Pos-3

Developmental Human-Robot Imitation Learning with Phased Structuring in Neuro Dynamical System

Keita Mochizuki, Harumitsu Nobuta, Shun Nishide, Hiroshi G. Okuno, and Tetsuya Ogata

Abstract— **This paper mainly deals with influences of teaching style and developmental processes in learning model to the acquired representations (primitives). We investigate these influences by introducing a hierarchical recurrent neural network for robot model, and a form of motionese (a caregiver's use of simpler and more exaggerated motions when showing a task to an infants). We modified a Multiple Timescales Recurrent Neural Network (MTRNN) for robot's self-model. The number of layers in the MTRNN increases according to learn complex events. We investigate our approach with a humanoid robot "Actroid" through conducting an imitation experiment in which a human caregiver gives the robot a task of pushing two buttons. Experiment results and analysis confirm that learning with phased teaching and structuring enables to acquire the clear motion primitives as the activities in the fast context layer of MTRNN and to the robot to handle unknown motions.**

I. INTRODUCTION

Imitation learning is considered to be one of the most promising approaches for creating a consistently developing robot. With imitation learning, even an ordinary person can easily teach a robot any task. Moreover, a robot that can learn a task by imitating motions a human showed it is valued even in the engineering field because it is hard and costly to perfectly control a robot's motions by hand.

We can get a key to imitation learning in development from how infants learn tasks in interaction with their parents. Studies on imitation in cognitive science and developmental psychology mainly focus on a caregiver or interaction between a caregiver and learner, rather than considering the problem as merely a robot simple substance such as do most existing studies on robots imitation learning. Good examples of approaches used in human developmental learning include "scaffolding" and "motionese" as mentioned detail in Section II.

There are two important factors in robot learning with a caregiver respect to scaffolding and motionese:

1) Phasing from lower learning to upper learning
2) Simultaneous design of caregiver and learner

Most existing studies on phased learning such as 1) have not considered 2), that is, interaction between a teacher and learner, despite the availability of knowledge on scaffolding and motionese. Our goal is to construct a developmental learning environment where a learner develops and a caregiver changes his ways of showing things in accordance with the phase of the learner's development.

In terms of the specific phases a caregiver should go through in teaching a task to a robot, our key idea is primitive motions or "primitives", the basic unit of sequential motion. Motionese is closely related to primitives as it plays a role in clarifying pauses in primitives performed in a sequential motion. When inducing learners to imitate motions, there are two inevitable factors that caregivers must keep in mind, (1) the restrictions of their own physical structure and capabilities and (2) the differences between their own physical structure and capabilities and those of others. They thus need to express the nature or intention of the motions they show as precisely as possible. Therefore, the approach has been taken of having a demonstrated sequential motion recognized as a symbol sequence of meaningful motions, i.e., a primitive sequence [1] [2].

Another point that needs to be discussed is how learners should be as they progress in learning from primitives to complicated motions, because we consider the simultaneous design of a caregiver and learner. Our key idea is a "start from immaturity" approach. Elman gradually increased memory capacity as an approach for language learning with RNN (Recurrent Neural Network), given a restricted memory capacity at the beginning [3]. Even in Nagai et al.'s study on joint attention, a robot's vision was made to develop in parallel with the gradual raising of difficulty for a task; they confirmed that this vision development improved post-training performance [**?**].

We adopt the idea of daring to impose restrictions on a learner at the beginning and have him develop gradually. On the basis of the finding in the above-mentioned related studies, we expected that an immature learner with restrictions would be better able to grasp essence of things he is learning.

Note that our goal is to investigate the influences of acquired internal representations by the changes of both a caregiver's teaching way and a learner's capability rather than to exceed the performances of existing model. Although many studies on imitation learning have succeeded using primitives and thier relationships, a few studies on imitation learning focus on changes of both a caregiver's teaching way and a learner's capability.

