The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Thereminist Robot: Development of a Robot Theremin Player with Feedforward and Feedback Arm Control based on a Theremin's Pitch Model

Takeshi Mizumoto[†], Hiroshi Tsujino[‡], Toru Takahashi[†], Tetsuya Ogata[†], Hiroshi G. Okuno[†]
†*Graduate School of Informatics, Kyoto University, Sakyo, Kyoto 606-8501, Japan*
‡*Honda Research Institute Japan, Co., Ltd., 8 - 1 Honcho, Wako-shi, Saitama 351-0114, Japan*
{*mizumoto, tall, ogata, okuno*}*@kuis.kyoto-u.ac.jp, tsujino@jp.honda-ri.com*

Abstract— **We propose a Thereminist robot system that plays the Theremin based on a Theremin's pitch model. The Theremin, which is a 1920s electronic musical instrument, is played by moving a player's hand position in the air without touching it. It is difficult to play the Theremin because the relationship between the hand position and Theremin's pitch (pitch characteristics) is non-linear and varies according to the electromagnetic field (hereafter called environment). These characteristics cause two problems: (1) Adapting to the environment change is required and (2) a naïve design tends to depend on robot's particular hardware. We implement the coarse-to-fine control system on the Thereminist robot using newly proposed two pitch models: parametric and nonparametric ones. The Thereminist robot works as below: first, the robot calibrates the pitch model by parameter fitting with the Levenberg-Marquardt method. Second, the robot moves its hand in a coarse manner by feedforward control based on the pitch model. Finally, the robot adjusts its position by feedback control (Proportional-Integral control). In these steps, the robot can play a required pitch quickly, because the robot moves its hand using the pitch model without listening to the Theremin's sound Thus, the time to play the exact pitch is shorter than when only feedback control is used. Three experiments were conducted to evaluate the robustness against the number of samples, environment change, and types of robots. The results revealed that our pitch model describes using only 12 samples of pitches for estimation of the parameters, and adapts if the environment changes. In addition, our system works on two different robots: HRP-2 and ASIMO.**

I. Introduction

Interaction between humans and robots through music in daily environments is expected to lead to new communication channels apart from spoken language. Because music is independent of a particular language, it should transcend language. To accomplish the interaction, robots should understand music with their own ears. This is because humans and robots listen to many kinds of music from TV, radio or other loudspeakers in daily environments.

Our purpose is to develop a music robot that understands music which is capable of interacting. In our previous work, we proposed a general architecture for music robots [1]. The architecture was based on the assumption that music-understanding robots consists of two capabilities: *music recognition* and *music expression*. Music robots in daily environments have to *recognize* the music by themselves because it is impossible to predict what kind of music robots will be exposed to. *Music expression* is also important because people should know the inner state of robots through their physical or acoustic expressions to have an musical interaction. In other words, this means that we should evaluate the robot's capability of music understanding by using the Turing Test [2]. The purpose of the test is to examine whether a machine has intelligent or not. If a human judge cannot reliably tell the robots from the human, the machine is said to have passed the test. Similarly, we insisted that if a human judge cannot distinguish the music robots from the human through an musical interaction, the robot has a music-understanding capability.

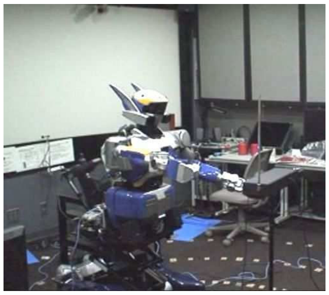
Fig. 1. HRP-2 plays the Theremin

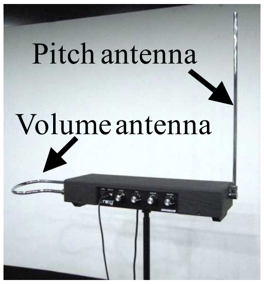

Fig. 2. The overview of the Theremin

Some studies have focused on music robots, e.g., a flutist [3], a drummer [4], a step-keeping [5] and a scatting robot [6]. The problems of these studies are: (1) they assumed the environment around the robot was stable and (2) they implemented the system on their particular robot. Strictly speaking, assumption (1) is invalid because the sounds that robots listen to or play depend on humidity, temperature and the dimensions of the room around them. Assumption (2) prevents us from transferring playing skills (expertise) from one robot to another.

