Integration of flutist gesture recognition and beat tracking for human-robot ensemble

Takeshi Mizumoto, Angelica Lim, Takuma Otsuka,
Kazuhiro Nakadai, Toru Takahashi, Tetsuya Ogata and Hiroshi G. Okuno

***Abstract*— A novel human-robot trio-ensemble system with a human flutist, a human drummer and a robot-thereminist is presented. The participants of the ensemble play music *in a score-based way* and *simultaneously*, which has only been achieved independently. In our ensemble, both auditory and visual cues are used to synchronize the participants' performances. The ensemble begins with the flutist's start gesture. The participants of the ensemble play in accordance with the drummer's playing speed, then, they finish playing at the flutist's end gesture. The ensemble system is developed by integrating the following three components. The robot recognizes the flutist's gestures by using a robust Hough line detection with random sample consensus and finite state machines. The drummer's playing speed is recognized using spectro-temporal pattern-matching based real-time beat-tracking. The robot plays the theremin using the feedforward control method based on the theremin's pitch and volume models. We preliminarily evaluate the performance using the difference of onset timings between the played sounds and the reference onset timings calculated from a score. The results suggest that our system realizes a trio-ensemble with two humans and a robot.**

I. Introduction

Musical entertainment is crucial to achieve symbiosis between robots and humans because music can provide entertainment for many people regardless of age, cultural, and linguistic barriers. For example, even if people use different languages or are different ages, they can clap to the same music. Therefore, the capability of a robot that plays or recognizes music is expected to facilitate the symbiosis.

Solo-player robots that play an instrument have been actively studied. For example, Sugano *et al.* developed a humanoid robot that plays a keyboard with its fingers [1], Kaneko *et al.* developed an artificial lip and a lip-control method for a robotic trombone player [2], Singer *et al.* report many robots that play music or robotic instruments developed in LEMUR [3], Solis *et al.* developed a human-like flutist robot [4], and Mizumoto *et al.* constructed the models of theremin's pitch and volume characteristics, and developed a feedforward hardware-independent theremin playing system based on these models [5].

Our goal is to realize *a human-robot ensemble*, because interaction is essential for entertainment robots. The human-robot ensemble makes entertainment more enjoyable, because people can participate in it instead of being the audience. From the viewpoint of entertainment robotics, the robots in these studies have limited interactivity, since they developed as sophisticated *solo*-player robots. In addition, the ensemble can be interpreted as a task of a human-robot cooperation that requires high-precision timing control.

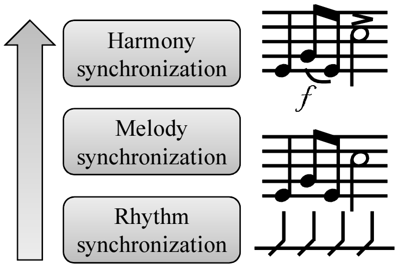

Fig. 1. Three levels of synchronization

One of the important questions in developing a human-robot ensemble is *how do we evaluate the ensemble as successful?* We define the ensemble as successful when the performances are *synchronized*, which is divided into three levels: rhythm which is focused on this paper, melody, and harmony (Fig. 1). We describe the overview and requirements of each level as follows.

Rhythm synchronization This level requires that onset times of each participant are the same without distinguishing the beats. For example, even if two participants plays the different positions of the score, they achieve this level when their onset timings match. Improvisation or duo-ensemble with a drummer can be realized when the level is achieved.

Melody synchronization This level requires that the playing positions of all participants are the same. They need to match the *global* position in his/her own score, as opposed to the rhythm synchronization which only requires *local* onset timings. A score-based ensemble that each participants has their own score will be realized when this level is achieved.

Harmony synchronization This level requires that the *balance* of each participant's play matches. Because an ensemble usually has multiple participants, music robots are required to take the relationship between other participants into account beyond the sophisticated play as a solo-player.

