Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Imitative Motion Generation for Humanoid Robots based on the Motion Knowledge Learning and Reuse

Yuki Okuzawa
Nagoya Institute of Technology
Nagoya, Japan
okuzawa@juno.ics.nitech.ac.jp

Shohei Kato
Nagoya Institute of Technology
Nagoya, Japan
shohey@juno.ics.nitech.ac.jp

Masayoshi Kanoh
Chukyo University
Nagoya, Japan
mkanoh@sist.chukyo-u.ac.jp

Hidenori Ito
Nagoya Institute of Technology
Nagoya, Japan
itoh@juno.ics.nitech.ac.jp

***Abstract*—A knowledge-based approach to imitation learning of motion generation for humanoid robots and an imitative motion generation system based on motion knowledge learning and reuse are described. The system has three parts: recognizing, learning, and modifying parts. The first part recognizes an instructed motion distinguishing it from the motion knowledge database by the hidden markov model. When the motion is recognized as being unfamiliar, the second part learns it using dynamical movement primitives and acquires a knowledge of the motion. When a robot recognizes the instructed motion as familiar or judges that its acquired knowledge is applicable to the motion generation, the third part imitates the instructed motion by modifying a learned motion. This paper reports some performance results: the motion imitation of several radio gymnastics motions.**

***Index Terms*—Imitation Learning, Hidden Markov Model, Modifying Learned Motion**

I. Introduction

Imitation learning is used for learning techniques in humanoid robots in many robotics research fields. Imitation learning is a technique where a robot acquires motions by imitating the instructed motions of an instructor [1] [2] [3]. The robot's learning of the instructed motions that equal a learned motion is inefficient learning. If an instructed motion that equals a learned motion is given, the robot would efficiently imitate the instructed motion by reusing the learned motion without any learning. The instructed motion is imitated by appropriately modifying the learned motion. The robot requires artificing to make its knowledge of a learned motion and to reuse partially a learned motion as needed.

Several related studies on robots acquiring knowledge and reusing a learned motion in imitation learning have been conducted. Inamura [4] enabled a robot to recognize a learned motion with such acquired knowledge. In this case, the knowledge is symbolized a full body motion of a humanoid robot by a hidden markov model (HMM). However, his approach did not recognize these motions as learned motions: upper body motions familiar to a learned motion and lower body motions unfamiliar to all learned motions. Samejima [5] proposed an imitation learning framework with MOSAIC reinforcement learning. A robot learns efficiently by combining learned motion data to fit an instructed motion, generating an imitative motion. However, this learning approach postulates that the robot has an identical system of "teacher". There is some doubt on the definition of imitation learning, and it is unclear what this approach applies to imitation learning for a humanoid robot. Ito [6] has shown an observed motion can be recognized and generated by a robot by learning an instructed motion with recurrent neural network parametric biases. Generable motion is limited to a simple motion that the robot flaps its arms, and a learned motion is not properly reused in this approach. Nakanishi [7] et al. proposed imitation learning using dynamical movement primitives, stating that it could modify a learned motion to fit a different environment. But, a learned motion is not reused for imitating a different motion.

In this research, we created a mechanism where a robot recognizes an instructed motion when the motion is similar to the robot's learned motion, and it acquires an imitative motion by modifying a learned one. We describe an information-processing model that gives a robot these two abilities:

- The ability to acquire motion knowledge by recognizing and learning an instructed motion
- The ability to imitate a motion by reusing motion knowledge

In this paper, the robot symbolizes an instructed motion that divides a full body motion by each joint, recognizes the instructed motion as familiar, or judges it using the HMM. In addition, the robot learns generating motion data that can be changed to a different motion by the dynamical movement primitives, and modifies a learned motion to fit the instructed motion using familiar information that is calculated using the HMM. We created an approach to imitation learning of motion generation that acquires symbols and motion data as motion knowledge and reuses motion knowledge to generate and modify an imitative motion.

II. HMM and Dynamical Movement Primitives

A. HMM

A HMM is a well-known tool that is a recognition technique of longitudinal data, and it is often used in voice recognition. We used the left-to-right HMM (as shown in Fig. 1) for recognizing instructed motions. The HMM consists of a finite set of states, $\mathcal{S} = \{S_1, \cdots, S_N\}$, a finite set of output motion elements, $\mathcal{U} = \{u_1, \cdots, u_{N_q}\}$, state transition probabilities $\mathcal{A} = \{a_{ij}\}$ that transit from state S_i to state S_j, and output probabilities $\mathcal{B} = \{b_{S_i}(u_j)\}$ that output motion element u_j on state S_i.

