29

Statistical Imitation Learning in Sequential Object Manipulation Tasks

Komei Sugiura, Naoto Iwahashi, Hideki Kashioka
and Satoshi Nakamura
National Institute of Information and Communications Technology
Japan

1. Introduction

Imitation learning research explores various methods for teaching robots new motions by user-friendly means of interaction (Kuniyoshi et al., 1994) (Breazeal & Scassellati, 2002). Imitation learning has garnered considerable interest from robotics and artificial intelligence communities.

For robots aimed at household environments, motions such as "to put the dishes in the cupboard" are fundamental, but difficult to program beforehand. The reason is that the desired motion depends on the size and shape of the dishes, as well as those of the cupboard, and also on whether the cupboard has a door. Furthermore, the functional capability of natural communication with users is crucial for such assistive robots. However, it is difficult to map words or symbols to motions because of the same reason mentioned above. In (Krüger et al., 2007), the difficulties involved in mapping symbols to motions are discussed in detail.

There have been studies which try to solve the problems of mapping symbols to motions in the framework of imitation learning. A motion learning/generation method based on hidden Markov models (HMMs) is proposed in (Inamura et al., 2004). (Ogawara et al., 2002a) and (Ogawara et al., 2002b) present a method in which the relative trajectories between two objects are modeled by hidden Markov models (HMMs). Furthermore, we have proposed a motion learning and generation method that is based on reference-point-dependent HMMs, which enabled the learning of motions such as rotating an object, drawing a spiral, and placing a puppet on a box (Sugiura & Iwahashi, 2007) (Haoka & Iwahashi, 2000). In (Takano et al., 2007), mocap data is learned by using HMMs, and those HMMs are converted for the retrieval of sequential motions. A method based on recurrent neural networks is proposed in (Sugita & Tani, 2005). This method is extended to deal with sequential motions in (Ogata et al., 2007).

In this chapter, we present a novel method that generates and recognizes sequential motions for object manipulation such as placing an object on another ("place-on") and moving it away ("move-away"). In this method, motions are learned using reference-point-dependent probabilistic models, which are then transformed and combined. These composite probabilistic models are used for the recognition of sequential motions performed by a user.

Moreover, motions can be generated from the composite probabilistic models in accordance with user instructions, which can then be performed by a robot arm. Fig. 1 shows the hardware platform used in this study. The system has multimodal interfaces such as a stereo vision camera and a microphone.
The main advantage of mapping symbols to motions and combing them is as follows. The machine can first decompose the given task into learned motions. It can then present a planned motion as a sequence of symbols or words. The sequence is grounded on the motions taught by the user, and so the user can understand the meaning. This is a significant safety feature because if the machine can inform the user of the planned motions before executing them, the user can determine in advance whether they are safe or not.
The rest of this chapter is organized as follows. Section 2 first states the problem we try to solve and briefly reviews related work. It then describes the proposed method in Section 3. Section 4 shows the results of simulation experiments in which the proposed method generated motions by combining learned probabilistic models. The results of physical experiments are described in Section 5 in detail. Section 6 discusses problems and possible applications with the proposed method. Finally, Section 7 concludes the chapter.

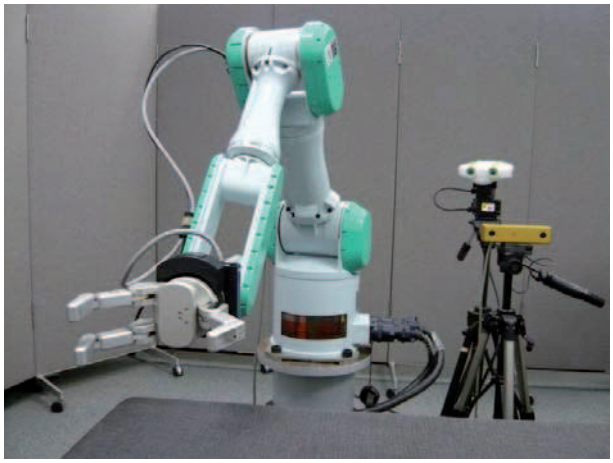


Fig. 1. Experimental platform used in this study.

2. Modeling Object-Manipulation

2.1 Reference-Point-Dependent Motions

Here, we consider the problem of learning reference-point-dependent motions in the framework of imitation learning (Kuniyoshi et al., 1994) (Breazeal & Scassellati, 2002) by a robot.
Clustering manipulation trajectories and mapping them to a verb are not sufficient for verb learning if the trajectories are considered only in the camera coordinate system. For simplicity, we assume that the mapping between the camera coordinate system and the world coordinate system is given, and that the user's utterance is accurately recognized. Let us consider the example shown in the left-hand side image of Fig. 2. The figure depicts a camera image, in which a user is placing a green puppet on the box. The trajectory itself is

meaningless since it depends on the position of the box. In the case of Fig. 2, clustering manipulation trajectories in the camera coordinate system works only if the position of the does not change.
On the other hand, if we consider a coordinate system with its center at the box, we are able to cluster the trajectories in the coordinate system and map them to a verb. We call such a coordinate system as an *intrinsic coordinate system*. The origin of an intrinsic coordinate system is called the *reference point*. It is to be noted that the origin of the intrinsic coordinate system changes with the position of the box.