In this paper, we demonstrate a trio-ensemble system using three components that we have developed: a robot-thereminist system [5] as a participant of the ensemble, gesture recognition [6] for the rhythm synchronization on

T. Mizumoto, A. Lim, T. Otsuka, T. Takahashi, T. Ogata and H. G. Okuno are with Graduate School of Informatics, Kyoto University, Sakyo, Kyoto 606-8501, Japan { mizumoto, angelica, ohtsuka, tall, ogata, okuno } @kuis.kyoto-u.ac.jp

K. Nakadai is with Honda Research Institute Japan, Co., Ltd., Wako, Saitama, 351-0114, Japan, and also with Graduate School of Information Science and Engineering, Tokyo Institute of Technology. nakadai@jp.honda-ri.com

start and stop timings, and beat tracking [7] for the rhythm synchronization during the ensemble. In our previous works, these methods were partially combined; thus, integration of the methods was a remaining work. The melody synchronization is required to completely realize the ensemble, we focus on the rhythm synchronization.

This paper is organized as follows: Section II introduces state-of-the-art human-robot ensemble studies, Section III describes the our trio-ensemble system, Section IV illustrates the experimental results that demonstrate the performance of our system, and Section V concludes the paper.

II. Related Works on Human-Robot Ensemble

This section introduces several state-of-the-art human-robot ensembles from the viewpoint of two skills: *playing* and *interaction*. The interaction skill is especially important for an ensemble because the ensemble is one of interactions which requires the communication among the participants.

A. Playing Skill

Here, we discuss the three related works on the playing skill for robots. These works attain an ensemble hence these robots are used as a participant of an ensemble, which is discussed in the next section.

Solis *et al.* developed *Waseda Flutist Robot* (WF) that plays the flute with its own lips and fingers [8]. Using the approach similar to WF, they also developed the robot that plays the saxophone [9] These robots have a sophisticated playing skill of the flute and saxophone as a *solo*-player.

Weinberg *et al.* developed the robot called *Haile* that plays the drum [10]. Haile improvises along a human drummer by detecting the pattern of human's drum onsets and playing the stochastically-changed pattern. Hoffman *et al.* developed the robot called *Shimon* that plays the marimba with a human keyboardist [11]. Shimon obtains a sequence of played notes, searches a phrase which matches the sequence from the robot's database, and responds by playing the phrase.

Mizumoto *et al.* developed a robot thereminist system using the models of theremin's pitch and volume characteristics [12]. The system works on two different robots whose physical structures and control methods are different, because their models are independent from a particular hardware.

B. Interaction Skill

The interaction skill works are categorized into two topics: *recognition methods* for obtaining other participants' states from auditory or visual information, and *synchronization methods* for controlling a robot so that the robot's behavior synchronizes with that of the other participants.

1) Recognition Method: Real-time beat tracking is one of the most popular methods for an ensemble. Goto proposed a beat tracking method based on a multi-agents system [13]. The method was applied to develop a music robot that steps to a music by Yoshii *et al.* [14]. Murata *et al.* also developed a beat tracking method that uses spectro-temporal pattern-matching, and applied the robot that steps to the music [7].

There are also some works on recognition method for an ensemble. Otsuka *et al.* proposed a score following method, which is a research topic on human-computer ensembles [15], that is applicable to the human-robot ensemble [16] Lim *et al.* developed a finite-state-machine based gesture recognition method for a flutist [6]. Their method detects the beginning and ending times of an ensemble, which are difficult to detect only with auditory information.

TABLE I
Summary of related works

Related work	Synchronization level	Score usage	Participants Human	Participants Robot
Kotosaka [18]	Rhythm	Score-based	1	1
Petersen [19]	alternately played	Score-based	1	1
Weinberg [20]	Rhythm	Improvisation	2	2
Otsuka [21]	Rhythm	Score-based	1	1
Mizumoto [22]	Rhythm	Score-based	1	1
Ours	Rhythm	Score-based	2	1

2) Synchronization Method: In contrary to the recognition skills, studies on the synchronization skill for human-robot ensembles have recently started, although there are several works on human-computer ensembles with no physical body, e.g., jazz session system [17]. These works are characterized through two aspects: (1) synchronization level and (2) score usage, i.e., the ensemble is played based on a score, or improvisational. Our ensemble focuses on rhythm synchronization in a score-based way. Following related works are summarized in Table I.