The rest of the paper is as follows. Section II describes related works. Section III gives an overview of our approach to developmental imitation learning. Section IV describes our implementation of imitation learning as preparation for an experiment. Section V details the experiment setup and the experiment results obtained. In Section VI we discuss

Keita Mochizuki, Harumitsu Nobuta, Shun Nishide, and Hiroshi G. Okuno are with Graduate School of Informatics, Kyoto University, Kyoto, Japan {motizuki, nobuta, nishide, okuno}@kuis.kyoto-u.ac.jp. Tetsuya Ogata is with School of Fundamental Science and Engineering, Waseda University, Tokyo, Japan ogata@waseda.jp

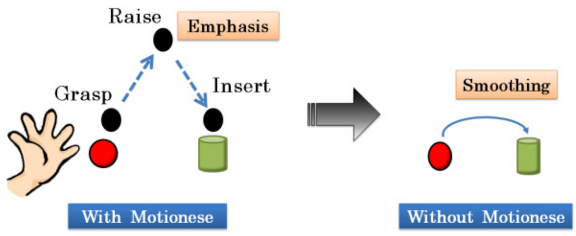


Fig. 1. Transition of motionese: an example of motionese in the task putting a ball into a cup. For an immature learner, the caregiver emphasizes pauses in sequential motion. As the learner gradually develops, the caregiver show sequential motions more smoothly.

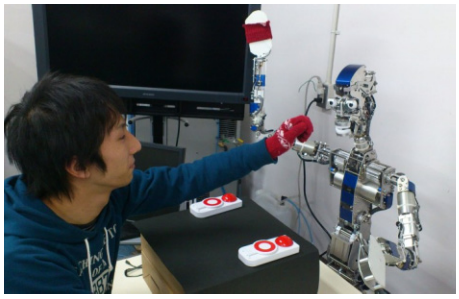


Fig. 2. Imitation experiment scene: the scene caregiver shows the motions for a task of pushing two buttons.

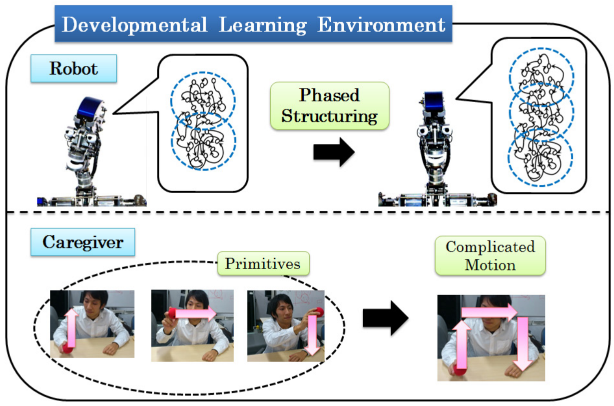


Fig. 3. Developmental learning environment: phased structuring is incorporated into a robot's self-model. Over time a caregiver changes from showing easy motions to showing complicated ones.

certain points relative to the results. Finally, we present our conclusions and mention future work to be done in Section VII.

II. RELATED WORKS

There are some existing studies on learning model introducing the concepts of "scaffolding" and "motionese".

"Scaffolding" is a way in which caregivers who are trying to help infants to learn, adjust the difficulty of a task according to an infant's perceived capabilities [5]. It is well known as an effective approach in helping infants to learn; the importance of a caregiver has been pointed out even in robotics science. Saunders et al. adopt the concept of scaffolding to robot's imitation learning [6]. In Nagai et al.'s above-mentioned study, they gradually increased the difficulty of a task by narrowing the range of a reward in reinforcement learning as an approach to acquiring joint attention [**?**]. They confirmed that this type of developmental learning accelerated learning more than usual.

The concept of "motionese", proposed by Brand et al. [7] in studies that attempted to clarify a caregiver's role in an infant's development, is defined as the modification of motions a caregiver performs when showing a task to an infant. Examples include exaggerating or simplifying a motion. Motionese also includes the effects of clarifying pauses in sequential motion by stopping or otherwise modifying motions and it is considered that motionese helps an infant recognize motions. Fig. 1 shows an example of motionese. Focusing on motionese, Nagai et al. analyzed an adult's motions while he showed a task to a robot [8]. Their results demonstrated that motionese is induced by a learner's immaturity and that effective learning is prompted by motionese. On the basis of their results, they claimed that dynamic interaction between a caregiver and learner is important.

III. OVERVIEW OF OUR MODEL

In our research, we dealt with imitation learning in the form of a human face to face with a robot teaching the latter the task of pushing two buttons on a table (Fig. 2). This section describes our approach to the imitation learning on the basis of discussion in Sec. I. The overview of our proposed model, a developmental learning environment, is described in Subsection A. Subsections B and C respectively describe our specific robot design method and the teaching method the human caregiver used in the environment.