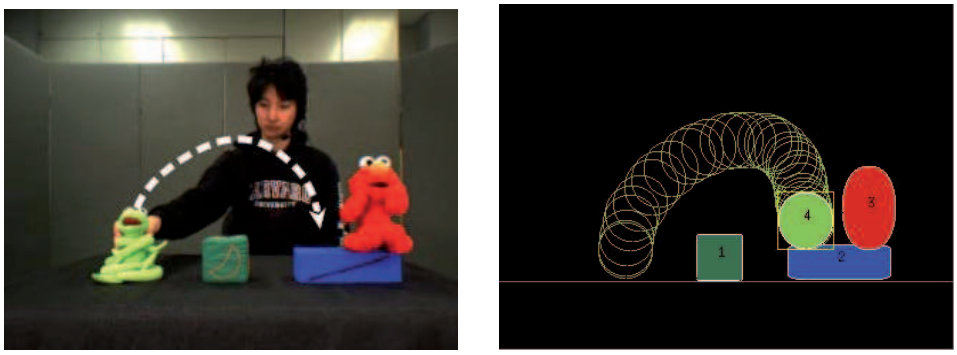


Fig. 2. Left: Example shot of camera images. The user is manipulating the green puppet. The dotted line represents the trajectory of manipulation. Right: Preprocessed visual features obtained from the image stream.

Regier investigated a model describing the spatial relationship between two objects (Regier, 1996). He proposed to model verbs as the time evolution of the spatial relationship between a trajector and a landmark. Here, a *trajector* is defined as a participant (object) that is focused on. A *landmark* has a secondary focus and a trajector is characterized with respect to a landmark. In cognitive linguistics, words representing spatial relationships such as “away” and “left of” are described as the relationship between a trajector and a landmark (Langacker, 1987). In (Ogawara et al., 2002b), the relative trajectories between two objects were modeled by using probabilistic models. The probabilistic models are used for the generation of manipulation trajectories.
In contrast, the proposed method estimates four components, which are necessary for learning object-manipulation verbs, from camera images. The components are as follows: (1) the trajector and landmark, (2) the reference point, (3) the intrinsic coordinate system, and (4) the parameters of the motion's probabilistic model. Fig. 3 shows examples, the verbs “raise” and “move-closer”. We can reasonably assume that the reference point of “raise” is the trajector's center of gravity. The intrinsic coordinate system can be a Cartesian coordinate system as shown in the left-hand figure. In the case of “move-closer”, another type of intrinsic coordinate system is necessary. In this case, the x-axis of the coordinate system passes through the centers of gravity of the trajector and the landmark. As explained in Section 4, we assume that there are several types of intrinsic coordinate systems.

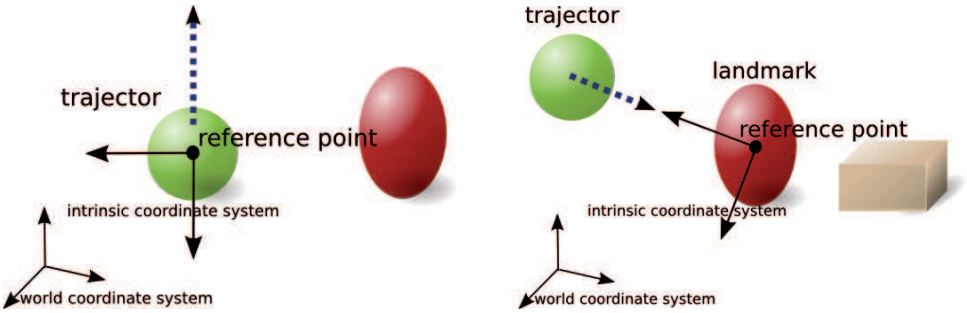


Fig. 3. Relationship between trajector/landmark, a reference point, and an intrinsic coordinate system. The spheres, ellipsoids, and box represent objects, and the arrows represent the axes of the intrinsic coordinate systems. Left: "raise". The small sphere is the trajector, and the reference point is its center. The x-axis of the intrinsic coordinate system is horizontal. Right: "move-closer". The direction of the x-axis is toward the trajector from the landmark.

