International Journal of Smart Computing and Artificial Intelligence
International Institute of Applied Informatics
2019, Vol. 3, No. 1, 55 – 70

Two Approaches to Supporting Improvisational Ensemble for Music Beginners based on Body Motion Tracking

Shugo Ichinose [*], Souta Mizuno [*], Shun Shiramatsu [*], Tetsuro Kitahara[†]

Abstract

Melody recognition consists of three cognitive elements: pitch contour, rhythm, and tonality. Pitch contour and rhythm can be relatively easily represented by body motion. In comparison, tonality is difficult to understand and to represent for music beginners. In this paper, we focus on two approaches to supporting improvisational ensembles for music beginners on the basis of body motion tracking: one using a 3D motion capture camera and one using sensors in a smartphone. Users of our systems using these approaches can participate in improvisational ensembles by making hand movements that correspond to the pitch contour and rhythm without considering tonality because the generated pitch is automatically adjusted to a consonant pitch with the chords of a background tune. To deal with the delay and error in gesture recognition due to a 3D motion capture camera, we improved methods for recognizing gesture. The experimental results show that the delay of our method was improved over that of the conventional one. Furthermore, we implemented a method for motion tracking on the basis of smartphone sensors. The experimental results show the difficulties of motion tracking with smartphone sensors. Moreover, we discuss perspectives on the social reuse of improvisational melody data shared as open data.

Keywords: Body motion, improvisational ensemble, pitch contour, motion sensor, smartphone sensor

1 Introduction

It is difficult for music beginners to attempt musical improvisation by producing harmonic sounds without impairing tonality. Conventional ways of participating in improvisational music performances are limited to clapping or call-and-response for music novices. Techniques for enabling a broad range of citizens to participate in improvisational music have great social significance because such interaction can be utilized as an icebreaker activity in events citizens participate in.

There are three cognitive elements in melody recognition: rhythm, pitch contour, and

[*] Department of Computer Science, Nagoya Institute of Technology, Nagoya, Japan
[†] College of Humanities and Sciences, Nihon University, Tokyo, Japan

tonality [1]. Pitch contour, also known as melodic outline [2], refers to the rough movement of pitch. Tonality is a musical concept relevant to key, chord, and consonance. Among these three cognitive factors, recognizing and keeping tonality are difficult for music novices.

For musical novices, it is desirable that inharmonic and atonal pitches are automatically changed to tonal pitches. To satisfy this requirement, outputable pitches need to be restricted harmnonic or tonal pitches by managing the correspondence relationship between chord progress and outputable pitches. Hereafter, we call this relationship a “tonality constraint.” In comparison, rhythm and pitch contour are relatively easy for novices. These elements can be expressed through intuitive body motion.

In this study, we aim to develop an improvisational ensemble support system that outputs tones satisfying tonality constraints by automatically adjusting the tone pitch when a user inputs a pitch contour and rhythm through body motion. Specifically, when the user inputs a pitch contour by moving his or her hands, our system converts the hand movements into a sequence of tone pitches satisfying the tonality constraint for the chord progression of background music. We assume that such an approach will enable music novices to participate in improvisation ensembles.

In this paper, we consider two approaches developing systems for supporting improvisational ensembles. The first one is based on a 3D motion sensor camera. The second is based on smartphone sensors. We describe the implementation of a system using a 3D motion sensor camera in section 2. In section 3 the implementation of a system using smartphone sensors is presented. Section 4 shows a preliminary experiment for the approach using smartphone sensors.

2 Approach with 3D Motion Sensor Camera

The approach to supporting improvisational ensembles on the basis of a 3D motion sensor camera is presented in this section.

2.1 System Architecture

Figure 1 shows a system architecture. A screenshot of the system is shown in Figure 2.

First, a user inputs a rhythm and pitch contour to a motion sensor camera by moving his or her hands. In this study, we use the Intel RealSense 3D Camera as a motion sensor camera. RealSense detects fingers and recognizes gestures, and it sends recognition results to our ensemble support system.

Inside the system, fingertip coordinates are determined and the sounds of a performance are controlled by making gestures. The pitch is determined on the basis of the coordinates of the hand and the tonality constraint and is outputted as performance sound. Tonality constraints were statistically generated from data gathered from the Web API of Songle [3], which is a service for active music listening.

2.2 Inputting Body Motion

Figure 3 shows an overview of inputting a pitch contour through intuitive body motion. The time change of the height of the hand becomes the pitch contour input, and the gesture specifies the offset timing and the distinction of sustained/attenuated sound. Our user inter-face has dynamically varied areas for the tonality constraints. The areas are changed with the previously analyzed timing of the chord changes of a background tune.