In this study, we aim to eliminate these assumptions by using a Thereminist robot system and implemented it on HRP-2 Promet, (Fig. 1). We adopted the Theremin because the musical instrument enables us to deal with these two problems:

1) *Robustness to an environment*: the pitch characteristics of the Theremin change drastically according to the electromagnetic environment, therefore, the Theremin is suite for evaluating the robustness, and
2) *Expertise transfer:* The hardware requirement for the

978-1-4244-3804-4/09/$25.00 ©2009 IEEE

Thereminist robots is only position control function because the Theremin products the sound only moving robots' hands in the air. Therefore, we can prepare many robots and demonstrate the expertise transfer.

Our Thereminist robot system has two key features: (a) We designed the system in two layers: the first is independent of a robot and the second depends on its hardware. This feature enables us to transfer the playing skills, by modifying only the machine-dependent layer. (b) the system uses a pitch model for playing melodies. The pitch model describes the pitch characteristics including environmental changes. After calibrating the pitch model in advance, the model enables the robot to play any pitch quickly without listening to Theremin's sounds if the surrounding environment is stable after the calibration. Obviously, we need to adjust the robot's hand position with feedback control to reduce model error. However, the convergence time is short compared to only using feedback control because its initial position is close to the correct position.

The rest of this paper is organized as follows: Section II introduces related work on the expressions of music robots. Section III describes what the Theremin is and difficulties with playing it, Section IV describes our Thereminist robot system, Section V presents the results from three experiments on the robustness of our pitch model and the capability of the expertise transfer of our system, and, Section VI summarizes the paper.

II. Music Expressions of Related Music Robots

This section introduces related music robots' *music expression* capabilities because our concern in this paper is on music expression. There are three categories in expressing music that are described below. Our Thereminist robot is classified into **Playing an instrument**, which involves both expression using embodiment and quantitative evaluations.

Playing an instrument: Robots that play a music by playing an instrument are classified into this group. We can evaluate this kind of expression using pitches and timing. Many kinds of instruments are used, e.g., Toyota developed the Partner Robot that plays the trumpet, Solis *et al.* developed a robot that plays the flute with artificial lips [3], and Weinberg *et al.* developed Halie that plays a drum along with a human [4]. Kotosaka *et al.* developed a robot that plays a drum rhythmically according to a human's drumming [7] and Alford *et al.* developed a robot Theremin player [8].

Dancing: Robots that express music through their body motions are classified into this group. This is the most popular way of expressing music, and we have seen many dancing demonstrations by robots. Existing studies on music robots are, e.g., imitations of human dancing discussed by Nakaoka *et al.* [9] and the keep-stepping function on ASIMO discussed by Yoshii *et al.* [5]. However, more quantitative evaluations are required of this type of expression. Some work has tried to create such an evaluation metric using a Laban system [10]that still has subjective features.

Singing songs: Robots that sing songs with music with their own voice are classified into this group. This is the freest expression because there is no physical constraints. The simplest way of singing is to just play a song through loudspeakers, but this form of expression does not tend to use the embodiment of robots. There have been some studies on singing by considering embodiment that a robot listens to its own singing voice, e.g., the beat-counting robot discussed by Mizumoto *et al.* [1] and, the singer robot that counted musical beats developed by Murata *et al.* [6].

We focused on the task of playing the Theremin. Alford *et al.* also developed a robot that plays the Theremin [8], which controls its arm to play a required note with feedback and look-up-table-based feedforward control. The look-up table, which provides a corresponding arm position for a given pitch, is constructed by calibrating in advance. The calibration takes much time because the robot needs to find an appropriate arm position in every note. For the robustness to the environment, the calibration should be done quickly because once the environment changes, re-calibration is needed.

III. Introduction and Problems of The Theremin

This section describes what the Theremin is and problems with playing it.

A. Brief Description of the Theremin

The Theremin is one of the oldest electronic musical instruments developed by Léon Theremin in 1919 [11], which is a monophonic instrument. As shown in Figure 2, the Theremin has two antennas: vertical and horizontal ones for control its pitch and volume respectively. Without touching the instrument, the player controls pitch through relative proximity of the right hand to the vertical antenna, and volume through similar movements of the left hand in relation to the horizontal antenna. The Theremin's production of sound is more related to that of trombones or violins than pianos or flutes because its sound changes continuously, i.e., the Theremin does not have an obvious musical scale. These features are caused by the Theremin's mechanism.