Kotosaka *et al.* proposed a rhythmic motion control method using neural oscillators and developed a drummer robot [18]. In their work, participants play improvisational and simultaneously. They achieved the rhythm synchronization with the human drummer.

Petersen *et al.* developed a duo-ensemble between a human saxophonist and WF [19]. Their ensemble system is (1) alternately-played because the robot listens to the human's saxophone-play and plays an appropriate phrase based on the histogram of the spectrogram played by the human, and (2) score-based because the paper assumes that the human's score and robot's score are known.

Weinberg *et al.* developed a four-player ensemble among the two robots, *Haile* and *Shimon*, and the two humans, a keyboardist and a drummer [20]. The participants play improvisational and simultaneously. Their strategy of generating improvisational melody and rhythms depends on the human's playing, i.e., *Haile* plays similar rhythm patterns to the human drummer's, and *Shimon* plays a similar melody to the human keyboardist's. Therefore, their method is difficult to apply to an ensemble whose participants play different kinds of instruments.

Otsuka *et al.* integrated a real-time beat tracking method [7] and a robot-thereminist [5], and developed a duo-ensemble between a human drummer and a robot thereminist [21]. Although their integration was simple, they achieved a score-based ensemble that plays simultaneously.

Mizumoto *et al.* improved this ensemble system using a coupled-oscillator model [22] and reduced the difference between the human's drum onset time and the robot's note-change time, i.e., *the onset error*. Their ensemble is based on a given score and played simultaneously.

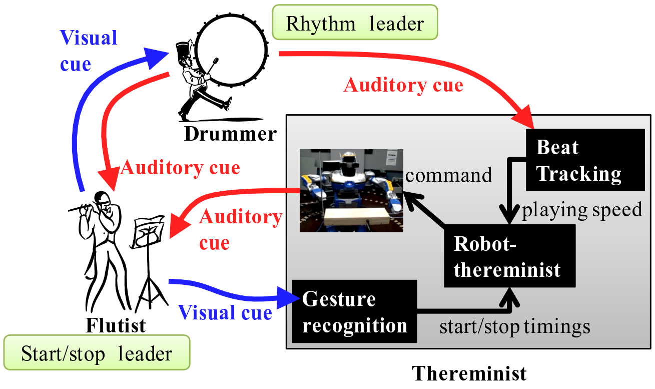

Fig. 2. Information flow of our trio-ensemble system

III. Human-Robot Trio-Ensemble System

This section describes our trio-ensemble system. First, we show the design of our system. Then, we briefly explain its three main components.

1) robot-thereminist system [5],
2) real-time beat tracking [7] and
3) flutist's gesture recognition method [6].

Finally, we describe the implementation of a trio-ensemble system including its architecture.

A. Design of trio-ensemble system

We develop a trio-ensemble system with simply integrating our state-of-the-art methods. The ensemble consists of three performers: a human flutist, a human drummer, and a robot-thereminist. Their roles are as follows.

1) **human flutist**: leads the beginning and the ending time by gestures, and plays the given score,
2) **human drummer**: leads the tempo during the ensemble by hitting the drum at the same interval, and
3) **robot-thereminist**: follows two other players and plays a given score.

We set the robot thereminist as a follower because we want to verify the capability of recognition of a human's gestures and playing speeds.

Fig. 2 illustrates an overview of our ensemble. During the ensemble, two kinds of cues are produced: visual and auditory. The drummer sends auditory cues to other participants, and the flutist sends visual cues to other participants. The robot recognizes the auditory cue using beat tracking, and the visual cue using gesture recognition. In contrary to the robot that has no confliction of cues, the flutist receives the two auditory cues from the drummer and thereminist.

B. Robot Thereminist System

A theremin is an electronic instrument which is played with no physical contacts (Fig. 3). It has two antennas for volume and pitch control, which is called a volume and a pitch antenna. When a player's left hand gets closer to the volume antenna, the volume gets smaller. When a right hand gets closer to the pitch antenna, the pitch gets higher.