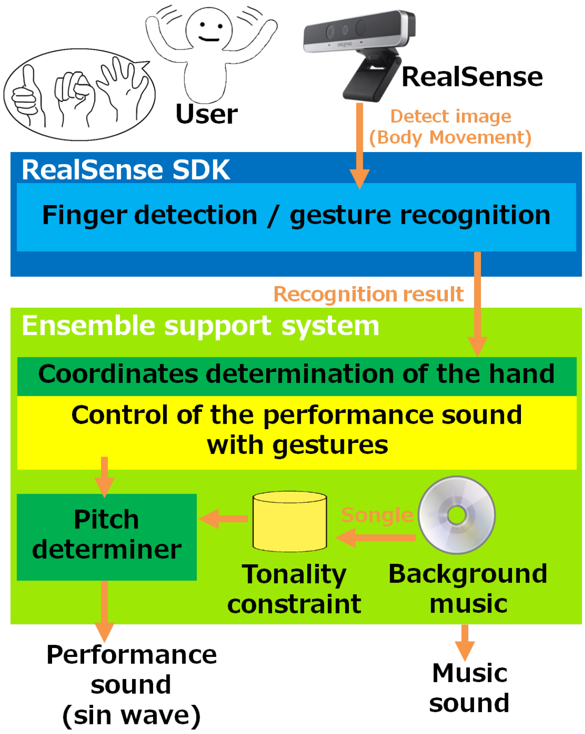


Figure 1: System Architecture

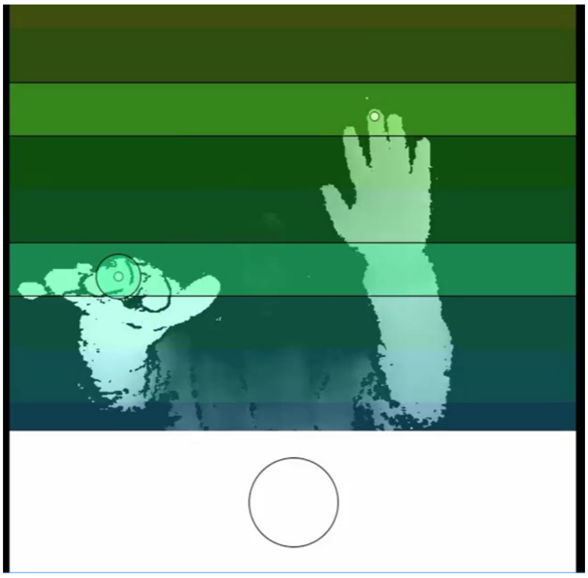


Figure 2: Screenshot of RealSense-based System

The number of areas depend on the number of outputable pitches. When a fingertip touches these areas, our system outputs a sine wave that satisfies the tonality constraints, i.e., the outputable pitches are restricted to consonant ones with these areas.

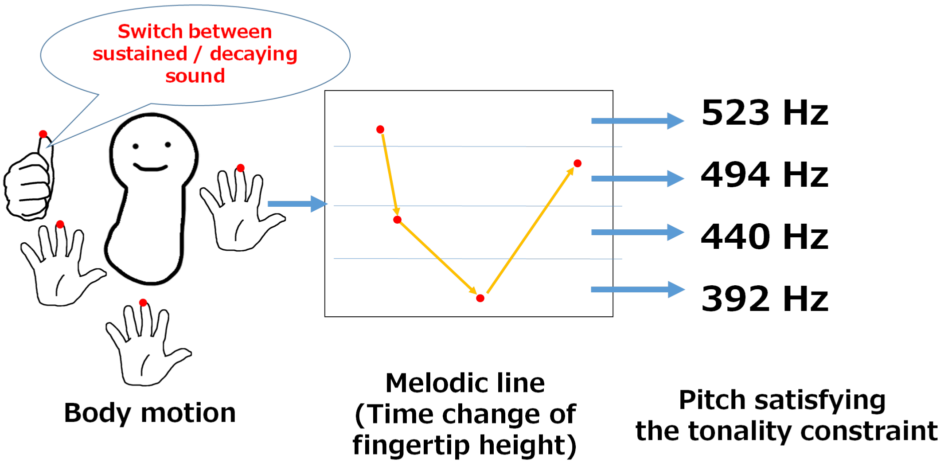

Switch between sustained / decaying sound
523 Hz
494 Hz
440 Hz
392 Hz
Body motion
Melodic line (Time change of fingertip height)
Pitch satisfying the tonality constraint

Figure 3: Method for Intuitively Inputting Pitch Contour

2.3 Changing Sound Patterns

Gesture assignments are as follows.

- “thumb_up”: Switch between sustained/decaying sound.
- “fist”: Specify timing of an offset of a sustained sound.
- “tap”: Specify attack timing of a decaying sound.

Since there were delay and missed recognition in the default gesture recognition functions of RealSense SDK, in this research, we improved the accuracies of the “fist” and “tap” functions as follows.