A. Construction of Developmental Learning Environment

In using the "motionese" technique, it is important to change the difficulty for a task given by a caregiver in accordance with the learner's development. Therefore, we designed a developmental environment of robot-human imitation learning as shown in Fig. 3. In the environment, both robot and human developmentally change, each in correspondence with the other. In the first phase, the immature-state robot learns easy motions. In the second phase, where phased structuring incorporated into the robot's self-model, the developed-state robot learns more complicated motions. We believe it is important that the robot's state correspond to the difficulty of a task. In this research, we define the developmental learning environment as a union of three factors: (1) the teacher's phased teaching, (2) the learner's development (phased structuring), and (3) the relation between (1) and (2).

B. Self-Model of Robot

1) MTRNN: We incorporated the MTRNN (Multiple Timescales Recurrent Neural Network), which Tani et al.

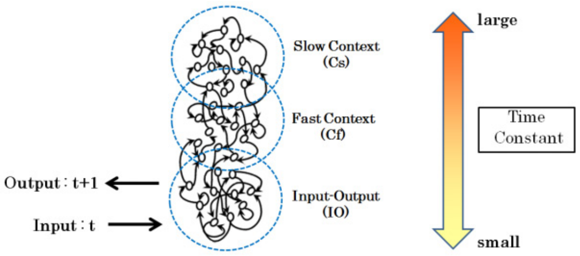


Fig. 4. Composition of MTRNN

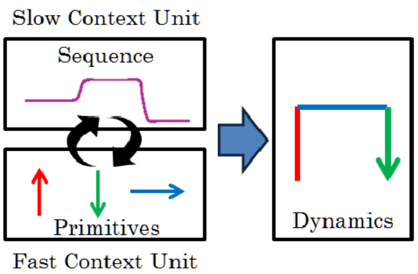


Fig. 5. Dynamics representation of MTRNN

[9] proposed as a neuro dynamical model into robot. This model, which acts as a predictor that inputs the current state and outputs the next state, can learn and generalize multiple non-linear time sequential data. It has a hierarchical structure, comprising three neuron units: an Input-Output Unit (IO) for the input-output layer, a Fast-Context Unit (Cf), and a Slow-Context Unit (Cs) for the context layer (Fig. 4). Each unit has a value called time constant; these values become large in the order of IO, Cf, and Cs. Since the internal state of a node that has a larger time constant is updated more slowly, this system provides each neuron unit with various hierarchical functions. In general, as shown in Fig.5, the Cf neuron unit represents primitives of time sequential data and the Cs neuron unit represents a sequence of the primitives. Moreover, a specific pattern can be deterministically generated by the initial Cf value (Cf_0) and the initial Cs value (Cs_0) and parametric space of the Cf_0 and Cs_0 are self-organizationally acquired by a correlation between data.

Training of the MTRNN is done using the Back Propagation Through Time (BPTT) algorithm [10]. The algorithm consists of forward calculation and weight updating.

First, the outputs of the neurons are calculated through forward calculation. The internal value of the ith neuron at step t, $u_{i,t}$ is calculated as

$$u_{i,t} = \begin{cases} (1-\frac{1}{\tau_i})u_{i,t-1} + \frac{1}{\tau_i}\left[\sum_{j\in N} w_{ij}x_{j,t-1}\right] & (t \neq 0) \\ 0 & (t = 0 \; and \; i \in IO) \\ Cf_{i,0} & (t = 0 \; and \; i \in Cf) \\ Cs_{i,0} & (t = 0 \; and \; i \in Cs) \end{cases} . \quad (1)$$

τ_i : time constant of the ith neuron
$x_{i,t}$: input value at step t
w_{ij} : weight value from the jth neuron to the ith neuron
N : set of neurons connected to the ith neuron

The output of the ith neuron is calculated by applying the sigmoid function

$$y_{i,t} \;=\; sigmoid(u_{i,t}) = \frac{1}{1+\exp(-u_{i,t})}. \quad (2)$$

Using the outputs calculated in the forward calculation, the weights are updated using the training error E defined as

$$E = \sum_t \sum_{n\in IO} (y_{i,t-1} - T_{i,t})^2 . \quad (3)$$

The weight from the jth input to the ith output is updated using the derivative of the training error $\partial E/\partial w_{ij}$ as

$$w_{ij}^{(n+1)} = w_{ij}^{(n)} - \alpha \frac{\partial E}{\partial w_{ij}}. \quad (4)$$

α : training coefficient
n : number of updates

The Cf_0 values are also updated using a back propagation algorithm along with the weight values as,

$$Cf_{i,0}^{(n+1)} = Cf_{i,0}^{(n)} - \beta \times \frac{\partial E}{\partial Cf_{i,0}}. \quad (5)$$

β : training coefficient of Cf neurons

The Cs_0 values is also calculated equally.