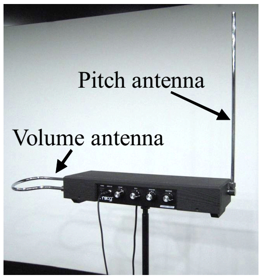

Fig. 3. Picture of theremin

We have developed the control method for robot thereminist in a feedforward way using parametric models of pitch and volume [12]. The model of a theremin's pitch characteristics, M_p, is formalized as follows.

$$\hat{p} = M_p(x_p; \boldsymbol{\theta}) = \frac{\theta_2}{(\theta_0 - x_p)^{\theta_1}} + \theta_3, \qquad (1)$$

where x_p denotes the robot's right arm position, $\boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2, \theta_3)$ denotes the model parameters, and $\hat{p}$ denotes the estimated pitch with M_p. The model of theremin's volume characteristics, M_v, is formalized combining polynomial functions with constants:

$$\hat{v} = M_v(x_v, x_p; \mathbf{a}(x_p), b(x_p)) = \begin{cases} \sum_{n=0}^{d} a_n(x_p) x_v^n & (b(x_p) < x_v) \\ v_{min} & (otherwise) \end{cases} \qquad (2)$$

where $\hat{v}$ denotes an estimated volume, d and $\mathbf{a}(x_p) = (a_n(x_p), \cdots, a_0(x_p))$ d denote the dimension and the coefficients of the polynomials, respectively, $b(x_p)$ denotes the boundary between the polynomial and the constant, and v_{min} denotes a level of background noise when the theremin is silent because its arm is enough close to the antenna.

The robot thereminist works with two phases: a calibration phase for parameter estimation and a performance phase for playing. In the calibration phase, the robot records both the robot's arm position, i.e., joint angle, the theremin's pitch, and volume at several positions. Then, the parameters of both models are estimated, using a least squares optimization method. In the performance phase, desired pitch and volume are converted to both arm positions using the inverse functions of Eqs. (1) and (2). We assume that the score has note names and volumes in dB. A note name is converted to the corresponding pitch with equal-temperament.

C. Real-time Beat Tracking for Auditory Cue Recognition

The method estimates the beat timings with (1) tempo estimation, (2) beat detection, and (3) beat prediction.

First, the tempo of the musical signal is estimated. Using the edge of a spectrogram obtained by Sobel filtering, the normalized cross-correlation function $R(t, i)$, which is defined by Eq. (3), is calculated. Sobel filter is used to detect the edge of the spectrogram, which corresponds to onsets.

$$R(t,i) = \frac{\displaystyle\sum_{f=1}^{62}\sum_{k=0}^{W-1} d_{inc}(t-k,f)\, d_{inc}(t-i-k,f)}{\sqrt{\displaystyle\sum_{f=1}^{62}\sum_{k=0}^{W-1} d_{inc}(t-k,f)^2 \cdot \sum_{f=1}^{62}\sum_{k=0}^{W-1} d_{inc}(t-i-k,f)^2}}, \qquad (3)$$

where t, f, i denotes the time, frequency and time-delay, respectively, d_{inc} denotes the Sobel-filtered spectrogram, W denotes the window length for estimating the tempo, and i denotes the shift offset. The maximum value of the local peaks of the correlation is the estimated tempo I. We set the window length W as 1 sec.

Then, the beat times are estimated using the Sobel-filtered spectrogram and the estimated tempo with two beat reliabilities, neighboring and continuous. The neighboring beat reliability, which is a function of the time-lag i, is obtained by adding the spectrum at the current frame $t+i$ and the frame of the beat time $t+I+i$. When the neighboring reliability is high, the next beat exists at the frame $t+I+i$. On the other hand, the continuous beat reliability reflects how long the beats come in a row. Finally, these two reliabilities are multiplied, and its peek is the estimated beat time.

Finally, the next beat timings is predicted. They are defined as the sum of the most recent beat time and the tempo.