2.3.1 Improving “Fist” Gesture Function

To recognize of a “fist,” a value called “openness” is used. Openness is the value of the degree of the opening and closing of the fingers that can be acquired from RealSense and represented by a numerical value from 0 to 100. The system recognizes “fist” when the amount of change in openness per 1/30 seconds is less than -10.

2.3.2 Improving “Tap” Gesture Function

To recognize a “tap,” the speed and movement distance of the fingertip and palm are recognized as shown in Figure 4. The system recognizes a “tap” when these values satisfy the following conditions at the same time.

- Speed of fingertip is 1.8 m/s or more.
- Speed of palm is 0.6 m/s or more and 1.5 m/s or less.
- Movement distance of fingertip against direction to RealSense camera is 20 mm or more.

2.3.3 Future Works

For our future work, a proposed method for recognizing the motion of beating a virtual drum [4] is considered to be helpful. In this method, to recognize the beating of the drum, a threshold for the acceleration and angular velocity of the drum stick is used. If we can determine the angular velocity of a fingertip with RealSense, we can expect to improve the accuracy of recognizing taps by using this method.

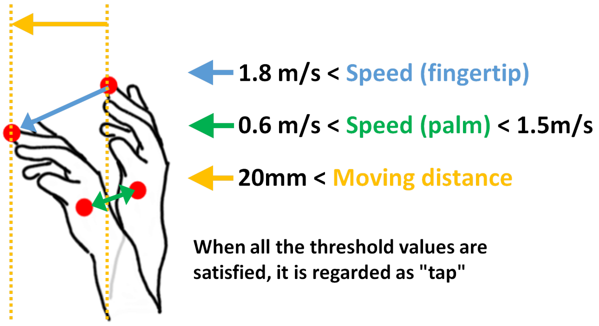

1.8 m/s < Speed (fingertip)
0.6 m/s < Speed (palm) < 1.5m/s
20mm < Moving distance
When all the threshold values are satisfied, it is regarded as "tap"

Figure 4: Threshold for Proposed “tap” Recognition Method

2.4 Generating Tonality Constraints

For musical novices, inharmonic and atonal pitches should be automatically changed to tonal pitches by tonality constraints. A tonality constraint is generated as a CSV file consisting of lines representing the millisecond timing of chord changes and available pitches after the timing. Available pitches are determined by statistical analysis, i.e., tones frequently co-occurring with the target tones and constituent tones of a chord.

To investigate the scale degrees frequently used for each chord, we obtained the chord and melody data of 100 songs from Songle and statistically selected scale degrees frequently used on each chord. Since Songle can acquire information on legally copyrighted music uploaded on the web, it is possible to generate the tonality constraints of all songs satisfying such conditions.

In this research, the scale degree length refers to the length of each note co-occurring with a music chord. A chord and pitch whose relative positions from the root tone of a key are equal are counted as those of the same tone. For example, our system regards the following cases as #VII sounds on the chord IM (I major).

- the scale degree of G# is used for the chord AM in A major or A minor, i.e. the tonic is A.
- the scale degree of B on the chord CM in C major or C minor, i.e. tonic is C.

Even if same tonics and chords are used in major and minor keys, the scale degree length is calculated separately for major and minor keys. Let sdl_s be a scale degree length. $S = \{I, \#I, II, III, \#III, IV, \#IV, V, VI, \#VI, VII, \#VII\}$ represents a set of sounds. Let $C = \{c_1, c_2, \cdots\}$ be a set of notes included in a particular chord c. Let $R = \{r_1, r_2, \cdots, r_n\}$ be a set of pitches in the tonality constraints for the chord c, i.e., outputable pitches over the chord c. C is a subset of R. $R - C$, a difference set of R and C, consists of notes satisfying the following condition.

$$sdl_s > \alpha \cdot sdl_0,$$

where sdl_0 is the highest frequency sdl_s of a note $s \in S$. α is an empirically set threshold. An empirically determined magnification and threshold were set by referring to a distribution obtained by sorting the frequencies of sounds other than chords that constitute pitches in descending order. Let the element of R of each chord progression be a frequency sequence.

3 Approach with Smartphone Sensors

Our approach to supporting improvisational ensembles on the basis of smartphone sensors is presented in this section. To apply our method to participatory music events, commonly used devices are required for recognizing the body motion of users. Mobile devices such as smartphones are more suitable to such situation than motion capture cameras because smartphones have recently become widespread and commonly used. We preliminarily investigated whether smartphone sensors can be used for inputting pitch contours or not.

Acceleration sensors and gyro sensors have the potential to be used for recognizing pitch contours. In this section, we implement a method for recognizing pitch contours by using Bayesian networks with input from smartphone sensors. When users raise or lower a smartphone, movements detected by our method are reflected in the pitch contour.