We will now describe the mechanism for the Theremin. Only the pitch control is explained because volume control is beyond the scope of this paper. The Theremin uses two oscillation circuits and a low-pass filter. Two oscillation frequencies are slightly different and this difference creates a beat. Then, the low-pass filter extracts the beat and it sounds through a loudspeaker. A capacitor in the circuit outside the Theremin acts as an antenna, and its capacitance changes as the player moves his/her hand to or from the antenna. This change in capacitance results in beat frequency, i.e., the Theremin's pitch.

B. Problems with Playing the Theremin

There are two problems with playing the Theremin:

1) the pitch characteristics is non-linear and
2) the pitch characteristics quickly varies according to the environment.

Due to these problems, playing music with the Theremin is difficult and needs training to play it. People who especially

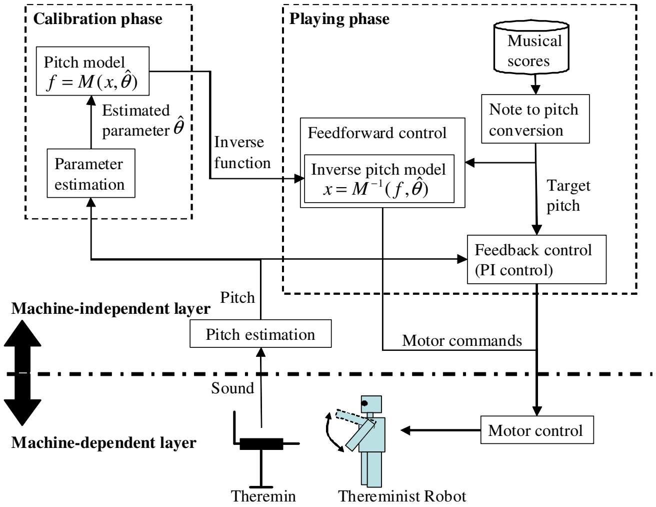

Fig. 3. Overview of a Thereminist Robot System

have no experience in playing an instrument cannot even play a required pitch exactly [8]. The feedback control was adopted to solve these problems in existing studies. However, it is not suite for playing melodies because this task has strict time constraints. Moreover, the convergence time for the current pitch to converge to the next varies due to the current pitch when only feedback control is used.

IV. Thereminist Robot System

This section describes our Thereminist robot system. Section IV-A overviews the system, Section IV-B presents two new pitch models of the Theremin, and, Section IV-C and IV-D describes the two phases of the system: the calibration and the playing phase.

A. Overview

Figure 3 shows a block diagram of the Thereminist robot system, which has two phases: the calibration and the playing phase. First, the robot records N points of the Theremin's sound in the calibration phase at the N robot's hand position x_i $(i = 0, \cdots, N-1)$. Next, the pitches of recorded sounds are estimated $f_i, (i = 0, \cdots, N-1)$, and pairs of (x_i, f_i) are used to estimate the parameters of the pitch model. Finally, the robot plays the required pitch in a coarse-to-fine manner in the playing phase. The robot moves its hand to play the pitch coarsely using the pitch model (feedforward), and adjusts its hand position finely with listening to the Theremin's sound (feedback).

The system also has two layers: machine-independent and dependent layers. To implement our system to another robot, only machine-dependent layer and the interface between two layers should be implemented for the particular hardware. We should notice that the machine-independent layer is the expertize to play the Theremin, and is not need to change to transfer to another robot.

B. Theremin's Pitch Model

This section presents the novel two Theremin's pitch models: parametric and non-parametric ones. If we can assume that the environment is stable after the calibration of the pitch model, the non-parametric model is useful because the pitch model uses "true" data themselves. However, if the environment is unstable, the parametric model is useful because the pitch model adapts the new environment faster by re-estimating the parameters. In both models, the model-based approach enables the robots to play a required pitch fast and coarsely by feedforward control.

1) Parametric Model: To describe Theremin's non-linear pitch characteristics, we constructed the following model:

$$f = M(x, \boldsymbol{\theta}) = \frac{\theta_2}{(\theta_0 - x)^{\theta_1}} + \theta_3, \tag{1}$$

where x is the distance between the antenna and a player's hand, $\boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2, \theta_3)$ are the parameters of the pitch model, M, and $f = M(x, \boldsymbol{\theta})$ is the pitch of the Theremin. The model has a constraint that $\theta_1 > 0$ to ensure taht the inverse-pitch model Eq. (18) is a real-valued function.