2.2 Motion Learning by Reference-Point-Dependent Probabilistic Models

Consider that L training samples are given for a verb. Let V_l denote the lth training sample. V_l consists of the motion information of the trajector, ξ_l, and the candidate set of reference points, $\mathbf{R}_l$, as follows:

$$V_l = (\xi_l, \mathbf{R}_l), \tag{1}$$

$$\xi_l = \{\mathbf{y}_l(t) \mid t = 0, 1, ..., T_l\}, \tag{2}$$

$$\mathbf{y}_l(t) = \left[\mathbf{x}_l(t)^{\mathrm{T}}, \dot{\mathbf{x}}_l(t)^{\mathrm{T}}, \ddot{\mathbf{x}}_l(t)^{\mathrm{T}}\right]^{\mathrm{T}}, \tag{3}$$

$$\mathbf{R}_l = \{\mathbf{O}_l, \mathbf{x}_l(0), \mathbf{x}_{\mathrm{center}}\} \equiv \{\mathbf{x}_{r_l} \mid r_l = 1, 2, ..., |\mathbf{R}_l|\} \tag{4}$$

where $\mathbf{x}_l(t)$, $\dot{\mathbf{x}}_l(t)$, and $\ddot{\mathbf{x}}_l(t)$ denote the position, velocity, and acceleration of the trajector, respectively; T_l denotes the duration of the trajectory; and $\mathbf{O}_l$ denotes the set of the static objects' centers of gravity. The operator $|\bullet|$ represents the size of a set. The reason why $\mathbf{O}_l$ is included in $\mathbf{R}$ is that the static objects are candidate landmarks. We also include the first position of the trajector, $\mathbf{x}_l(0)$, in $\mathbf{R}$ so that we can describe a motion concept that is dependent only on the object's trajectory. Additionally, the center of the camera image, $\mathbf{x}_{\mathrm{center}}$, is added to $\mathbf{R}$ to describe motion concepts that are independent of the positions of the objects.

We assume that there are K types of intrinsic coordinate systems, and these types are provided by the designer. We denote the type of the intrinsic coordinate system by k. k corresponds to a verb, and the reference point corresponds to each V_l. We obtain the estimated intrinsic coordinate system for the lth data from the estimation of k and the reference point $\mathbf{x}_{r_l}$.

Let ${}^{C_k(\mathbf{x}_{r_l})}Y_l$ denote the trajectory in the intrinsic coordinate system $C_k(\mathbf{x}_{r_l})$. Henceforth, parameters in a particular coordinate system are written in a similar manner. Now, the index series of reference points, $\mathbf{r} = \{r_l \mid l = 1, 2, \ldots, L\}$, the type of the intrinsic coordinate

system, k, and the parameters of a probabilistic model regarding trajectories, λ, are searched for using the following maximum likelihood criterion:

$$\left(\hat{\mathbf{r}}, \hat{k}, \hat{\lambda}\right) = \underset{\mathbf{r},k,\lambda}{\operatorname{argmax}} \sum_{l=1}^{L} \log P\left(Y_l \mid r_l, k, \lambda\right) \tag{5}$$
$$= \underset{\mathbf{r},k,\lambda}{\operatorname{argmax}} \sum_{l=1}^{L} \log P\left({}^{C_k(x_{rl})}\xi_l ; \lambda\right)$$

where $\hat{\bullet}$ represents estimation. In (Sugiura & Iwahashi, 2007) and (Haoka & Iwahashi, 2000), the solution to Equation (5) is explained in detail.

3. Combination of Reference-Point-Dependent HMMs

3.1 Transformation of HMMs

Here, we consider the problem of the recognition and generation of sequential motions based on composite reference-point-dependent HMMs.

In speech recognition, HMMs are transformed for speaker adaptation by using transformation matrices. Here, the transformation matrices are independent of the order of HMMs. However, we cannot combine two reference-point-dependent HMMs in this manner. This is because the jth HMM parameters are dependent on the (j - 1)th HMM parameters (Fig. 4).

Fig. 5 illustrates an example of the process of combining two reference-point-dependent HMMs. To combine HMMs corresponding to "raise" and "move-closer," the output probability distributions of each HMM must be transformed since they represent the distributions on different coordinate systems.

An advantage of transforming intrinsic coordinate systems is the smoothness of the composite trajectories. In the proposed method, velocity and acceleration data as well as position data are used for learning. For safety reasons, changes in the velocity and acceleration data should be continuous. It is therefore important to obtain smooth trajectories of $\dot{\mathbf{x}}_l(t)$ and $\ddot{\mathbf{x}}_l(t)$ when combining two HMMs. Let us consider a case in which verbs dependent only on velocity information, e.g., "throw," are to be combined. If two HMMs were simply aligned to generate the composite trajectory, the velocity changes might be discontinuous in this case. In contrast, the proposed method, which is described in detail below, generates a smooth trajectory.