978-1-4244-2794-9/09/$25.00 ©2009 IEEE

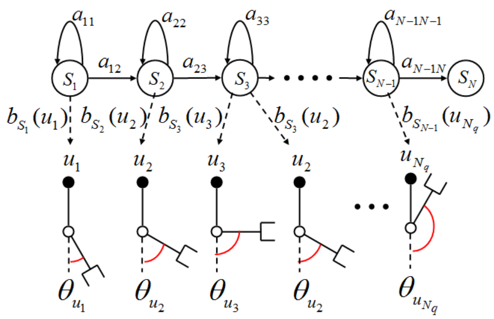

Fig. 1. HMM in our recognition module

B. Motion Elements

Motion element u_i is quantized data of the joint trajectory over time. Where the number of quantizations is set at N_q, a set of motion angle elements are determined using $\mathcal{D} = \{\theta_{u_1}, \cdots, \theta_{u_{N_q}}\}$. θ_{u_i} is defined by the following equation:

$$\theta_{u_i} = \theta_{min} + \frac{(\theta_{max} - \theta_{min})i}{N_q - 1}, \tag{1}$$

where θ_{min} is the minimum angle of the range of motion, and θ_{max} is the maximum angle of the range of motion. We set the quantization number, N_q, to an appropriate value beforehand. The HMM can be applied to motion recognition by converting the joint trajectory of the instructed motion into the motion element vector.

C. Motion Symbols

The motion element, u_i, that the HMM outputs depends on two parameters: the state transition probability, a_{ij}, and the output probability, $b_{S_i}(u_j)$. The HMM indicates a motion when the state transition probabilities, $\mathcal{A} = \{a_{ij}\}$, and the output probabilities, $\mathcal{B} = \{b_{S_i}(u_j)\}$, are as shown in Fig. 1. Thus, we define the motion symbol as follows:

$$\lambda \stackrel{\text{def}}{=} \{\mathcal{A}, \mathcal{B}\}. \tag{2}$$

When the motion element vector, $O = [o_1, \cdots, o_T](o_i = u_{k_i}, k_i \in \{1, \cdots, N_q\})$, and time frame T are given, the motion symbol is learned by estimating $\mathcal{A}$ and $\mathcal{B}$ to maximize the output probability of O by the Baum-Welch algorithm [9].

D. Dynamical Movement Primitives

The dynamical movement primitives is a dynamical system to learn and generate a motion by using a limit cycle oscillator, and it can easily modify a learned motion by scaling the parameters. Consider the following limit cycle oscillator characterized in terms of an amplitude r and a phase ϕ as a canonical dynamical system that generates basic rhythmic patterns:

$$\tau\dot{\phi} = 1, \quad \tau\dot{r} = -\mu(r - r_0), \tag{3}$$

where τ is the temporal scaling factor, r_0 determines the desired (relative) amplitude, and μ is a positive constant. This rhythmic canonical system is designed to provide an amplitude signal, $\tilde{v} = [r\cos\phi, r\sin\phi]^T$, and phase variable $mod(\phi, 2\pi)$ to the following second order dynamical system (z, y), where output y is used as the desired trajectory for the robot.

$$\tau\dot{z} = \alpha_z(\beta_z(y_m - y) - z) \quad \tau\dot{y} = z + f(\tilde{v}, \phi), \tag{4}$$

where α and β are the time constants, and y_m is an offset of the output trajectory. f is a nonlinear function approximater using locally linear models [10] of the form

$$f(\tilde{v}, \phi) = \frac{\sum_{i=1}^{N} \psi_i w_i^T \tilde{v}}{\sum_{i=1}^{N} \psi_i}. \tag{5}$$

Each local model ψ_i is weighted by a Gaussian kernel function, where w_i is the parameter vector of the k-th local model that is determined by locally weighted learning [10] from a trajectory of the instructed motion. The imitative motion that is generated is decided by a set of parameters $W = \{w_i\}$. The amplitude, frequency, and offset of the learned rhythmic patterns can be easily modified by scaling the parameters r_0, τ, $Y_m = \{y_{m\,i}\}$ individually.