2) Phased Structuring of MTRNN: In constructing a developmental learning environment, it is desirable for the learner to initially be in an immature state, as described in Section I. Accordingly, we propose a phased structuring process for the MTRNN. This involves first training MTRNN with the two-layer structure of IO-Cf and then developing MTRNN into a three-layer IO-Cf-Cs structure at a certain stage, although the MTRNN is usually trained with the latter structure at the outset. The interaction between the upper layer (Cs unit) and the lower layer (Cf unit) is made to continue even after addition of the upper layer.

C. Phased teaching

The human teaches the robot in phases as described in Section I. He first teaches easy motions and then complicated ones. In this study, primitives are regarded as easy motions and motions combining multiple primitives are regarded as complicated motions.

IV. FLOW OF IMITATION LEARNING

A. Imitation Learning Algorithm

The robot's imitation learning is done by using the following algorithm in interaction with the human.

1) The human shows a motion to robot.
2) The robot recognizes the motion from the neck joint angle and hand coordinates within its view.
3) The robot actually generates the motion from the Cf_0 and Cs_0 acquired in Step 2.
4) MTRNN is trained with the time sequential data acquired in step3.
5) Go back to Step 1.

TABLE I
LEARNING CONDITION

		Phased teaching	
		w/	w/o
Phased structuring of MTRNN	w/	TD/MD	TND/MD
	w/o	TD/MND	TND/MND

Start Finish

Fig. 6. Example of primitives: Push(R)

TABLE II
DETAILES OF PRIMITIVES

Name of Motion	Initial Position	Vector Sequence
PUSH(R)	upper right	↓
RAISE(R)	lower right	↑
SLIDE	upper right	←
PUSH(L)	upper left	↓
RAISE(L)	lower left	↑

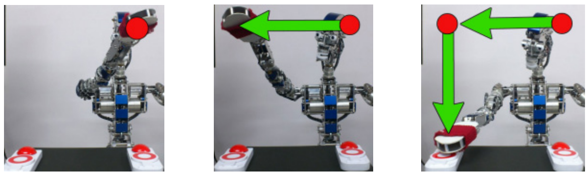
B. *Imitation Learning Process*

We prepared four teaching conditions consisting two conditions of a human caregiver (w/ phased teaching and w/o phased teaching) and two conditions of a robot learner (w/ phased structuring of the MTRNN and w/o phased structuring). Table I lists these teaching conditions. In the TD/MD condition, the MTRNN structure and the teaching style are changed at the same time. In the TD/MND condition, the MTRNN initially has three-layer structure, and only the teaching style are changed. In the TND/MD condition, the human caregiver teaches complicated motions from the beginning of learning process, and only the MTRNN structure are changed. Finally, in the TND/MND condition, the MTRNN initially has three-layer structure and the human caregiver teaches complicated motions from the beginning of learning process.

V. EXPERIMENT: IMITATION LEARNING

A. *Experiment Setup*

In this experiment of imitation learning with a human and robot, the human taught the robot the task of pushing two buttons. We used an "Actroid" humanoid robot as our test bed (Fig. 2). Actroid's joints are made flexible by controlling them with air pressure. The MTRNN input has six dimensions and the arm joint (shoulder and armpit) angle, the neck joint angle, and the hand coordinates within Actroid's view all have two dimensions. The MTRNN size is six IO nodes, 40 Cf nodes, and two Cs nodes. To enable



Start Middle Finish

Fig. 7. Example of complicated motions (Initial position: upper right. Vector sequence: ← ↓.)

Actroid to follow its hand, the robot was designed to be able to move its neck so that its hand should always be in the center of its view. Fig. 2 shows the experiment scene. As can be seen in the figure, both the human and Actroid had a red marker on their hand to facilitate detection of the hand.