D. Gesture Recognition of Flutist for Visual Cue Recognition

Flutists generates their gestures by moving up or down their flute, i.e., changing the orientation of it. The flutist's gesture recognition method hence needs two phases: (1) detecting the orientation of the flute, and (2) recognizing the gesture from the time series of the orientation. We describe for each phase as follows.

In the first phase, a robust line detection method is required because a flute is shiny, which leads to mis-estimated lines, i.e., outliers. The solution is a Hough-line detection [23] and pruning of outliers using a RANSAC (RANdom SAmple Consensus) algorithm [24].

As the preprocessing of the second phase, we translate from the time series of the flute's orientations to that of the state of the flute. We defined three states of the flute, down, up, and still, and translated using the given threshold. When the time-difference of the orientation is larger than the threshold, we assume that the flute's state changed.

Then, we input the time series to following two finite-state-machines (FSMs):

Start Cue	Down-Up-Down
End Cue	Down-Up

These FSMs are conflicted, for example, when the flutist gives the Down-Up-Down gesture, the method recognizes end cue at the second state, and start cue at the third state. However, we can filter the recognition easily using contexts, e.g, before the ensemble begins, we can only focus on the start cue. Note that we eliminate the third FSM, which indicates the flutist's playing beat, mentioned in [6] because it conflicts with the drummer's auditory cue. The integration of the flutist's auditory and visual cues are discussed in [6].

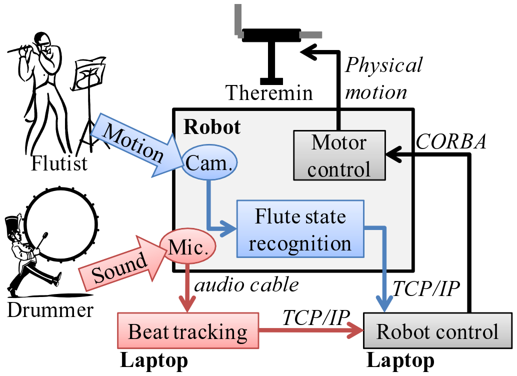

Fig. 4. Implementation of the ensemble system: the blue, red, and gray elements shows the data flow of visual cues, auditory cues, and motor commands, respectively.

E. Implementation

Fig. 4 shows an overview of the implementation of the trio-ensemble system. The system has three computers: (1) a computer in the robot for motor control, (2) a laptop for beat tracking, and (3) a laptop for robot controller. It is easy to integrate these two systems because (2) and (3) are connected through TCP/IP independently. In contrast, we need to concern about a computational cost to combine (1) and the other systems because (1) handles a real-time motor control.

The system works as follows. The robot captures the flutist's gestures with its own camera and recognizes it. Then, it sends the result, i.e., start/end cue, to the robot controller through TCP/IP. The robot also records mixture of the drummer's and theremin's sound, and sends it to the computer for beat-tracking through an audio cable. It sends the beat tracking result through TCP/IP. The beat tracking of the drum sound works well because the drum sound is in-harmonic, i.e., the vertical line appears in a spectrogram when the drum sound exists. The robot controller obtains these two results, and sends a motor command to the robot through CORBA, common object request broker architecture. After the robot receives the command, it plays the theremin with its two arms.

IV. Experiment

This section demonstrates the performance of our trio-ensemble system. We evaluate only the overall performance because the evaluation of each component is already done in our previous works. We evaluate how the ensemble achieved the rhythm synchronization, by comparing the onsets of each participants. Implicit interactions exists between human's and robot's because human participants receives the robot's playing sound and motion.