In this paper, we define primitive data ξ as follows:

$$\xi \stackrel{\text{def}}{=} \{W, r_0, \tau, Y_m\}. \tag{6}$$

III. Imitative Motion Generation System

In this research, the imitative robot could recognize, learn, and modify the motions by dividing the instructed full body motion into each joint. The learned motions accumulate in the database by making them the motion knowledge.

The accumulated motion knowledge is used for recognizing, generating, and modifying an instructed motion in the next imitation learning. The motion database consists of the motion knowledge database and the modifying database. The motion knowledge database holds the learning data, $DB_K = \{(\lambda_X, \xi_X), \cdots\}$, that are a combination set (λ_X, ξ_X) of the motion symbol and the primitive data of motion X. The modifying database holds the modifying data, $DB_M = \{\xi_{X\to Y}, \cdots\}$, that are a set of primitive data $\xi_{X\to Y}$ that modify learned motion X to fit motion Y. In this study, we homologized between the instructor and the imitative robot for each joint. Afterwards, we describe the joint trajectory unless it is otherwise noted. Fig. 2 shows an overview of the imitative motion generation system based on learning and reusing motion knowledge. This system has three parts: recognizing, learning, and modifying parts. The first part recognizes an instructed motion, the second part learns the instructed motion and generates the imitative motion, and the third part generates the imitative motion by modifying a learned motion.

The robot observes the time frame, T_A, and the joint trajectory, $Pos_A = [pos_{A\,1}, \cdots, pos_{A\,T_A}]$, of motion A when the instructor poses instructed motion A, and it imitates the instructed motions as follows.

A. Recognizing Part

Transform Joint Trajectory into Motion Element Vector: Joint trajectory Pos_A is transformed into the motion element

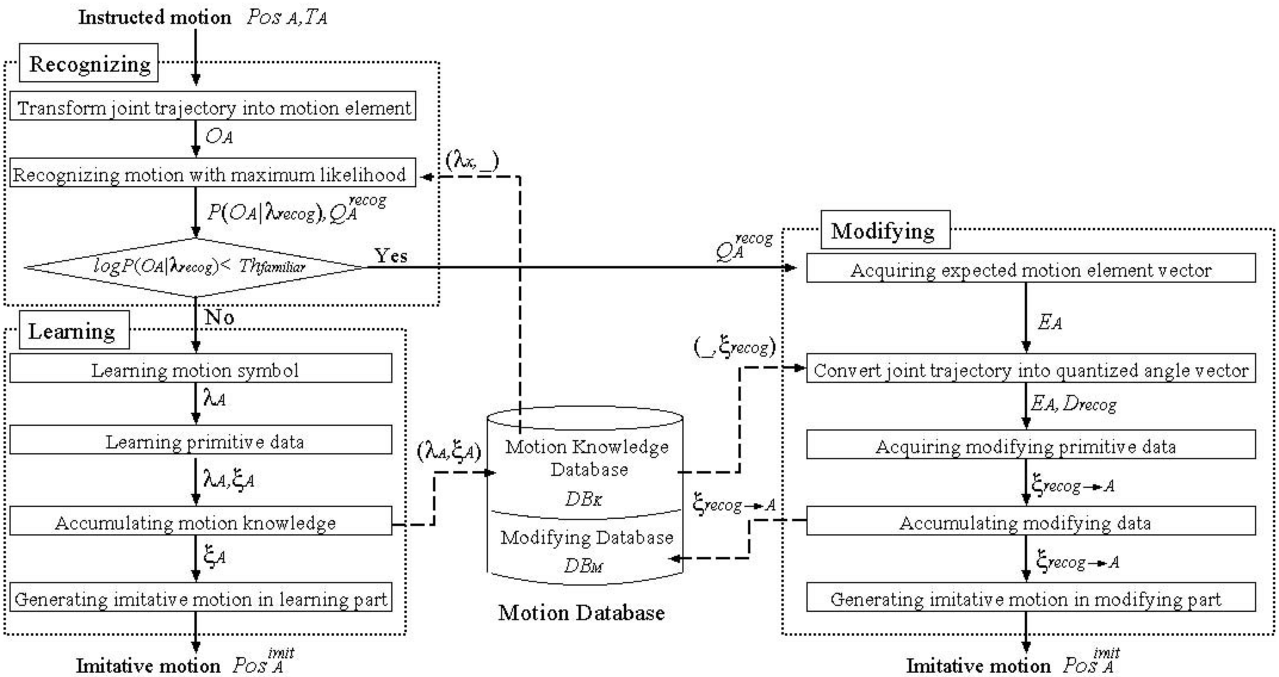

Fig. 2. Overview of our system

vector, $O_A = [o_{A\,1}, \cdots, o_{A\,T_A}]$, as the following equation for recognizing the instructed motion A with the HMM:

$$o_{A\,i} = \arg\min_{u_i \in \mathcal{U}} |\theta_{u_i} - pos_{A\,i}|. \tag{7}$$