The specific motions shown as primitives or complicated motions on imitation learning by the human are described below. The motions used in our experiment can be represented by an initial position and vector sequence, i.e., the queue of the move direction.

Primitives

Primitives are defined as five kinds of straight line motions needed to accomplish the task of pushing two buttons (Fig. 6). The details of each primitive are shown in Table II.

Complicated motions

Complicated motions consist of two kinds of primitives (Fig. 7). Although combining two kinds of primitives produces six different patterns, only four of these were used for imitation learning. The training data not used for imitation learning were the complicated motions whose initial positions are lower right or lower left and vector sequences are (↑ ↓).

Phased structuring and phased teaching are done when the imitation learning algorithm repeats 15 times.

B. *Learning Result*

Under each of the four learning conditions, TD/MD, TD/MND, TND/MD, and TND/MND, imitation learning was carried out. The transitions of learning errors are shown in Fig. 8. The error [cm] was evaluated by calculating the difference between a motion shown by the human and the result of Actroid's imitating the motion. The vertical axis is the error value, and the horizontal axis is the number of loops executed in the imitation learning algorithm in section IV-A.

From Fig. 8, it is confirmed that only the learning on the TND/MD condition failed while learning on other conditions converge. This result is natured considering the condition that the immature robot was taught complicated motions.

C. *Evaluation of Imitation Capability*

With the trained MTRNN, Actroid imitated seven new kinds of complicated motions combining three kinds of primitives, while also comparing the performance obtained on each condition. Hereafter, even though all seven of these motions were untrained motions, we refer to those comprising the combinations of primitives trained in Phase 2 of

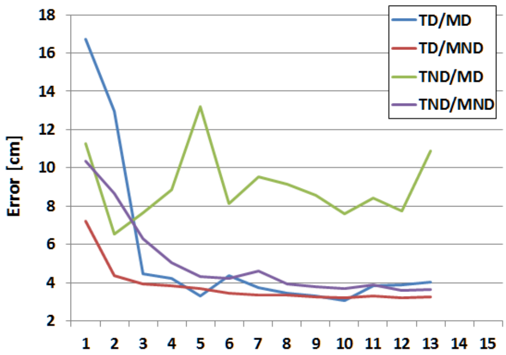


Fig. 8. Error curve

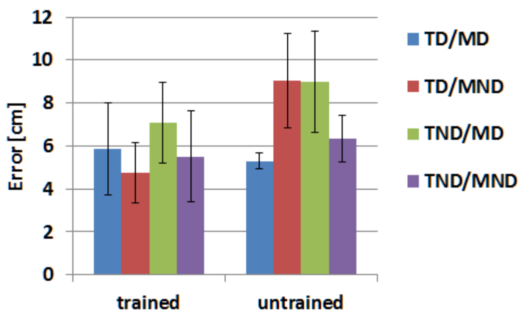


Fig. 9. Imitation performance results: for the motions composing the untrained combinations of primitives, the blue bar, the performance on the TD/MD condition, is the best, and the variance is also the smallest.

imitation learning as trained motions, while those including an untrained combination of primitives are called untrained motions. Examples of trained motions include the complicated motion whose initial position is lower right and the vector sequence is (↑ ← ↓), and ones of untrained motions include the complicated motion whose initial position is upper right and the vector sequence is (↓ ↑ ↓). Under this definition, three of the motions used for evaluation were trained ones and four were untrained ones.

The result of evaluation is shown in Fig. 9. The vertical axis is the error value, and a smaller value shows a better performance. The result confirms that the performances for untrained motions on TD/MD and TND/MND conditions are better than that on TD/MND and TND/MD.