Although there are no other options for the operation done to specify a pitch contour, there are three options for that for specifying rhythm.

1. **Shake**: The attack timing of notes is determined when a smartphone is shook. This shaking is detected by acceleration sensors and gyro sensors.

2. **Clap**: The attack timing of notes is specified by making a motion clap-like motion, which is detected by an ambient light sensor.

3. **Touch**: The attack timing of notes is determined when a button located on the screen of the smartphone is touched.

We designed a Bayesian network for estimating the pitch notation from sensors, shown in Figure 5, and a Bayesian network for estimating the attack timing, shown in Figure 6. Although the network for estimating the pitch notation is used for all motion options, that for estimating attack timing is used only for ”Clap” and ”Shake” because ”Touch” does not require attack timing to be estimated because the touchscreen of a smartphone is reliable enough.

To obtain Bayesian network models, we collected training data with an examinee experiment. The data were sensor data that were obtained as five examinees raised and lowered their smartphones in accordance with the pitch contour of the melody of an existing tune, “Edelweiss.” The models were trained for estimating the note pitch of the melody from the smartphone sensor data.

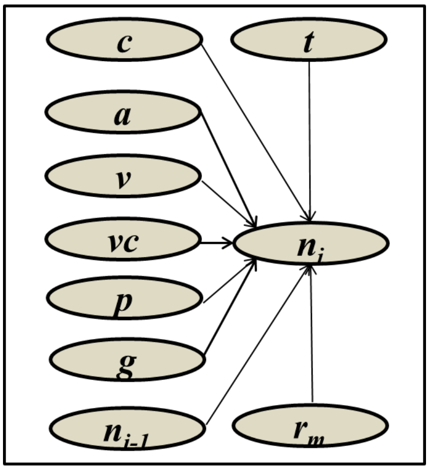


Figure 5: Bayesian Network for Estimating Pitch Notation

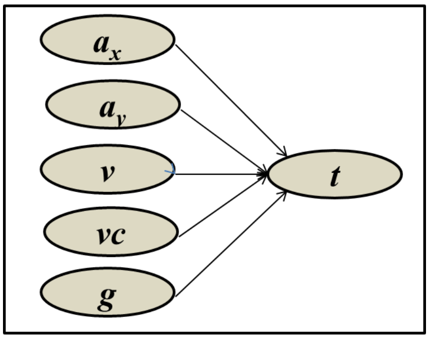


Figure 6: Bayesian Network for Estimating Attack Timing

4 Evaluation Experiment

4.1 Experiment with RealSense-based System

An evaluation experiment on the gesture recognition accuracy and tonality constraint was conducted with the RealSense-based system. Since delay was particularly noticeable in the tap functions of RealSense SDK, we evaluated only tapping in this study. Twelve men and women ranging in age from the 20's to 50's used the system and then were asked to answer the following questions with an evaluation value of 1 to 7.

First, subjects answered questions on musical skills.

- What is your musical experience?
  (1: no experience, 7: expert)

For the experiment on gesture function, subjects answered the following questions.

- Experiment 1. Comparison of proposed “ tap ” recognition method and RealSense SDK default one
  - Question 1-1: When a sound came out, was the timing as intended?
    (1: it was not the intended timing, 7: it was the intended timing)
  - Were there times when a “ tap ” was not recognized and sound did not come out?
    (1: it happened frequently, 7: it worked as intended)

In the exeperiment on tonality constraints, subjects answered the following questions.

- Experiment 2. Comparison of statistically generated tonality constraints and tonality constraints of only constituent sounds of a chord
  - Question 2-1: Did you feel that dissonance occurred?
    (1: dissonance was felt, 7: dissonance was not felt)
  - Question 2-2: Did the sound rise or fall as you intended?
    (1: it did not work as intended, 7: it worked as intended)

Also, as they became more familiar with the operation of the interface, there was the possi-bility that later evaluations may be affected. Therefore, to equalize the conditions, half the subjects were evaluated in reverse order.

As a supplement to Experiment 2, eight men and women ranging in age from the 20s to 50s operated the interface and measured the timing of the “tapping” and the delay of the performance.

- Experiment 3. “Tap” every beat and record the time when a “tap” was recognized (a circle flashed for each beat as a guide).

The results of Experiments 1 and 2 are shown in Figures 7 and 8. From Figure 7, in Experiment 1, the “tap” recognition method related to both Questions 1-1 and 1-2 gained a higher rating. From Figure 8, in Experiment 2, the result for Question 2-1 was inferior to the tonality constraint consisting of only pitches constituting a chord but did not greatly deteriorate. The result for question 2-2 showed that the statistically generated tonality constraint was slightly higher in the evaluation. Thus, the effectiveness of our proposed method for generating tonality constraints has not been proved at present. We need to check the consistency between the tendency of scale degree length and rules in musical theories. For example, there is a possibility that dissonance can be eliminated with simple rules for avoiding notes.