The pitch model is based on the following fact: The Theremin's pitch changes slightly if the player's hand and the antenna are far apart, and this change increases as the distance gets shorter. This means that the relationship between the Theremin's pitch and the distance between the hand and antenna is nonlinear. The parameters of the pitch model are estimated in the calibration phase using pairs of the robot's hand position and the pitch at the position (x_i, f_i) where $i = 1, \cdots, N-1$ (see Section IV-C for details).

2) Non-parametric Model: We can also use the non-parametric model, if we assume that the environment is stable. This pitch model has smaller error than the parametric model because it uses measured data themselves. However, if the environment changes, we need to re-measure many points (40 to 80 points in practice.) Thus, the pitch model has a fewer capability for adaptation.

Assume that pairs of (x_i, f_i) are given, we can then calculate the corresponding hand position x' for any required pitch f' with linear interpolation:

$$x' = x_l + (x_h - x_l)\,\frac{f' - f_l}{f_h - f_l}, \tag{2}$$

where (x_l, f_l) and (x_h, f_h) satisfy:

$$(x_l, f_l) = (x_{i_l}, f_{i_l}) \ \ s.t. \ i_l = \operatorname*{argmin}_{i, f_i < f'} |f' - f_i| \, and \tag{3}$$

$$(x_h, f_h) = (x_{i_h}, f_{i_h}) \ \ s.t. \ i_h = \operatorname*{argmin}_{i, f_i > f'} |f' - f_i| \, . \tag{4}$$

In other words, i_l is an index whose corresponding pitch is the closest to f' and less than f'. The i_h is calculated similarly.

C. Calibration Phase

The calibration phase consists of three steps: (1) moving the robot's hand to some positions, and recording the Theremin's sound in these positions, (2) estimating the Theremin's pitch at all positions, and (3) estimating model parameters $\hat{\boldsymbol{\theta}}$ using pairs of pitch and corresponding hand positions (x_i, f_i). (3) is not used for non-parametric model.

1) Estimation of Pitch: We used an auto-correlation-based method of estimating pitch [12]. Auto-correlation function $r(t)$ of the signal $x(t)$ describes the correlation of the signal with itself after time-lag t.

$$r(t) = \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{T/2} x(t')x(t'+t)dt' \tag{5}$$

We can obtain $r(t)$ by computing the inverse Fourier transform of the power spectrum of $x(t)$ with Wiener-Khinchin's theorem.

$$r(t) = \mathcal{F}^{-1}\left[|X(f)|^2\right], \tag{6}$$

where $X(f) = \mathcal{F}[x(t)]$ is the frequency-domain representation of the signal, and $\mathcal{F}$ means the Fourier transform and and $\mathcal{F}^{-1}$ means the inverse Fourier transform.

This method is used to find the frequency $\hat{p}$ whose reciprocal maximizes $r(t)$ and the frequency is the pitch of the signal $x(t)$. The formal expression of the pitch is:

$$\hat{p} = \underset{f<f_{\max}}{\operatorname{argmax}}\, r\left(\frac{1}{f}\right), \tag{7}$$

where $\hat{p}$ is the estimated pitch and $f_{\max}$ is the upper boundary to ensure that $\hat{p}$ is finite.

2) Estimation of Parameters: We used the least squares method to estimate the parameters of the pitch model (Eq. 1). The (f_i, x_i) $(i = 0, \cdots, N-1)$ represents the Theremin's pitchs and these correspond to a player's hand position. The evaluation function $\boldsymbol{J}(\boldsymbol{\theta})$ for the least squares is defined by:

$$\begin{aligned} J_i(x_i, f_i, \boldsymbol{\theta}) &= f_i - M(x_i, \boldsymbol{\theta}) & (8)\\ \boldsymbol{J}(\boldsymbol{\theta}) &= (J_0, \cdots, J_{N-1})^T & (9)\end{aligned}$$

The problem statement with parameter estimation is:

Problem Statement

Input: Learning data (f_i, x_i) and initial parameters $\boldsymbol{\theta}$
Output: The parameter $\boldsymbol{\theta}^*$ that minimizes the cost function $\boldsymbol{K}(\theta)$

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\, \mathbf{K}(\boldsymbol{\theta}) and \tag{10}$$

$$\mathbf{K}(\boldsymbol{\theta}) = \frac{1}{2}\boldsymbol{J}(\boldsymbol{\theta})^T\boldsymbol{J}(\boldsymbol{\theta}). \tag{11}$$