D. Analysis of Parametric Space of MTRNN

In order to investigate how trained MTRNN represented motions, we analyzed the parametric space of the trained MTRNN for each learning condition except TND/MD because learning failed on this condition. The process of the analysis is as follows. First, we trained MTRNN to recognize the five kinds of primitives, and then Cf_0 was obtained for each primitive. Here, we define Cf_0 obtained by recognition of PUSH(R) as $Cf_0^{PUSH(R)}$. Cf_0 for other motions are defined in the same manner. Second, we divided the Cs_0 space into 225 (15×15) segments, and trained MTRNN to generate

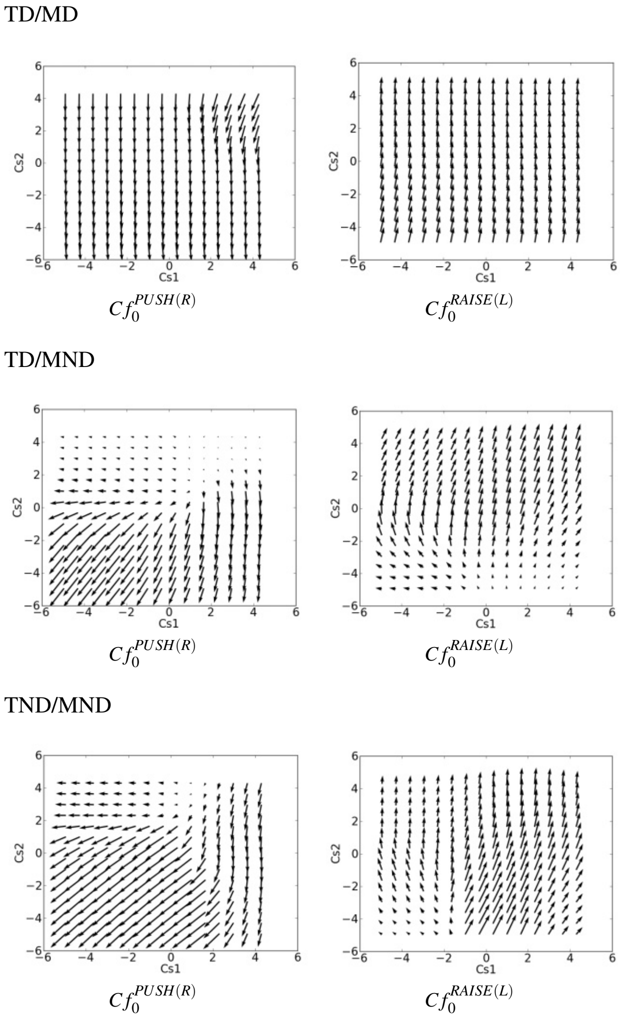


Fig. 10. Representation of Cs_0 space: each figure shows generated motions with fixed Cf_0 and altering Cs_0. A position of a vector represents the value of Cs_0 and a direction of a vector represents the direction of a motion generated with the Cs_0 and the Cf_0 under each figure. On the TD/MD condition, the directions of the vectors are almost unique.

motions from each segment. Cf_0 and the Cs_0 corresponding to the each segment were acquired.

Examples of the results are shown in Fig. 10. In this figure, the position of a vector represents Cs_0 used for generation, and the direction of a vector represents the direction of the generated motion. For example, the upper left figure shows directions of 255 motions generated from $Cf_0^{PUSH(R)}$ and Cs_0 of each segment. Note that Cf_0 is fixed and Cs_0 is altered for motion generation in each figure.

From Fig. 10, it is confirmed that the Cs_0 space of MTRNN alters on each learning condition, and vectors point the same direction on only the TD/MD condition. In other words, the generated motion changes when changing not Cs_0 but Cf_0 on the TD/MD condition. This fact shows that a direction of a motion is controlled by Cf. On the other hand, it is controlled by Cs on the other conditions.

VI. DISCUSSION

In Section V, we described Cf neuron unit controlled a direction of a motion on the TD/ND condition. Considering that Cs neuron unit controls behavior of Cf neuron unit, it can be said that Cf neuron unit acquires the function of controlling the direction and Cs neuron unit acquires the one of controlling direction order. In other words, the MTRNN trained on the condition can represent a sequential motion as a sequence of symbols, directions. The fact is same as the idea of primitives that a sequential motion should be recognized as a sequence of symbols. Note that this "primitives" mean not the term we defining but the generic term. On the other hand, a direction of a motion was controlled by Cs neuron unit on the other condition as described in Section V. From the fact, it can be said that the MTRNNs trained on other conditions regard a sequential motion as a track. In other words, they merely learn the whole track of a motion.

As described in Section I, acquiring primitives have a better influence in recognition of motions. Of course, the function controlling relationships of primitives is also necessary for recognition of sequential motions. Our experiment and analysis investigate that this two functions, primitives and the constitution, can be realized by using the hierarchical structure of MTRNN and the acquirement needs both phased teaching and structuring. These facts support the effectiveness of our phased learning.