The results of Experiment 3 are shown in Figures 9 and 10. The average delay was -34.62 ms for the proposed method and 69.63 ms for the default one. According to one report [5], human beings are more uncomfortable with delays of more than 50 ms (gray area in Figures 9 and 10). Therefore, the delay average was closer to the beat timing and was within 50 ms, so it can be said that the proposed method was more accurate than the default one.

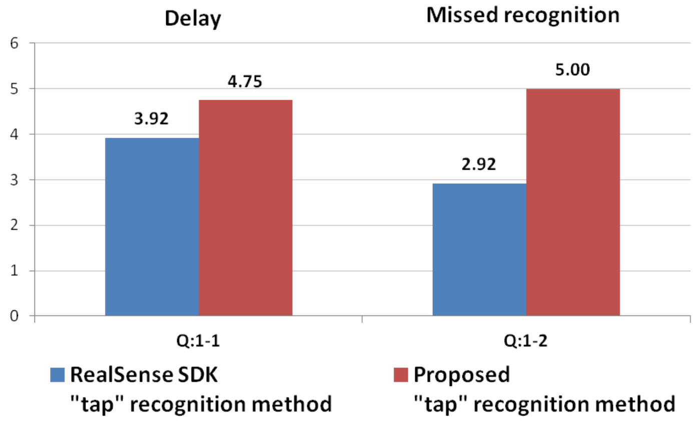


Figure 7: Result of Experiment 1. The higher the value, the better the evaluation. For Q1-2, the higher the value, the less the false recognition.

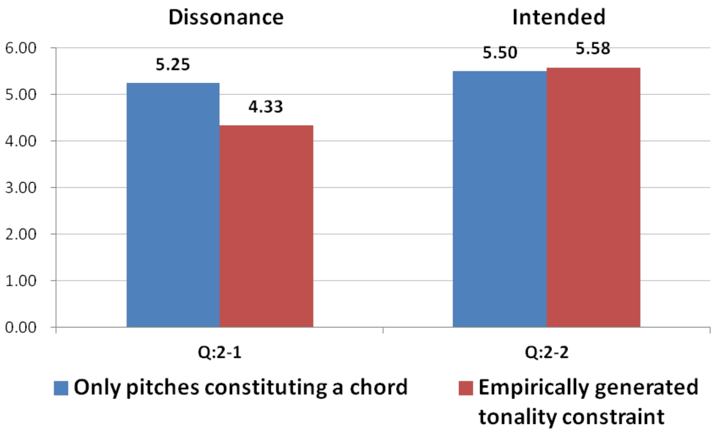


Figure 8: Result of Experiment 2. The higher the value, the better the evaluation.

Correlation coefficients between the musical skills and the score of each question are shown in Table 1. Subjects who have music experience tend to make harsh evaluations on the whole. In particular, the strongest trend was for Question 2-1 on adjustment.

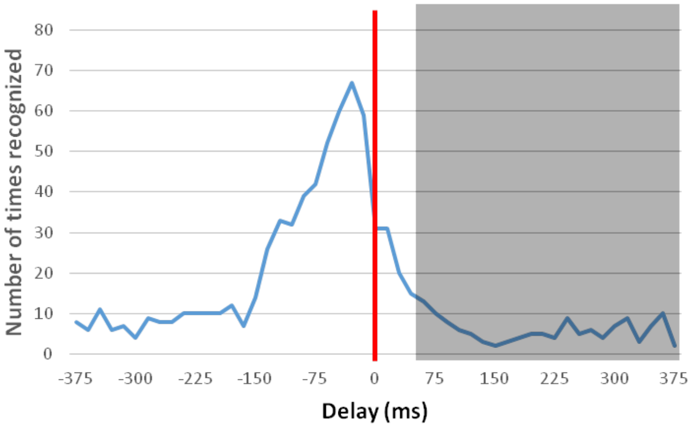


Figure 9: Proposed “Tap” Recognition Method

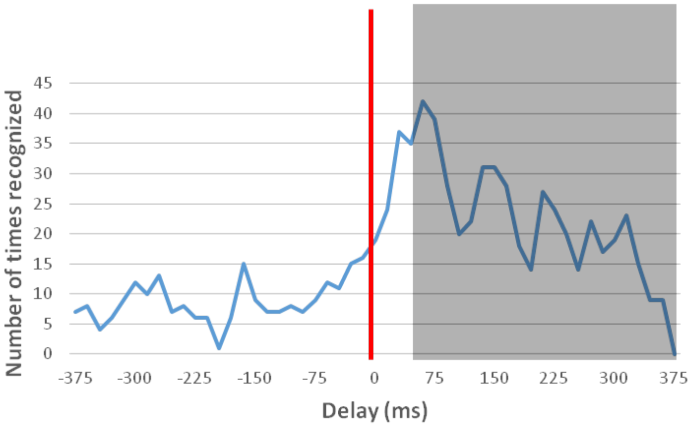


Figure 10: RealSense SDK “Tap” Recognition Method

Table 1: Correlation Coefficients Between Musical Skills and Score of Each Question