Recognizing Motion with Maximum Likelihood: Likelihood $P(O_A|\lambda)$ that is calculated by the motion element vector, O_A, and the learned motion symbol, λ, is used for the motion recognition. The more instructed motion A is familiar to the motion that is indicated by λ, the more the value of $P(O_A|\lambda)$ grows. The motion symbol of the most familiar motion to A is requested from all the motion symbols in the motion knowledge database as follows:

$$\lambda_{recog} = \arg\max_{(\lambda_x, \_) \in DB_K} P(O_A|\lambda_x). \tag{8}$$

The motion $recog$ that indicates the maximum likelihood is the most familiar motion to A in all learned motions. The eq. (8) are calculated using a Viterbi algorithm [9]. At the same time, this calculation computes the maximum likelihood state vector $Q_A^{recog} = [q_{A\,1}^{recog}, \cdots, q_{A\,T_A}^{recog}]$.

Judged Threshold Value Based on Motion Knowledge : In this phase, the motion knowledge that is acquired in the recognizing part is determined whether it is used for imitating instructed motion A or is not used by the threshold value, $Th_{familiar}$.

- $\log P(O_A|\lambda_{recog}) < Th_{familiar}$: go to the learning part
- $\log P(O_A|\lambda_{recog}) \geq Th_{familiar}$: go to the modifying part

Note that $Th_{familiar}$ is set to an appropriate value beforehand.

B. Learning Part

Learning Motion Symbol: Motion symbol λ_A of motion A that maximizes the output probability of motion element vector O_A is learned using the Baum-Welch algorithm. Also, the motion symbol is used for motion recognition during the next imitation learning.

Learning Primitive Data: Primitive data $\xi_A = \{W^A, r_0^A, \tau^A, Y_m^A\}$ of motion A are acquired from the joint trajectory, Pos_A, of instructed motion A by the dynamical movement primitives, as described in section II-D.

Accumulating Motion Knowledge: The combination of λ_A and ξ_A of motion A are accumulated in the motion knowledge database, DB_K.

$$DB_K = DB_K \cup \{(\lambda_A, \xi_A)\}. \tag{9}$$

Generating Imitative Motion in Learning Part: The dynamical movement primitives that are plugged in ξ_A generate joint trajectory Pos_A^{imit} of imitative motion A.

C. Modifying Part

Acquiring Expected Motion Element Vector: The expected motion element vector, $E_A = [e_{A\,1}, \cdots, e_{A\,T_A}]$, is calculated from output probability $b_{q_{A\,i}^{recog}}(u_j) for\ all\ q_{A\,i}^{recog} \in Q_A^{recog}$ and the maximum likelihood state vector, Q_A^{recog}, which is generated in the recognizing part, as the follow equation:

$$e_{A\,i} = \sum_{u_j \in \mathcal{U}} b_{q_{A\,i}^{recog}}(u_j)\theta_{u_j}. \tag{10}$$

$E_A = [e_{A\,1}, \cdots, e_{A\,T_A}]$ becomes the target value to modify learned motion $recog$ to fit instructed motion A.

Convert Joint Trajectory into Quantized Angle Vector: First, primitive data $\xi_{recog} = \{W^{recog}, r_0^{recog}, \tau^{recog}, Y_m^{recog}\}$ are taken out of the motion knowledge, λ_{recog}, that is familiar to instructed motion A. Joint trajectory Pos_{recog}^{imit} is generated by the dynamical movement primitives that are plugged in the primitive data ξ_{recog}. Next, joint trajectory Pos_{recog}^{imit} is converted into the quantized angle vector, $D_{recog} = [d_{recog\,1}, \cdots, d_{recog\,T_A}]$, as follows:

$$d_{recog\,i} = \arg\min_{\theta_{u_j} \in \mathcal{D}} |\theta_{u_j} - pos_{recog\,i}^{imit}|. \quad (11)$$

D_{recog} is the angle vector that is quantized from the joint trajectory of the motion $recog$ based on the motion knowledge. The imitative motion that is modified by reusing the motion knowledge is generated by achieving D close to the expected motion element vector E_A.