We solved this problem with the Levenberg-Marquardt (LM) method [13]. To estimate the minimizer $\boldsymbol{\theta}^*$ of non-linear function $\mathbf{K}$, the LM method calculates iteratively. The LM method uses a Jacobi matrix, which is defined by $(\mathbf{Jac}(\boldsymbol{\theta}))_{ij} = \frac{\partial}{\partial\theta_j} J_i(\boldsymbol{\theta})$. The elements of the Jacobi matrix are partial derivatives of the cost function J_i. All the kinds of elements of the Jacobi matrix are:

$$\begin{cases} \partial J_i/\partial\theta_0 = -\theta_1\theta_2(\theta_0 - x_i)^{-(\theta_1-1)}, \\ \partial J_i/\partial\theta_1 = \theta_2(\theta_0 - x_i)^{-\theta_1}\log(\theta_0 - x_i), \\ \partial J_i/\partial\theta_2 = -\frac{1}{(\theta_0 x_i)^{\theta_1}}, \\ \partial J_i/\partial\theta_3 = -1. \end{cases} \tag{12}$$

At each iteration step, Eq. 13 is solved and the parameter is updated.

$$\begin{aligned} (\mathbf{Jac}(\boldsymbol{\theta})^T\mathbf{Jac}(\boldsymbol{\theta}) + \mu\mathbf{I})\mathbf{h} &= -\mathbf{Jac}(\boldsymbol{\theta})^T\mathbf{J}(\boldsymbol{\theta}) and (13)\\ \boldsymbol{\theta}_{new} &= \mathbf{h} + \boldsymbol{\theta}, & (14)\end{aligned}$$

where $\mathbf{I}$ is a unit matrix, μ is the damper parameter and $\mathbf{h}$ is a update portion of the parameter.

To establish whether the LM method stops at a finite time, it is important to find that $\mathbf{h}$ decreases the cost function. We show that the direction of $\mathbf{h}$ decreases the cost function $\mathbf{K}$. The derivative of the cost function is $\mathbf{K}' = \mathbf{Jac}(\boldsymbol{\theta})^T\mathbf{J}(\boldsymbol{\theta})$. If $\mathbf{hK}' < 0$ then $\mathbf{h}$ decreases $\mathbf{K}$.

$$\begin{aligned} \mathbf{hK}'(\boldsymbol{\theta}) &= -\mathbf{hJac}(\boldsymbol{\theta})^T\mathbf{J}(\boldsymbol{\theta}), & (15)\\ &= -\mathbf{h}(\mathbf{Jac}(\boldsymbol{\theta})^T\mathbf{Jac}(\boldsymbol{\theta}) + \mu\mathbf{I})\mathbf{h}, and & (16)\\ &= -\mathbf{hJac}(\boldsymbol{\theta})^{\mathbf{T}}\mathbf{Jac}(\boldsymbol{\theta})\mathbf{h} - \mu\mathbf{hIh}. & (17)\end{aligned}$$

Matrices in right side are positive-definite because $\mathbf{Jac}(\boldsymbol{\theta})^T\mathbf{Jac}(\boldsymbol{\theta})$ and $\mathbf{I}$ are symmetrical. Thus, $\mathbf{hK}' < 0$ and $\mathbf{h}$ decreases the cost function.

D. Playing Phase

A robot plays the Theremin in two ways in the playing phase: feedforward control for playing quickly and feedback control for playing accurately.

1) Feedforward Control Based on the Pitch Model: The input for our pitch model was robot's hand position, and the output was the Theremin's pitch at the corresponding position. To play the Theremin with the pitch model, we needed an inverse model whose input was an pitch and output was the corresponding position. Thus, we needed an inverse function, $M^{-1}(f, \hat{\boldsymbol{\theta}})$. By simply manipulating the formula, we obtain the inverse model

$$\hat{x} = M^{-1}(f, \hat{\boldsymbol{\theta}}) = \theta_0 - \left(\frac{\theta_2}{f - \theta_3}\right)^{\frac{1}{\theta_1}}. \tag{18}$$

The input for the Thereminist robot system consists of two components: a target note number $(C = 0, C\sharp = 1, \cdots, B = 10, B\sharp = 11)$ and its octave number $(0, \cdots 8)$. The pair of a note and an octave is transformed to an pitch [Hz] by equal temperament, which is the most popular rule for transformation. The formal expression of this transform is

$$p = 440 \cdot 2^{o-4}\sqrt[12]{2^{n-9}}, \tag{19}$$

where n is the note number, o is the octave number and p is the pitch corresponding to n and o.

