using a sequence of simple chords (toy data) and a Japanese popular song (real data) in a standard echoic room without a singing function. Each signal was played back from a loudspeaker. The audio signals were captured through a microphone behind the robot. The distance between the loudspeaker and the microphone was about 1 m. Our robot has great potential as an entertainment robot because we felt that the robot generated chord-aware beat-synchronous dance movements. The dance response, however, came after a delay of two beats after new chords began because the robot has no chord prediction function. The development of prediction capability should be included in future work. Another research direction would be to generate more flexible and realistic dance movements by considering the body constraints of a robot. For example, it would be more exciting for a robot to be able to incrementally learn a human partner's dance movements to mimic those movements instead of generating predefined movements. To achieve this, the joint movements of a humanoid robot should be estimated such that the generated dancing motions are as close as possible to human motions, as in [7].

6. Conclusion and Future Work

This paper presented an audio-visual real-time beat-tracking method for a robot dancer that can perform in synchronization with music and human dancers. The proposed method, which focuses on both music audio signals and the joint movements of human dancers, is designed to be robust to noise and reverberation. To extract acoustic features from music audio signals, we estimate audio tempo likelihoods over possible tempos and an onset likelihood in each frame. Similarly, we calculate visual tempo likelihoods in each frame by analyzing the periodicity of the joint movements. These features included in each beat interval are gathered together into an observation vector and then fed into a unified state-space model that consists of latent variables (tempo and beat time) and observed variables (acoustic and visual features). The posterior distribution of the latent variables is estimated in an online manner by using a particle filter. We described an example implementation of a singing and dancing robot using HARK robot audition software and the Robot Operating System (ROS). The comparative experiments using two types of datasets, namely motion capture data and Kinect data, clearly showed that the probabilistic integration of intermediate information obtained by audio and visual analysis significantly improved the performance of real-time beat tracking and was robust against noise.

Future work will include improvement of audio-visual beat tracking, especially when Kinect is used, by explicitly estimating the failure or success of joint-position estimation in a state-space model. When microphones are attached to a robot and the recorded music signals are contaminated by self-generated noise, semi-blind independent component analysis (ICA) [31] is a promising solution, canceling such kinds of *highly predictable* noise (see [5]). In addition, it is important to estimate bar lines and relative positions of beat times in a bar by extending the latent space of a state-space model to generate more rhythm-aware dance movements. To develop a more advanced robot to dance with humans, we plan to conduct subjective experiments using various kinds of music.

Acknowledgements

This study was supported in part by JSPS KAKENHI Nos. 24220006, 26700020, 24700168, 26280089, 15K16054, and 16J05486 as well as the JST CREST OngaCREST and ACCEL OngaACCEL projects.

References:

[1] Y. Sasaki, S. Masunaga, S. Thompson, S. Kagami, and H. Mizoguchi, "Sound localization and separation for mobile robot teleoperation by tri-concentric microphone array," J. of Robotics and Mechatronics, Vol.19, No.3, pp. 281-289, 2007.

[2] Y. Sasaki, M. Kaneyoshi, S. Kagami, H. Mizoguchi, and T. Enomoto, "Pitch-cluster-map based daily sound recognition for mobile robot audition," J. of Robotics and Mechatronics, Vol.22, No.3, pp. 402-410, 2010.

[3] Y. Kusuda, "Toyota's violin-playing robot," Industrial Robot: An Int. J., Vol.35, No.6, pp. 504-506, 2008.

[4] K. Petersen, J. Solis, and A. Takanishi, "Development of a aural real-time rhythmical and harmonic tracking to enable the musical interaction with the Waseda flutist robot," Int. Conf. on Intelligent Robots and Systems (IROS), pp. 2303-2308, 2009.

[5] K. Murata, K. Nakadai, R. Takeda, H. G. Okuno, T. Torii, Y. Hasegawa, and H. Tsujino, "A beat-tracking robot for human-robot interaction and its evaluation," Int. Conf. on Humanoid Robots (Humanoids), pp. 79-84, 2008.

[6] K. Kosuge, T. Takeda, Y. Hirata, M. Endo, M. Nomura, K. Sakai, M. Koizumu, and T. Oconogi, "Partner ballroom dance robot – PBDR –," SICE J. of Control, Measurement, and System Integration, Vol.1, No.1, pp. 74-80, 2008.

[7] S. Nakaoka, K. Miura, M. Morisawa, F. Kanehiro, K. Kaneko, S. Kajita, and K. Yokoi, "Toward the use of humanoid robots as assemblies of content technologies – realization of a biped humanoid robot allowing content creators to produce various expressions –," Synthesiology, Vol.4, No.2, pp. 80-91, 2011.

[8] W. T. Chu and S. Y. Tsai, "Rhythm of motion extraction and rhythm-based cross-media alignment for dance videos," IEEE Trans. on Multimedia, Vol.14, No.1, pp. 129-141, 2012.