Acquiring modifying primitive data: Learned primitive data ξ_{recog} are varied with the primitive data $\xi_{recog \to A}$ by calculating the difference in the expected motion element vector E_A and the quantized angle vector D_{recog}. The learned motion $recog$ can be modified to fit instructed motion A using $\xi_{recog \to A}$. $\xi_{recog \to A}$ is calculated using the following equations:

$$\begin{aligned} \xi_{recog \to A} &= \{W^{recog}, r_0^{recog}, \tau^{recog \to A}, Y_m^{recog \to A}\} && (12) \\ y_{m\,i}^{recog \to A} &= e_{A\,i} - d_{recog\,i} && (13) \\ \tau^{recog \to A} &= T_A / T_{recog}. && (14) \end{aligned}$$

Accumulating modifying data: Primitive data $\xi_{recog \to A}$ that are acquired by modifying the learned motion $recog$ to fit the instructed motion A are accumulated in the modifying database, DB_M.

$$DB_M = DB_M \cup \{(\xi_{recog \to A})\}. \quad (15)$$

Generating Imitative Motion in Modifying Part: Joint trajectory Pos_A^{imit} of imitative motion A is generated by the dynamical movement primitives that are plugged in primitive data $\xi_{recog \to A}$.

IV. EXPERIMENTS

We conducted experiments with KHR-2HV to test the effectiveness of the imitative motion generation system. Fig. 3 is a photograph and model of the KHR-2HV. The KHR-2HV has seventeen degrees of freedom (DOF): the neck has one DOF, each side of the arm has three DOF, and each side of the leg has five DOF.

A. Imitation Learning by Our System

In this experiment, we used the KHR-2HV as the instructor and as an imitative robot. We prepared seven kinds of full body motions $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$, $\boldsymbol{E}$, $\boldsymbol{F}$, and $\boldsymbol{G}$ as instructed motions (as shown Table I), and imitation learning was conducted in the order of $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$, $\boldsymbol{E}$, $\boldsymbol{F}$, and $\boldsymbol{G}$. Fig. 4 shows each instructed motion. We selected the instructed motions from motions included in "Radio Gymnastics," which does not contain twist movements of waist and jumping movements. ("Radio Gymnastics" are typical motions for maintaining and improving one's fitness level, and it is done by people of all ages in Japan.) The order of learning equals the order of motions that appeared in the "Radio Gymnastics." We set the state number to $N = 20$, the quantization number to $N_q = 10$, and the threshold value to $Th_{familiar} = -100$.

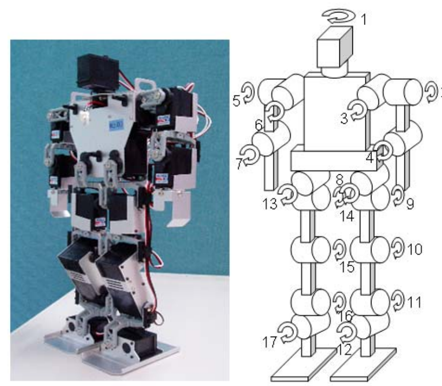

Fig. 3. Humanoid robot KHR-2HV

TABLE I
INSTRUCTED MOTIONS

Symbol	Motion
$\boldsymbol{A}$	overstretch oneself
$\boldsymbol{B}$	flexural elongation of legs and arms
$\boldsymbol{C}$	swing one's arms
$\boldsymbol{D}$	arch one's back
$\boldsymbol{E}$	lean to the right
$\boldsymbol{F}$	lean to the left
$\boldsymbol{G}$	extend one's arms above and below

Fig. 5 shows the imitative motions that were acquired from the instructed motions. Table II shows the motion knowledge of each joint that was used to generate the imitative motions of the full body. The row label shows the joint number, and the column label shows the instructed motion of the full body. Symbol "A" shows acquired motion A of the imitative robot by new learning, the symbol "$A \to B$" shows acquired motion B of the imitative robot by modifying learned motion A. The learning cost is recorded in the lowest row in the table, indicating the amount of motion knowledge that is acquired by the learning part to imitate each of the instructed motions. These results show that the imitator greatly reduces the learning cost by generating imitative motions by modifying a learned motion after learning full body motion $\boldsymbol{A}$. Because the total acquired motion knowledge is 57, the that imitative robot can imitate seven kinds of instructed motions in this experiment by acquiring only 57 types of motion knowledge.