2) Feedback Control with a PI Controller: To play the Theremin accurately, we used a Proportional-Integral(PI) controller to adjust the pitch. The PI controller is a typical mechanism in classical control. It controls using error (the difference between the current and required output). P means it controls using error, and I means error has accumulated.

$$\begin{aligned} e(t) &= p - \hat{p} \quad and & (20)\\ u(t) &= K_p e(t) + K_i\int_0^t e(\tau)d\tau, & (21)\end{aligned}$$

where $e(t)$ is the error signal, p is the target pitch and $\hat{p}$ is the estimated pitch by using Eq. 7, K_p is proportional gain and K_i is integral gain. We need to prevent from being too sensitive because the pitch estimation occasionally fluctuates. Therefore, we did not used a PID(Proportional-Integral-Derivative) Controller , which is also typical in classical control theory.

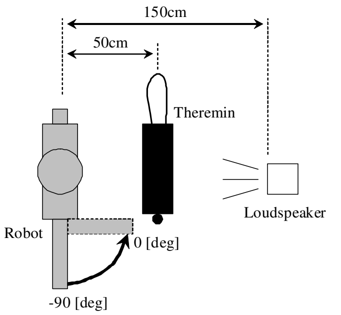

Fig. 4. Configuration for Experiments

Fig. 5. ASIMO plays the Theremin

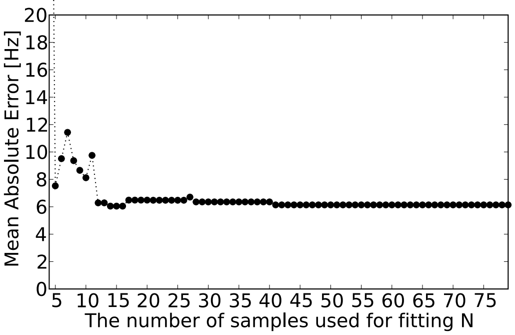

Fig. 6. Experiment 1: Mean absolute error of the model

V. Experiments

We evaluated our pitch model and the Thereminist robot system. The pitch models were evaluated by : (1) changing the number of data used for fitting and (2) changing the environment around the Theremin to demonstrate the model's robustness. Finally, the system was evaluated using musical performances by two robots to examine the possibility of expertise transfer.

A. Common Conditions

Figure 4 outlines the configuration for the experiments. We implemented the Thereminist robot system on a humanoid robot, i.e. Kawada Industries, Inc's HRP-2 Promet. The robot has eight-channel microphones on its head (we only used one microphone in front of the robot's head). We controlled the roll angle of the robot's right shoulder (See Fig. 4) for it to play the Theremin. Therefore, we replaced the hand position x in Eq. 1 with the roll angle (the range of this degree of freedom is from -90 [deg] to 0 [deg]). We used the Etherwave Theremin from Moog Music. The distance between the microphone and the loudspeaker of the Theremin was 150 [cm], and that between the robot and the Theremin was about 50 [cm]. The sound of the Theremin was recorded at a sampling rate of 48 [kHz], and frequency analysis for estimating pitch was applied at 8192 [points].

According to the Theremin's mechanism, we assume that the pitch increases as the hand moves closer. Thus, we removed the data which do not confirm to the assumption.

B. Experiment 1: Robustness to Changing Samples

We evaluated the model error by using the mean absolute error (MAE):

$$MAE = \frac{1}{N-1}\sum_{i=0}^{N}\left|\hat{f}_i - M(\hat{f}_i, \hat{\boldsymbol{\theta}})\right|, \tag{22}$$

where N is the number of recorded sounds, $\hat{f}_i$ is the estimated pitch at the ith recording and $\hat{\boldsymbol{\theta}}$ is the estimated parameter. MAE is zero for non-parametric model because $M(\hat{f}_i) = \hat{f}_i$.

To evaluate the parametric model, we measured 80 pairs of hand positions and the Theremin's pitch $(x_i, \hat{f}_i)$ (i.e., $N = 80$). Then, we estimated the parameters by changing the number of pairs used for fitting ($N = 4, \cdots, 80$). The pairs were selected with equal spacing.

Figure 6 shows the results. Obviously, the more samples were used, the more MAE decreased. However, this decrease saturates when about 12 samples are used. Thus, 12 samples are sufficient for estimating the parameters.