[9] T. Shiratori, A. Nakazawa, and K. Ikeuchi, "Rhythmic motion analysis using motion capture and musical information," Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI), pp. 89-94, 2003.

[10] T. Itohara, T. Otsuka, T. Mizumoto, T. Ogata, and H. G. Okuno, "Particle-filter based audio-visual beat-tracking for music robot ensemble with human guitarist," Int. Conf. on Intelligent Robots and Systems (IROS), pp. 118-124, 2011.

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

[12] G. Weinberg, A. Raman, and T. Mallikarjuna, "Interactive jamming with shimon: A social robotic musician," Int. Conf. on Human Robot Interaction (HRI), pp. 233-234, 2009.

[13] K. Petersen, J. Solis, and A. Takanishi, "Development of a real-time instrument tracking system for enabling the musical interaction with the Waseda flutist robot," Int. Conf. on Intelligent Robots and Systems (IROS), pp. 313-318, 2008.

[14] A. Lim, T. Mizumoto, L. K. Cahier, T. Otsuka, T. Takahashi, K. Komatani, T. Ogata, and H. G. Okuno, "Robot musical accompaniment: Integrating audio and visual cues for real-time synchronization with a human flutist," Int. Conf. on Intelligent Robots and Systems (IROS), pp. 1964-1969, 2010.

[15] M. Ohkita, Y. Bando, Y. Ikemiya, K. Itoyama, and K. Yoshii, "Audio-visual beat tracking based on a state-space model for a music robot dancing with humans," Int. Conf. on Intelligent Robots and Systems (IROS), pp. 5555-5560, 2015.

[16] K. Nakadai, H. G. Okuno, and H. Kitano, "Real-time auditory and visual multiple-speaker tracking for human-robot interaction," J. of Robotics and Mechatronics, Vol.14, No.5, pp. 479-489, 2002.

[17] S. Dixon," Evaluation of the audio beat tracking system BeatRoot," J. of New Music Research, Vol.36, No.1, pp. 39-50, 2007.

[18] M. Goto, "An audio-based real-time beat tracking system for music with or without drum-sounds," J. of New Music Research, Vol.30, No.2, pp. 159-171, 2001.

[19] A. M. Stark, M. E. P. Davies, and M. D. Plumbley, "Realtime beat-synchronous analysis of musical audio," Int. Conf. on Digital Audio Effects (DAFx), pp. 299-304, 2009.

[20] D. P. W. Ellis, " Beat tracking by dynamic programming," J. of New Music Research, Vol.36, No.1, pp. 51-60, 2007.

[21] M. E. P. Davies and M. D. Plumbley, "Context-dependent beat tracking of musical audio," IEEE Trans. on Audio, Speech, and Language Processing, Vol.15, No.3, pp. 1009-1020, 2007.

[22] J. L. Oliveira, G. Ince, K. Nakamura, K. Nakadai, H. G. Okuno, L. P. Reis, and F. Gouyon, "Live assessment of beat tracking for robot audition," Int. Conf. on Intelligent Robots and Systems (IROS), pp. 992-997, 2012.

[23] A. Elowsson, "Beat tracking with a cepstroid invariant neural network," Int. Society for Music Information Retrieval Conf. (ISMIR), pp. 351-357, 2016.

[24] S. Böck, F. Krebs, and G.Widmer, "Joint beat and downbeat tracking with recurrent neural networks," Int. Society for Music Information Retrieval Conf. (ISMIR), pp. 255-261, 2016.

[25] F. Krebs, S. Böck, M. Dorfer, and G. Widmer, "Downbeat tracking using beat synchronous features with recurrent neural networks," Int. Society for Music Information Retrieval Conf. (ISMIR), pp. 129-135, 2016.

[26] S. Durand and S. Essid, "Downbeat detection with conditional random fields and deep learned features," Int. Society for Music Information Retrieval Conf. (ISMIR), pp. 386-392, 2016.

[27] C. Guedes, "Extracting musically-relevant rhythmic information from dance movement by applying pitch-tracking techniques to a video signal," Sound and Music Computing Conf. (SMC), pp. 25-33, 2006.

[28] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," IEEE Trans. on Signal Processing, Vol.50, No.2, pp. 174-188, 2002.

[29] K. Nakadai, T. Takahashi, H. G. Okuno, H. Nakajima, Y. Hasegawa, and H. Tsujino, "Design and implementation of robot audition system 'hark' – open source software for listening to three simultaneous speakers," Advanced Robotics, Vol.24, No.5-6, pp. 739-761, 2010.

[30] R. A. Rasch, "Synchronization in performed ensemble music," Acta Acustica united with Acustica, Vol.43, No.2, pp. 121-131, 1979.

[31] R. Takeda, K. Nakada, K. Komatani, T. Ogata, and H. G. Okuno, "Exploiting known sound source signals to improve ICA-based robot audition in speech separation and recognition," Int. Conf. on Intelligent Robots and Systems (IROS), pp. 1757-1762, 2007.

[32] S. Maruo, "Automatic chord recognition for recorded music based on beat-position-dependent hidden semi-Markov model," Master's thesis, Kyoto University, 2016.