As an example of the modifying motion, Fig. 6 shows it by learned motion B when imitating motion G is in joint number 3. The imitative robot recognized an instructed motion that was similar to a learned motion and generated a joint trajectory by modifying the learned motion.

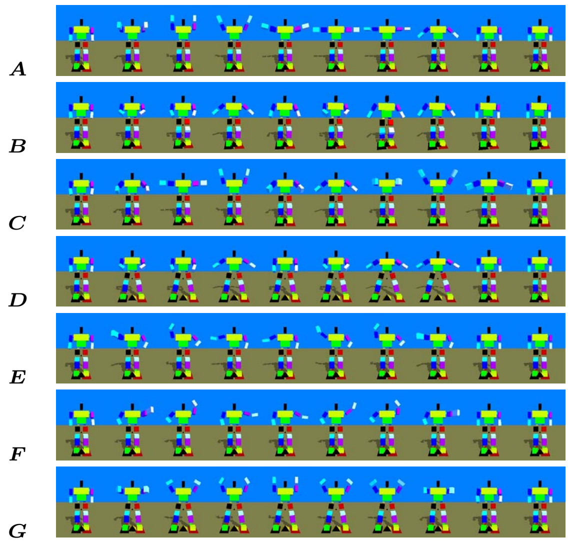

Fig. 4. Instructed motions at 500 msec intervals

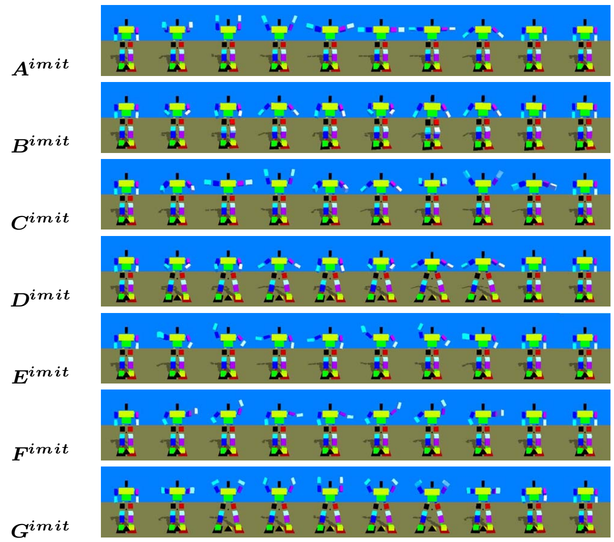

Fig. 5. Imitative motions at 500 msec intervals

TABLE II
MOTION KNOWLEDGE LEARNING AND REUSE

	$\boldsymbol{A}$	$\boldsymbol{B}$	$\boldsymbol{C}$	$\boldsymbol{D}$	$\boldsymbol{E}$	$\boldsymbol{F}$	$\boldsymbol{G}$
Joint 1	A	$A \to B$	$A \to C$	$A \to D$	$A \to E$	$A \to F$	$A \to G$
Joint 2	A	B	C	$B \to D$	E	F	G
Joint 3	A	B	C	D	$B \to E$	F	$B \to G$
Joint 4	A	B	C	$B \to D$	E	F	G
Joint 5	A	B	C	$B \to D$	E	F	G
Joint 6	A	B	C	D	E	$B \to F$	$B \to G$
Joint 7	A	B	C	$B \to D$	E	F	G
Joint 8	A	$A \to B$	$A \to C$	D	$A \to E$	$A \to F$	$D \to G$
Joint 9	A	B	$A \to C$	D	$A \to E$	$A \to F$	$A \to G$
Joint 10	A	B	$A \to C$	$A \to D$	$A \to E$	$A \to F$	$A \to G$
Joint 11	A	B	$A \to C$	$A \to D$	$A \to E$	$A \to F$	$A \to G$
Joint 12	A	$A \to B$	$A \to C$	D	$A \to E$	$A \to F$	$D \to G$
Joint 13	A	$A \to B$	$A \to C$	D	$A \to E$	$A \to F$	$D \to G$
Joint 14	A	B	$A \to C$	D	$A \to E$	$A \to F$	$A \to G$
Joint 15	A	B	$A \to C$	$A \to D$	$A \to E$	$A \to F$	$A \to G$
Joint 16	A	B	$A \to C$	$A \to D$	$A \to E$	$A \to F$	$A \to G$
Joint 17	A	$A \to B$	$A \to C$	D	$A \to E$	$A \to F$	$D \to G$
Learning Cost	17	12	6	8	5	5	4