Methods	Default “tap” Method		Proposed “tap” Method		Default Tonality Method		Proposed Tonality Method	
Questions	1-1	1-2	1-1	1-2	2-1	2-2	2-1	2-2
Correlation coefficients	-0.58	-0.52	-0.22	-0.28	-0.71	-0.65	-0.44	-0.49

4.2 Experiment with Smartphone-based System

We experimented the smartphone-based system to examine the accuracy of a user‘s vertical motion matching the vertical change of pitch notation from the previous position of pitch notation n_{i-1} when a user performed the predefined shake, touch (using absolute distance), and touch (using relative distance) motions.

- Experiment 4. using input parameter n_{i-1}
- Experiment 5. not using input parameter n_{i-1}

The results shown in Figure 11 indicate that the accuracy in the matching between a user‘s vertical motion and vertical change in pitch notation was about 50% with each motion in Experiment 4. In Experiment 5, both shake and touch (using absolute distance) were not significantly different. These results show that a user‘s vertical motion did not match the vertical change in pitch notation often. However, in the results with touch (using relative distance), those of Experiment 5 are higher than those of Experiment 4. These results show that pitch patterns in a background tune more highly influenced the estimation of pitch notation than the vertical change in a user’s motion when using the value at the previous position of pitch notation n_{i-1} as an input value.

5 Social Reuse of Improvisational Melody Data

Our systems convert user body motion into a pitch contour and improvisational tonal melody. Pitch contour data and improvisational melody data are easily generated from each trial use. If users can publish and share their performance data as open data, the data can be socially reused for collaborative music composition or remixing. In particular, pitch contour data without tonality constraints can be applied to various chord progressions. For example, a loop sequencer based on a melodic outline [6] is a potential application of the open data of pitch contours. Such open data has great potential for derivative music created by musical novices.

6 Expansion for Improvisation Ensemble between Users

The systems implemented in the previous sections provided a function for improvising as an ensemble between a user and an existing background tune, i.e., a user can easily append a melody upon a background tune. In this section, we try to expand the RealSense-based system to enable users not only to create an ensemble with existing tunes but also to create one between users with dynamically controlled chord progressions.

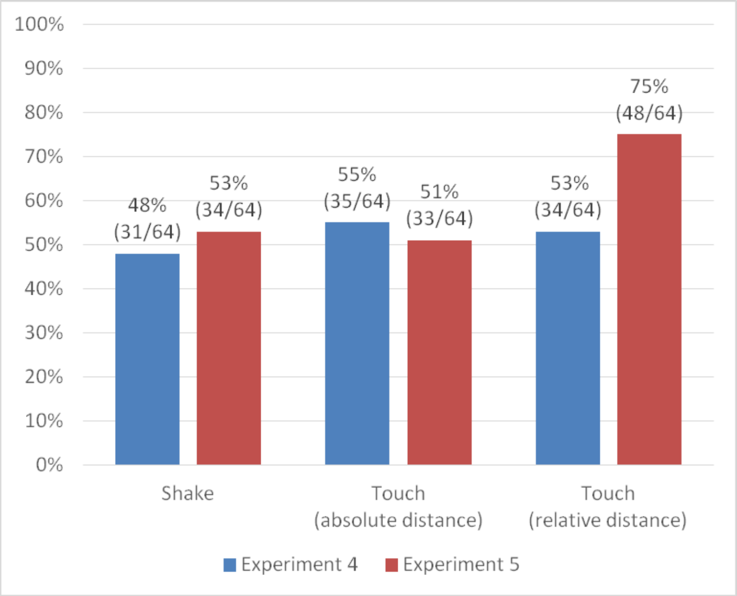


Figure 11: Result of Experiment for Smartphone-based System.

Figure 12 shows the architecture of the expanded system. This system consists of one central unit for an ensemble master that controls an accompaniment and multiple sub units for other users who play melody parts.

The prototype of the central unit was developed. Figure 13 shows a screenshot. The ensemble master can dyamically change the inversion of chords and the note patterns of an accompaniment with her/his body motion detected by RealSense. Several note patterns are prepared: whole note mode, quarter note mode, arpeggio mode, delay mode and free mode. The note patterns are selected with the icons shown in Figure 14 and specified by the location of the user's hand. In free mode, users can specify the attack timing of chords with a tap motion. Tap recognition on the central unit is also improved on the basis of our proposed method because a preliminary experiment before the improvement of the tap recognition showed that users were frustrated by the misrecognition of the default tap detection.