A. Configurations

We used a humanoid robot, HRP-2 from Kawada Industries, Inc., for the robot thereminist, and an Etherwave

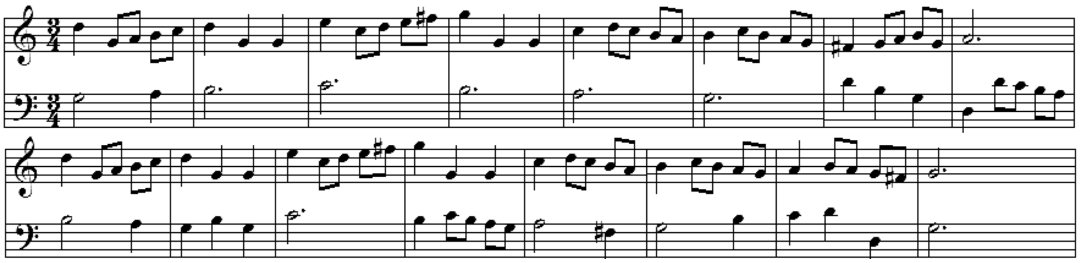

Fig. 5. Score of Menuett

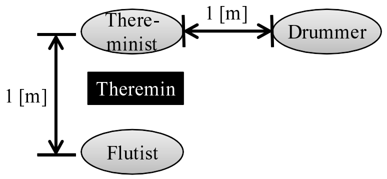

Fig. 6. Experimental configuration

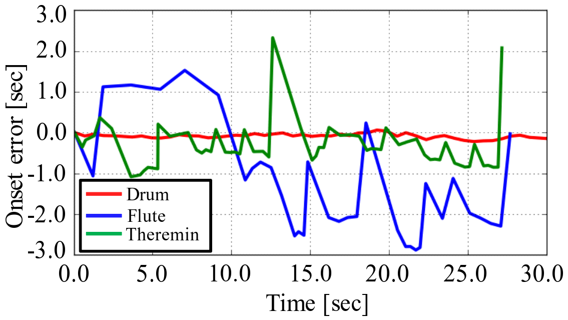

Fig. 8. Onset error of three participants compared to each score.

Theremin from Moog Music as the instrument. The distance between the robot and the theremin, the human flutist, and the human drummer were 50, 100 and 100 cm, respectively (Fig. 6). We adopted J. S. Bach's *Menuett, BWV Ahn. 114* as the music for the ensemble (Fig 5). The robot thereminist played the main part, i.e., the upper score of Fig. 5, and the human flutist played the sub part, i.e., the lower score.

The theremin's sound was recorded at 48 kHz through a microphone embedded in the robot's own head. The drummer's sound was also recorded with the microphone. The flutist's gestures were captured using the camera embedded in the robot's head.

The onsets of each participants are detected as follows. For the drum sound, we defined the onset timing as the time when the power of the sound is larger than 40 percent of the maximum power. For the theremin and flute sound, we defined the onset timing as the time when the pitch changed suddenly. We used a time series of pitch, instead of volume. When the pitch trajectory is flat, we determined that its beginning is the onset timing.

B. Results

Fig. 7 illustrates the onset timing for each participants. The horizontal line denotes time, and each circle denotes the onset of an instrument. The green, blue, and red circles denote the onset of the theremin, flute, and drum, respectively.

As shown in Fig. 7, the robot-thereminist plays the score, adapting to the drummer's onsets even if the onset estimation occasionally fails, for example, although the second onset of the theremin's and the drum's timings do not match, the error recovered at 4 sec, i.e., the fourth onset of the flute.

We also plot the onset error between the ideal onset timings calculated from the score (Fig. 8). Note that the error in Fig. 8 is calculated by comparing to the score not the drum's onset. The red line denotes the onset error of the drum. The red line is flat. This means that the drummer played at a nearly constant speed.

The green line denotes the onset errors of the theremin. Even if the drum's beat interval is not exactly the same because of the drum player's flctuation, the onset error of the theremin keeps around 0 sec. This is because the theremin adapts to the drummer using real-time beat tracking.

The blue line denotes the onset errors of the flute, which has the worst error. We guess that this is because the flutist could not decide which player to follow when onset of the drum and the theremin conflicted. The conflict occurs because of the beat-tracking method's mis-estimation or the motor-delay.

According to 13 - 14 sec in Fig. 8. the error of blue line (flutist) suddenly increases promptly after that of the green line (thereminist) increases. This correlation suggests that the flutist try to track the thereminist's play.