Supporting Online Materials:

[a] Murata Manufacturing Co., Ltd., Cheerleaders Debut, 2015.
http://www.murata.co.jp/cheerleaders/
[Accessed September 1, 2016]

[b] University of Cyprus, Dance Motion Capture Database, 2014.
http://dancedb.cs.ucy.ac.cy/ [Accessed September 1, 2016]

Name:
Misato Ohkita

Affiliation:
Speech and Audio Processing Group, Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University

Address:
Room 417, Research Bldg. No.7, Yoshida-honmachi, Sakyo-ku, Kyoto 606-8501, Japan

Brief Biographical History:
2015- Graduate School of Informatics, Kyoto University

Main Works:
- "Audio-Visual Beat Tracking Based on a State-Space Model for a Music Robot Dancing with Humans," 2015 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS 2015), pp. 5555-5560, 2015.

Membership in Academic Societies:
- Information Processing Society of Japan (IPSJ)

Name:
Yoshiaki Bando

Affiliation:
Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University
JSPS Research Fellow DC1

Address:
Room 417, Research Bldg. No.7, Yoshida-honmachi, Sakyo-ku, Kyoto 606-8501, Japan

Brief Biographical History:
2014 Received M.Inf. degree from Graduate School of Informatics, Kyoto University
2015- Ph.D. Candidate, Graduate School of Informatics, Kyoto University

Main Works:
- "Posture estimation of hose-shaped robot by using active microphone array," Advanced Robotics, Vol.29, No.1, pp. 35-49, 2015 (Advanced Robotics Best Paper Award).
- "Variational Bayesian Multi-channel Robust NMF for Human-voice Enhancement with a Deformable and Partially-occluded Microphone Array," European Signal Processing Conf. (EUSIPCO), pp. 1018-1022, 2016.
- "Microphone-accelerometer based 3D posture estimation for a hose-shaped rescue robot," IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 5580-5586, 2015.

Membership in Academic Societies:
- The Institute of Electrical and Electronic Engineers (IEEE)
- The Robotics Society of Japan (RSJ)
- Information Processing Society of Japan (IPSJ)

Name:
Eita Nakamura

Affiliation:
JSPS Postdoctoral Fellow, Speech and Audio Processing Group, Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University

Address:
Room 417, Research Bldg. No.7, Yoshida-honmachi, Sakyo-ku, Kyoto 606-8501, Japan

Brief Biographical History:
2012 Received Ph.D. degree from Department of Physics, University of Tokyo
2013- Postdoc Researcher, National Institute of Informatics; Meiji University; Kyoto University
2016- JSPS Postdoctoral Fellow, Graduate School of Informatics, Kyoto University

Main Works:
- "A Stochastic Temporal Model of Polyphonic MIDI Performance with Ornaments," J. of New Music Research, Vol.44, No.4, pp. 287-304, 2015.

Membership in Academic Societies:
- The Institute of Electrical and Electronic Engineers (IEEE)
- Information Processing Society of Japan (IPSJ)

Name:
Katsutoshi Itoyama

Affiliation:
Assistant Professor, Speech and Audio Processing Group, Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University

Address:
Room 417, Research Bldg. No.7, Yoshida-honmachi, Sakyo-ku, Kyoto 606-8501, Japan

Brief Biographical History:
2011 Received Ph.D. degree from Graduate School of Informatics, Kyoto University
2011- Assistant Professor, Graduate School of Informatics, Kyoto University

Main Works:
- "Query-by-Example Music Information Retrieval by Score-Informed Source Separation and Remixing Technologies," EURASIP J. on Advances in Signal Processing, Vol.2010, No.1 pp. 1-14, January 17, 2011.

Membership in Academic Societies:
- The Institute of Electrical and Electronics Engineers (IEEE)
- The Acoustical Society of Japan (ASJ)
- Information Processing Society of Japan (IPSJ)

Name:
Kazuyoshi Yoshii

Affiliation:
Senior Lecturer, Speech and Audio Processing Group, Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University

Address:
Room 412, Research Bldg. No.7, Yoshida-honmachi, Sakyo-ku, Kyoto 606-8501, Japan

Brief Biographical History:

2008- Received Ph.D. degree from Graduate School of Informatics, Kyoto University

2008- Research Scientist, Information Technology Research Institute (ITRI), National Institute of Advanced Industrial Science and Technology (AIST)

2013- Senior Researcher, AIST

2014- Senior Lecturer, Graduate School of Informatics, Kyoto University

Main Works:

- "A Nonparametric Bayesian Multipitch Analyzer Based on Infinite Latent Harmonic Allocation," IEEE Trans. on Audio, Speech, and Language Processing, Vol.20, No.3, pp. 717-730, 2012.

Membership in Academic Societies:

- The Institute of Electrical and Electronic Engineers (IEEE)
- Information Processing Society of Japan (IPSJ)
- The Institute of Electronics, Information, and Communication Engineers (IEICE)