Sub units can be RealSense-based or Smartphone-based interfaces, and they are connected with the central unit. Timing information for synchronization should be sent from the central unit to sub units. This synchronization function is currently implemented. We are planning to conduct an experiment for verifying whether the expanded system enables improvisational ensembles between multiple users as future work.

7 Related Work

KAGURA [8] is also a gesture-based musical interface using RealSense. Although it enables a user to play music with intuitive body motion, it does not support the multi-party ensembles with distributed sub units we aimed for at in our expanded system in Section 6.

Spheremin [9] is a gesture-based musical interface using Kinect. This system focused on musical performances visualized on a spherical surface.

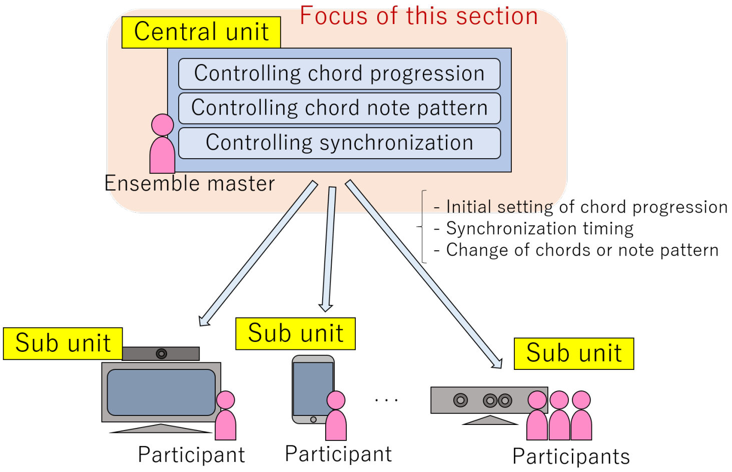
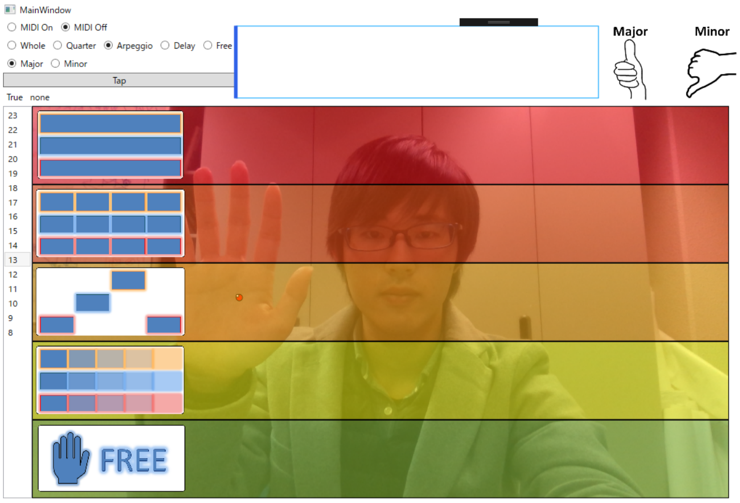

Focus of this section
Central unit
Controlling chord progression
Controlling chord note pattern
Controlling synchronization
Ensemble master
- Initial setting of chord progression
- Synchronization timing
- Change of chords or note pattern
Sub unit
Sub unit
Sub unit
Participant
Participant
Participants