B. Determining Motion Errors

We evaluated the errors between the instructed motions and the imitative motions to determine the appropriate modifying learned motions. Motion error err_X of motion X is defined by the following equation:

$$err_X = \frac{1}{T_X} \sum_{1}^{T_X} |pos_{X\,t} - pos_{X\,t}^{imit}|, \tag{16}$$

where T_X is the time frame of motion X, $pos_{X\,t}$ is the degree of instructed motion X at time t, and $pos_{X\,t}^{imit}$ is the degree of imitative motion X at time t.

Fig. 7 shows the motion errors of the imitative motions in the experiment described in section IV-A. We evaluated only the driving joint in this analysis. The symbols of the horizontal axis show the motions that were generated with each joint when the instructed motion was given. For example, $D2$ is the motion of joint number 2 when instructed motion D was given. In addition, for comparison, the figure shows the motion error in the case of learning for comparison. The results, were that when we modified the learned motion, motion error was larger than the motion error when learned motion was not modified, but motion error did not exceed 4 degrees. As well, the motion error when our method was used was kept within the allowable range because the maximum value of the motion error that was caused by not modifying the learned motion but by new learning was 4.7 degrees (see $C3$ in Fig. 7).

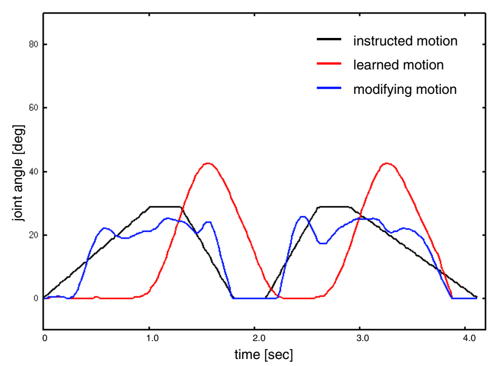

Fig. 6. Joint trajectory of modifying motion

C. Evaluate Difference in Order of Learning

The modifying motion depends on the acquired motion knowledge when a robot imitates an instructed motion. The learning and modifying motion are varied by the acquired motion knowledge even if a robot imitates the same instructed motion. It is depends on the robot's pre-acquired knowledge. To examine the variation, we changed the order of four kinds of full body motion, $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{D}$, and $\boldsymbol{G}$ (see Table I), and instructed the robot these motions. In this experiment, we lim-

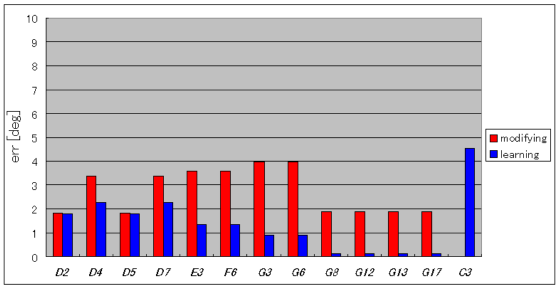

Fig. 7. Motion error of modifying motions

TABLE III
LEARNING COST AND MOTION ERROR IN ORDER OF LEARNING

Pattern	1st	2nd	3rd	4th	Cost	Error
1	A	B	D	G	41	203.3
2	A	B	G	D	43	193.0
3	A	D	B	G	41	203.5
4	A	D	G	B	41	203.5
5	A	G	B	D	45	175.4
6	A	G	D	B	43	193.2
7	B	A	D	G	39	261.8
8	B	A	G	D	41	251.5
9	B	D	A	G	39	261.8
10	B	D	G	A	39	261.8
11	B	G	A	D	41	251.5
12	B	G	D	A	41	251.5
13	D	A	B	G	41	203.5
14	D	A	G	B	41	203.5
15	D	B	A	G	39	262.0
16	D	B	G	A	39	262.0
17	D	G	A	B	41	203.5
18	D	G	B	A	41	203.5
19	G	A	B	D	45	175.4
20	G	A	D	B	43	193.2
21	G	B	A	D	45	175.4
22	G	B	D	A	45	175.4
23	G	D	A	B	43	193.2
24	G	D	B	A	43	193.2

ited the learning motions to four kinds to avoid combinational explosion.