Analysis focusing functions of a model acquired by phased learning has not been studied though performance or learning speed have been discussed. We consider that such analysis plays an important role on discussing generalization. For example, if a fact that a model regards PUSH(R) and PUSH(L) as same is shown, this represents generalization of position. It is interesting to investigate how the capability of such generalization alters by phased learning.

VII. CONCLUSIONS

This paper mainly dealt with influences of teaching style and developmental processes in learning model to the acquired representations (primitives). We investigated these influences by introducing a hierarchical recurrent network for robot model, and a form of motionese.

In this study, we constructed a developmental learning environment as an approach to imitation learning. We applied phased structuring to the MTRNN, which we incorporated into a robot, and had a caregiver show primitives as easy motions at the beginning before showing complicated motions that combined multiple primitives. Moreover, these two developments correspond to each other.

Our experiment results and analysis confirm that learning with phased teaching and structuring enables to represent a sequential motion as a sequence of symbols and to the robot to handle unknown motions.

As our next step, we plan to apply motionese to imitation learning. Since motionese emphasizes the important points of a task, we are considering an approach whereby the parts of a motion that motionese emphasizes, e.g., quiescent points, are more strongly learned. Kuniyoshi et al. also claims that there are more important parts on whole a task [11]. We are hopeful that this approach will make it possible to achieve more dynamic and effective imitation learning.

VIII. ACKNOWLEDGMENTS

This research was supported by JST PRESTO "Information environment and humans", the Ministry of Education, Science, Sports and Culture, Grants-in-Aid for Scientific Research (B) (21300076), (S) (19100003), and Creative Scientific Research (19GS0208).

REFERENCES

[1] S. Nakaoka, A. Nakazawa, K. Yokoi, H. Hirukawa, and K. Ikeuchi: "Generating Whole Body Motions for a Biped Humanoid Robot from Captured Human Dances", Proc.of 2003 IEEE International Conference On Robotics and Automation, 2003.

[2] Dana Kulic, Christian Ott, Dongheui Lee, Junichi Ishikawa, and Yoshihiko Nakamura: "Incremental learning of full body motion primitives and their sequencing through human motion observation", The International Journal of Robotics Research, Vol. 31, No. 3, pp. 330-345, 2012.

[3] J. L. Elman: "Learning and development in neural networks: the importance of starting small", Cognition,48, 71-99, 1993

[4] Y. Nagai, "The Origin and Role of Motionese: Investigating Scaffolding from Robotics", Proceedings of the 29th Annual Conference of the Robotics Society of Japan, 1A2-6, September 2011 (in Japanese).

[5] L. E. Berk and A. Winsler: "Scaffolding Children's Learning: Vygotsky and Early Childhood Education", naeyc, 1995.

[6] J. Saunders, CL. Nehaniv, K. Dautenhanh, A. Alissandrakis: "Self-imitation and environmental scaffolding for robot teaching", Int. J. Adv. Robot. Syst., vol4, 109-124, 2007.

[7] R. J. Brand, D. A. Baldwin, and L. A. Ashburn: "Evidence for motionese: modifications in mothers infant-directed action," Developmental Science, vol. 5, pp. 72 83, Mar. 2002.

[8] Y. Nagai and K. J. Rohlfing: "Computational Analysis of Motionese Toward Scaffolding Robot Action Learning", IEEE Transactions on Autonomous Mental Development, vol. 1, no. 1, pp. 44-54, 2009.

[9] Y. Yamashita and J. Tani: "Emergence of Functional Hierarchy in a Multiple Timescale Neural Network Model: a Humanoid Robot Experiment", PLoS Comput. Biol., vol.4, no.11, 2008.

[10] P. Werbos: "Backpropagation through time: What it does and how to do it", Proceedings of the IEEE - Vol.78, No.10, pp.1550-1560, 2002.

[11] Kuniyoshi Y, Ohmura Y, Terada Y, Nagakubo A, Eitoku S, Yamamoto T: "Embodied basis of invariant features in execution and perception of whole-body dynamic actions-knacks and focuses of Roll-and-Rise motion", Robotics and Autonomous Systems, vol48, no.4, pp.181 - 201, 2004.