Figure 12: Architecture of The Expanded System


MainWindow

○ MIDI On ● MIDI Off

○ Whole ○ Quarter ● Arpeggio ○ Delay ○ Free

● Major ○ Minor

True none

Figure 13: Screenshot of Prototype

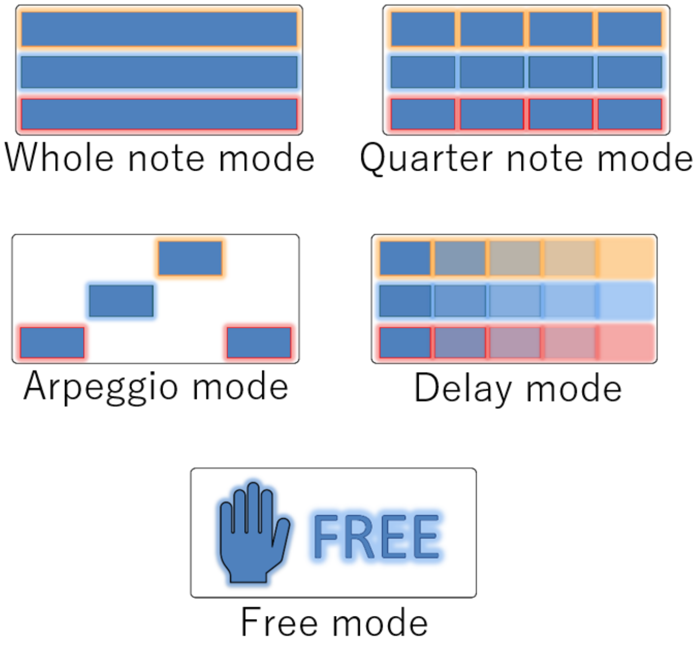


Figure 14: Icons for Representing Note Patterns of Accompaniment

RadarTHEREMIN [10] is a touchless musical interface using a harmonic table based on the fifth interval, major third interval, and minor third interval. Although the presumed user experiences of these systems are similar to that of our system, the focus and the approach are different from those of our study. While Spheremin and radarTHEREMIN use larger installations such as a dome or a room with a blue background, our system uses only a motion sensor camera because we aim to apply our system to events in which citizens participate.

There is a study on skill transmission through view-sharing between a theremin ex-pert and a novice using head-mounted displays [11]. Although we do not focus on skill transmission and music education, such view-sharing can be effective for grasping anothers player's performance in multi-party improvisation in our future work.

8 Conclusion

We designed and implemented two systems that support improvisational ensembles by detecting user hand movements corresponding to pitch contours and rhythm. The first system is based on a 3D motion sensor camera, RealSense 3D, and the second is based on smartphone sensors. The systems enable music novices to participate in improvisational ensembles because they automatically adjust note pitches to satisfy the tonality of background music.

The result of an experiment on the RealSense-based system shows that the proposed "tap" recognition has less delay and fewer omissions than the default one.

Regarding tonality constraints, we could not verify the effectiveness of the proposed method at present. Improvements such as applying rules of musical theory or using machine learning are necessary in the future.

The result of an experiment on the smartphone-based system showed the difficulty of estimating the pitch notation and attack timing with “shake” and “clap.” The accuracy should be improved to apply the methods to improvisational ensembles.

In the future, we will consider improving the tonality constraints on the basis of our model for tonality cognition [7]. Moreover, by extending our systems to multi-player improvisational ensembles, we will verify whether there is an ice-breaker effect from using improvisation ensembles in situations where agreement formation is necessary and whether there is an influence on user interaction and human relations. Furthermore, we are also con-sidering the social reuse of improvisational melody data shared as open data.

Acknowledgment

This work was partially supported by JSPS KAKENHI (No. 25870321, 17K00461, 17H00749, 16K16180, 16H01744, 26280089, 26240025).

References

[1] G. Hatano, “Music and Cognition,” University of Tokyo Press, 1981.

[2] T. Kitahara and Y. Tsuchiya, “Short-term and Long-term Evaluations of Melody Editing Method based on Melodic Outline,” Proceedings of the Joint International Computer Music and Sound and Music Computing Conference (ICMC|SMC| 2014), 2014, pp. 1204-1211.

[3] M. Goto et al., “Songle: A Web Service for Active Music Listening Improved by User Contributions,” Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011), 2011, pp. 311-316.

[4] H. Kanke et al., “Airstic Drum: Construction of a Drumstick for Integration of Real and Virtual Drums,” Transactions of Information Processing Society of Japan, 2013, pp. 1393-1401.

[5] K. Tanaka, K. Higuchi, and K. Ueno, “The effect of sound delay conditions on electronic drum performance,” Technical Committee of Musical Acoustics of Acoustical Society of Japan, 2013, pp. 1-6.

[6] T. Kitahara et al., “A Loop Sequencer That Selects Music Loops based on the Degree of Excitement,” Proceedings of the 12th Sound and Music Computing Conference (SMC 2015), 2015, pp. 435-438.

[7] S. Shiramatsu, T. Ozono, and T. Shintani, “A Computational Model of Tonality Cognition Based on Prime Factor Representation of Frequency Ratios and Its Application,” Proceedings of the 12th Sound and Music Computing Conference (SMC 2015), 2015, pp. 133-139.

[8] Shikumi Design Inc.: "KAGURA: Change Your Motion Into Music," https://www.kagura.cc (accessed on Jan. 2018).

[9] C. Mayer et al., "An Audio-visual Music Installation With Dichotomous user Interactions," November 2014 ACE '14: Proceedings of the 11th Conference on Advances in Computer Entertainment Technology.

[10] D. Marinos et al., "Design of a Touchless Multipoint Musical Interface in a Virtual Studio Environment," November 2011 ACE '11: Proceedings of the 8th International Conference on Advances in Computer Entertainment Technology.

[11] K. Kurosaki et al., "Skill Transmission for Hand Positioning Task Through View-sharing System," March 2011 AH '11: Proceedings of the 2nd Augmented Human International Conference.