Also, we can see that error (about 6 [Hz]) is not disappeared even if all samples are used. Therefore, we need a feedback control to play a piece accurately.

TABLE I
Experiment 2: Estimated Parameters

Condition	θ_0	θ_1	θ_2	θ_3	MAE
0	56.78	4.65	1.70×10^{11}	79.455	7.91
1	39.36	3.97	2.73×10^{9}	192.99	3.05
2	30.20	3.42	1.44×10^{8}	279.22	3.21
3	71.17	5.86	4.39×10^{13}	807.23	10.7

C. Experiment 2: Robustness to Changing an Environment

To change the environment around the Theremin, we moved a metal box closer to it in four steps. There was no metal box in Condition 0. The box was at the farthest distance under Condition 1, at the intermediate distance under Condition 2 and the box was at the closest distance to the Theremin under Condition 4. In this experiment, we used 20 pairs of the robot's hand position and corresponding pitch to estimate the parameters (i.e., $N = 20$).

Figure 7 plots the parametric and non-parametric models, and Table I lists the estimated parameters of the parametric model. According to the figure, moving the box closer to the Theremin increases the minimum pitch but does not change the maximum pitch. This means that if there is something made of metal near the Theremin, its pitch range narrows. θ_3 shows the lowest pitch for the Theremin in that environment. As the Theremin's highest pitch (around 1400 [Hz]) does not change, we can obtain the Theremin's pitch range by using the estimated parameters.

D. Experiment 3: Expertize Transfer between Two Robots

We implemented our playing system on two robots to demonstrate it's portability: HRP-2 and ASIMO. These robots were controlled in different ways: HRP-2 was controlled by specifying its joint angles, and ASIMO is controlled by specifying its three-dimensional coordinate hand positions (task vector) [14]. The piece the robots played was Japanese nursery rhyme, the "frog song". Figure 8 shows the score for the song. Figure 9 shows the notes used in the score and corresponding pitches.

We used the non-parametric model to achieve accurate play. To apply our non-parametric model to task-vector

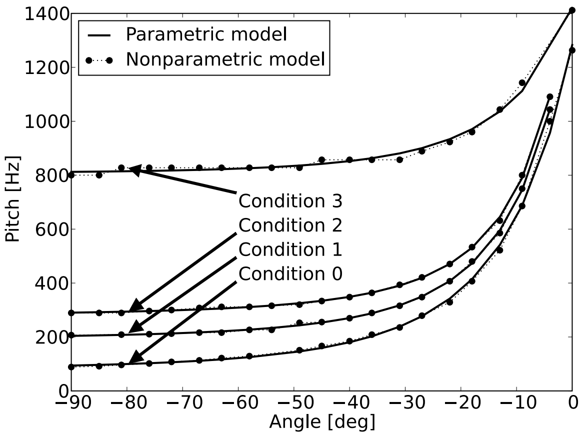

Fig. 7. Experiment 2: Fitting curves in various environments

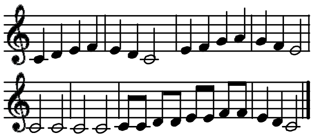

Fig. 8. Score of frog song

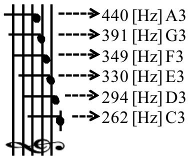

Fig. 9. Pitches of notes

control, we replaced the hand position x in Eq. 1 with the Euclid distance. We determined the nearest point $\mathbf{p}_0 = (x_0, y_0, z_0, f_0)$ and the farthest point $\mathbf{p}_1 = (x_1, y_1, z_1, f_1)$. Here, x, y, and z are the three-dimensional coordinates, and f is the Theremin's pitch at that position. The robot moves its hand along a line that links two points $\mathbf{p}_0$ and $\mathbf{p}_1$. Task vector $\mathbf{p} = (x, y, z)$ satisfies:

$$\frac{x - x_0}{x_1 - x_0} = \frac{y - y_0}{y_1 - y_0} = \frac{z - z_0}{z_1 - z_0}, \tag{23}$$

where $x_0 < x < x_1$, $y_0 < y < y_1$ and $z_0 < z < z_1$. We divided the line into N equally in the calibration phase and recorded the Theremin's pitch at all points.