V. Conclusion

We presented a trio-ensemble system that uses visual and auditory cues for rhythm synchronization. We assumed that participants of the ensemble are: a human flutist who sends visual cues using gestures, a human drummer who sends auditory cues by a sequence of onsets, and a robot thereminist who plays the theremin using its pitch and volume characteristic models. The robot recognizes visual cues through finite-state-machine based gesture recognition, and auditory cues through real-time beat-tracking. Experimental results suggested that our methods are capable of constructing a trio-ensemble system. The results also suggested that when the other two participants' play conflicts, it is difficult for the remaining player to determine which participant to follow.

We have two ideas for future work. First, we need to solve the problem when cues are conflicted, which is ignored in this paper. Ignoring the problem is not suitable for the natural

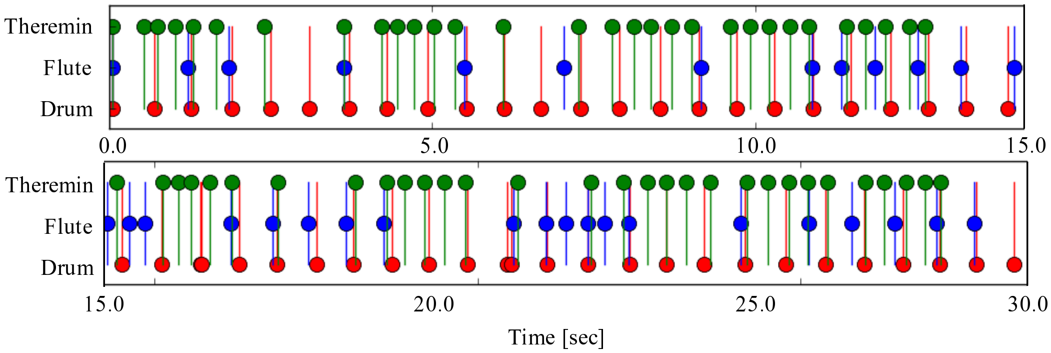

Fig. 7. Onset timings of each participant in trio-ensemble: each circle and line denotes each onset timing of each participant. There the rhythm synchronization will be achieved, when the circles are at the same time.

ensemble because, for example, a flutist sends auditory cues by playing a melody, and a drummer may sends visual cues. In prior to solve this problem, we need to separate each participant's playing sound and recognize individually. A robot audition system HARK [25] is a promising tool for sound separation. After we separate each playing sound, solving the cue conflicts is a next work for ensemble with multi-players. Finding a leader and tracking the turn taking in the ensemble is a way of solving this conflict.

Second, although we concentrated on rhythm synchronization in this paper, a melody synchronization is also important for an ensemble in which all participants have their own scores. After the synchronization achieved, we will tackle the problem of harmony synchronization.

Acknowledgements

This study was partially supported by a Grant-in-Aid for Scientific Research (S) (No. 19100003) and a Grant-in-Aid for Scientific Research on Innovative Areas (No. 22118502) from the MEXT, and the Global COE Program from the JSPS, Japan.

References

[1] S. Sugano and I. Kato. WABOT-2: Autonomous robot with dexterous finger-arm - finger-arm coordination control in keyboard performance -. In *Proc. of ICRA*, pages 90–97, 1987.

[2] Y. Kaneko, K. Mizutani, and K.Nagai. Pitch controller for automatic trombone blower. In *Proc. of ISMA*, pages 5–8, 2004.

[3] E. Singer, J. Feddersen, C. Redmon, and B. Bowen. LEMUR's musical robots. In *Proc. of NIME*, pages 181–184, 2004.

[4] J. Solis, M. Bergamasco, K. Chiba, S. Isoda, and A. Takanishi. The anthropomorphic flutist robot wf-4 teaching flute playing to beginner students. In *Proc. of ICRA*, pages 146–151, 2004.

[5] T. Mizumoto, H. Tsujino, T. Takahashi, T. Ogata, and H. G. Okuno. Thereminist robot: Development of a robot theremin player with feedforward and feedback arm control based on a theremin's pitch model. In *Proc. of IROS*, pages 2297–2302, 2009.