Table III shows the learning cost and the summation of all joint motion errors for each pattern. The summation of motion error increased when the learning cost decreased, and the summation of the motion error decreased when the learning cost increased. These results were caused by the motion error of the modifying motion increasing beyond the motion error of the learning motion. Patterns 7, 9, 10, 15, and 16 are the least learning cost in order of the four kinds of full body motions. A common characteristic of them is that $\boldsymbol{B}$ is learned before both $\boldsymbol{A}$ and $\boldsymbol{G}$, and $\boldsymbol{D}$ is learned before $\boldsymbol{G}$. Thus, this common characteristic of order facilitates learning four kinds of full body motions efficiently. This is the best result on condition that the learning motions are limited to $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{D}$, and $\boldsymbol{G}$. However, the best result is not guaranteed when a different instructed motions are additionally given. A lack of motion knowledge makes imitation learning inefficient. This experimental result say that there is an effective order of learning motions in the limited motions.

V. CONCLUSIONS

This paper introduced a new imitative motion generation system based on motion knowledge learning and reuse by the creation of two abilities: the ability to acquire motion knowledge by recognizing and learning instructed motions, and the ability to imitate unknown motions by reusing motion knowledge. We imitated several full body motions to test the effectiveness of our system. The results, were that our system reduced the learning cost by recognizing and modifying a familiar motion in a learned motion. We evaluated motion error to determine the appropriate modifying learned motions.

However, the motion imitation between robots with different dynamics is not guaranteed in our proposed system. We will extend our system using David B. Grimes's research [11] as reference, they presented an approach to motion imitation between different dynamics based on dynamic Bayesian nets.

Research on evaluation of similarity of motion [12] indicated that humans focus their attention on a fingertips when they imitate motions below the shoulder. For this reason, we postulate that the acceptable range of motion error differs according to each joint. The evaluation of modifying the motion requires consideration of not only the unit of the joint but also the relationship between two or multiple joints.

We will evaluate our system by varying the number of quantizations and the threshold values, and we conduct an evaluation involving familiar motions with a subjective assessment.

ACKNOWLEDGMENTS

This work was supported in part by the Ministry of Education, Science, Sports and Culture, Grant–in–Aid for Scientific Research under grant #17500143 and #20700199, and by the Tatematsu Foundation.

REFERENCES

[1] M. Mataric: "Getting humanoids to move and imitate.", IEEE Intelligent Systems, pp. 18-24, 2000.
[2] S. Shaal: "Is imitation learning the way to humanoid robots?", Trends in Cognitive Sciences, vol. 3, no. 6, pp. 233-242, 1999.
[3] A. Alissandrakis, C. L. Nehaniv, K. Dautenhahn: "Learning How to Do Things with Imitation", American Association for Artificial Intelligence, pp. 1-6, 2000.
[4] T. Inamura:"Embodied Symbol Emergence based on Mimesis Theory", International Journal of Robotics Research, Vol. 23, No. 4, 2004.
[5] K. Samejima, K. Doya, K. Kawato: "MOSAIC Reinforcement Learning Architecture: Symbolization by Predictability and Mimic Learning by Symbol", Journal of Robotics Society of Japan, vol. 19, No. 5, pp. 551-556, 2001 (in Japanese).
[6] M. Ito, J. Tani: "On-line Imitative Interaction with a Humanoid Robot Using a Dynamic Neural Network Model of Model of a Mirror System", Adaptive Behavior, vol. 12, no. 2, pp. 93-115, 2004.
[7] J. Nakanishi: "Learning from Demonstration and Adaptation of Biped Locomotion with Dynamical Movement Primitives", IEEE-RAS/RSJ International Conference on Humanoid Robots, pp. 919-924, 2004
[8] J. J. Odel: "The use of context in large vocabulary speech recognition", PhD dissertation, Cambridge, University, 1995.
[9] S. Young et al. :The HTK Book. Microsoft Corporation, 2000
[10] S. Schaal et al. : "Constructive Incremental Learning from Only Local Information", Neural Computation, Vol. 10, No. 8, pp. 2047-2084, 1998.
[11] David B. Grimes, Daniel R. Rashid, Rajesh P. N. Rao, "Learning Nonparametric Models for Probabilistic Imitation", Advances in Neural Information Processing Systems, pp. 521-528, 2007.
[12] M. J. Mataric et al. : "Fixation behavior in observation and imitation of human movement", Cognitive Brain Research, pp. 191-202, 1998.