Figure 10 shows the pitch trajectories for the song played by HRP-2 and ASIMO. The upper trajectory is generated by ASIMO, and the lower one is generated by HRP-2. The fluctuations were caused by miss-estimation of the pitch. The upper trajectory is curvilinear, and ASIMO's body motion is smoothly and human-like. In contrast, because we used only one angle of HRP-2, the lower on is linear, and its motion is unnatural and robot-like. The curvilinear trajectory shows that the audience is difficult to distinguish what note the robot plays. Thus, the linear trajectory shows the better performance. Therefore, ASIMO's play has better physical appearance and HRP-2's play has better auditory appearance. This result shows that there is a trade-off between physical and auditory appearance.

VI. Conclusions

This paper proposed a Thereminist robot system and two Theremin's pitch models. According to experimental result 1, the pitch model is capable of describing the pitch characteristics with small amount of samples. Experimental result 2 showed that the model can describe the characteristics even if the electromagnetic field is changed. The system has machine-dependent and independent layers. As the third experimental result shows, we can transfer the expertize by modifying machine-dependent layer and the interface between the layers. The model-based playing system enabled the robot to play a melody which has strict time constraints in coarse-to-fine manner. After the calibration phase, the Thereminist robot can play any score if all note in the score is in the Theremin's pitch range.

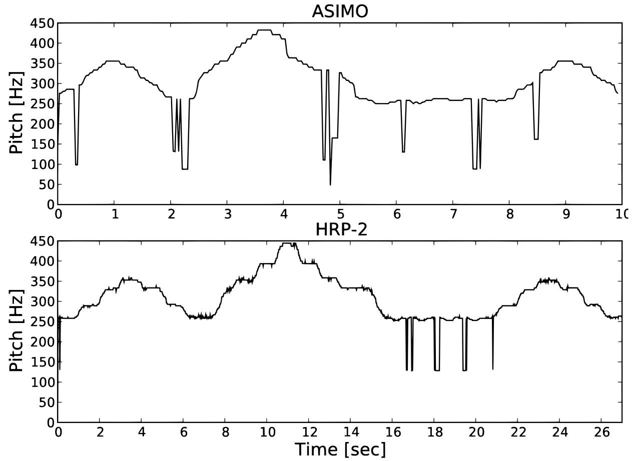

Fig. 10. Experiment 3: Trajectories of pitch played by HRP-2 and ASIMO

In future work, we will build a Theremin's volume model for entire Thereminist system and implement a special kind of Theremin play, e.g., *vibrato*.

VII. Acknowledgements

A part of this study was supported by Grant-in-Aid for Scientific Research (S) and Global COE Program.

References

[1] T. Mizumoto *et al.* A robot listens to music and counts its beats aloud by separating music from counting voice. *IROS08*, pages 1538–1543.

[2] A. Turing. Computing machinery and intelligence. *Mind*, LIX(235):433–460, October 1950.

[3] J. Solis *et al.* Development of Waseda flutist robot WF-4RIV: Implementation of auditory feedback system. *ICRA08*, pages 3654–3659.

[4] G. Weinberg *et al.* The interactive robotic percussionist - new deveopments in from, mechanics, perception and interaction design. *HRI07*, pages 456–461.

[5] K. Yoshii *et al.* A biped robot that keeps steps in time with musical beats while listening to music with its own ears. *IROS07*, pages 1743–1750.

[6] K. Murata *et al.* A robot uses its own microphone to synchronize its steps to musical beats while scatting and singing. *IROS08*, pages 2459–2464.

[7] S. Kotosaka and S. Shaal. Synchronized robot durumming by neural oscillator. *RSJ*, 19(1):116–123, 2001.

[8] A. Alford *et al.* A music playing robot. *FSR99*, pages 29–31.

[9] S. Nakaoka *et al.* Task model of lower body motion for a biped humanoid robot to imitate human dances. *IROS05*, pages 2769–2774.

[10] T. Nakata, T. Mori, and T. Sato. Analysis of impression of robot bodily expression. *Journal of Robotics and Mechatronics*, 14(1):24–36, 2002.

[11] A. V. Glinsky. *The Theremin in the Emergence of Electronic Music*. PhD thesis, New York Univ., 1992.

[12] A. Camacho. *SWIPE: A sawtooth waveform inspired pitch estimator for speech and music*. PhD thesis, University of Florida, 2007.

[13] K. Madsen *et al.* Methods for non-linear least squares problems (2nd ed.). Informatics and Mathematical Modeling, DTU, 2004.

[14] M. Toussaint *et al.* Optimization of sequential attractor-based movement for compact behaviour generation. *Humanoids07*.