[6] A. Lim, T. Mizumoto, L. 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. In *Proc. of IROS*, 2010. *to appear*.

[7] K. Murata, K. Nakadai, K. Yoshii, R. Takeda, T. Torii, and H. G. Okuno. A robot uses its own microphone to synchronize its steps to musical beats while scatting and singing. In *Proc. of IROS*, pages 2459–2464, 2008.

[8] J. Solis, K. Taniguchi, T. Ninomiya, T. Yamamoto, and A. Takahashi. Development of Waseda flutist robot WF-4RIV: Implementation of auditory feedback system. In *Proc. of ICRA*, pages 3654–3659, 2008.

[9] J. Solis and A. Takanishi. Understanding the mechanisms of the human motor control by imitating saxophone playing with the waseda saxophonist robot WAS-1. In *Proc. of IROS Workshop*, pages 49–54, 2009.

[10] G. Weinberg and S.Driscoll. The interactive robotic percussionist - new developments in form, mechanics, perception and interaction design. In *Proc. of Proc. of HRI*, pages 456–461, 2007.

[11] G. Hoffman and G. Weinberg. Shimon: An interactive improvisational robotic marimba player. In *Proc. of ACM CHI*, pages 3097–3102, 2010.

[12] T. Mizumoto, H. Tsujino, T. Takahashi, K. Komatani, T. Ogata, and H. G. Okuno. Development of a theremin player robot based on arm-position-to-pitch and -volume models. *J. of Information Processing*, 2010. (In Japanese) *to appear*.

[13] M. Goto. An audio-based real-time beat tracking system for music with or without drum-sounds. *Journal of New Music Research*, 30(2):159–171, 2001.

[14] K. Yoshii, K. Nakadai, T. Torii, Y. Hasegawa, H. Tsujino, K. Komatani, T. Ogata, and H. G. Okuno. A biped robot that keeps steps in time with musical beats while listening to music with its own ears. In *Proc. of IROS*, pages 1743–1750, 2007.

[15] R.B. Dannenberg and C. Raphael. Music score alignment and computer accompaniment. *Comm. of the ACM*, 49:38–43, 2006.

[16] T. Otsuka, K. Nakadai, T. Takahashi, K. Komatani, T. Ogata, and H. G. Okuno. Design and implementation of two-level synchronization for interactive music robot. In *Proc. of AAAI*, 2010. *to appear*.

[17] M. Goto, I. Hidaka, H. Matsumoto, Y. Kuroda, and Y. Muraoka. A jazz session system for interplay among all players - VirJa session (virtual jazz session system). In *Proc. of ICMC*, pages 346–349, 1996.

[18] S. Kotosaka and S. Shaal. Synchronized robot drumming by neural oscillator. *Journal of Robotics Society of Japan*, 19(1):116–123, 2001.

[19] 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. In *Proc. of IROS*, pages 2303–2308, 2009.

[20] G. Weinberg, B. Blosser, T. Mallikarjuna, and A. Raman. The creation of a multi-human, multi-robot interactive jam session. In *Proc. of NIME*, pages 70–73, 2009.

[21] T. Otsuka, T. Mizumoto, K. Nakadai, T. Takahashi, K. Komatani, T. Ogata, and H. G. Okuno. Music-ensemble robot that is capable of playing the theremin while listening to the accompanied music. In *Proc. of IEA/AIE*, pages 102–112, 2010.

[22] T. Mizumoto, T. Otsuka, K. Nakadai, T. Takahashi, K. Komatani, T. Ogata, and H. G. Okuno. Human-robot ensemble between robot thereminist and human percussionist using coupled oscillator model. In *Proc. of IROS*, 2010. *to appear*.

[23] R. Duda and P. E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Comm. of the ACM*, 15(1):11–15, 1972.

[24] M. A. Fichler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Comm. of the ACM*, 24(6):381–395, 1981.

[25] K. Nakadai, H. G. Okuno, H. Nakajima, Y. Hasegawa, and H. Tsujino. Design and implementation of robot audition system "HARK". *Advanced Robotics*, 24:739–761, 2009.