Now we consider the problem of obtaining a composite HMM from the transformation and combination of reference-point-dependent HMMs. Let $\lambda^{\langle j \rangle}$ and $C^{\langle j \rangle}$ denote the parameters and the intrinsic coordinate system, respectively, of the jth HMM. Those HMMs are modeled as left-to-right HMMs. The output probability density function of each state is modeled by a single Gaussian. The mean position vector at state s, ${}^{C^{\langle j \rangle}}\boldsymbol{\mu}_x(s)$, is transformed by the following homogeneous transformation matrix:

$$\begin{bmatrix} {}^{W}\boldsymbol{\mu}_x(s) \\ 1 \end{bmatrix} = \begin{bmatrix} {}^{W}_{C^{\langle j \rangle}}R & {}^{W}\boldsymbol{\mu}_x^{\langle j-1 \rangle}\left(S_{j-1}\right) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{C^{\langle j \rangle}}\boldsymbol{\mu}_x(s) - {}^{C^{\langle j \rangle}}\boldsymbol{\mu}_x(1) \\ 1 \end{bmatrix} \quad \left(\mathrm{j} = 1,2,\ldots,\mathrm{D},\ \mathrm{s} = 1,2,\ldots,\mathrm{S_j}\right), \tag{6}$$

where ${}^{W}_{C^{\langle j \rangle}}R$ denotes the rotation matrix from $C^{\langle j \rangle}$ to the world coordinate system W. Furthermore, $s = 0$ and $s = S_j+1$ are defined as the initial and final states of the jth HMM, respectively. The mean vector of velocity, ${}^{C^{\langle j \rangle}}\boldsymbol{\mu}_{\dot{x}}(s)$, and the mean vector of acceleration, ${}^{C^{\langle j \rangle}}\boldsymbol{\mu}_{\ddot{x}}(s)$, are rotated as follows:

$$ {}^{W}\boldsymbol{\mu}_{\dot{x}}(s) = {}^{W}_{C^{\langle j \rangle}}R \, {}^{C^{\langle j \rangle}}\boldsymbol{\mu}_{\dot{x}}(s) \tag{7} $$

$$ {}^{W}\boldsymbol{\mu}_{\ddot{x}}(s) = {}^{W}_{C^{\langle j \rangle}}R \, {}^{C^{\langle j \rangle}}\boldsymbol{\mu}_{\ddot{x}}(s) \tag{8} $$

In contrast, the diagonal items of covariance matrices for position are approximated as follows:

$$ \mathrm{diag}\ {}^{W}\Sigma_{x}(s) = \mathrm{diag}\ {}^{C^{\langle j \rangle}}\Sigma_{x}(s) \tag{9} $$

where ${}^{W}\Sigma_{x}(s)$ and ${}^{C^{\langle j \rangle}}\Sigma_{x}(s)$ denote the covariance matrices at state s in coordinate systems W and $C^{\langle j \rangle}$, respectively. The non-diagonal items of the matrices are equal to zero. The matrices for velocity and acceleration are transformed by the same simple approximation. We do not perform a rotation of the covariance matrix because the HMM-based trajectory generation method (Tokuda et al., 1995) that we use does not deal with full covariance matrices.

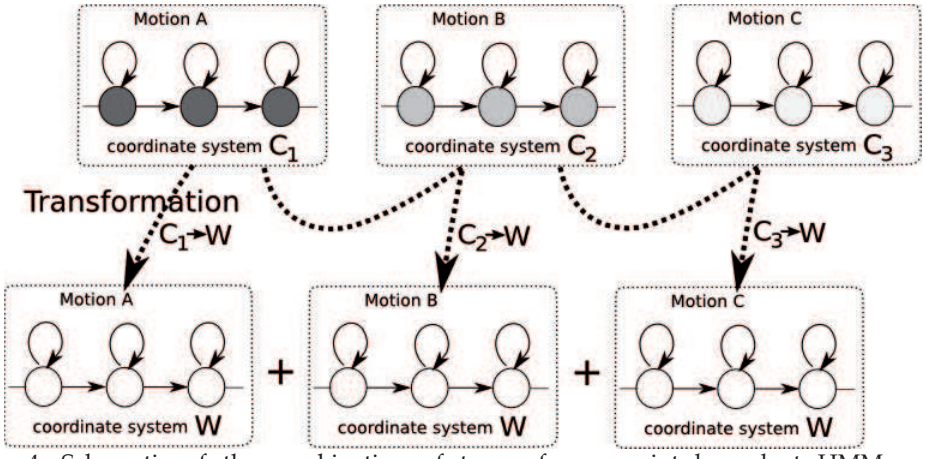


Fig. 4. Schematic of the combination of two reference-point-dependent HMMs. W represents the world coordinate system.

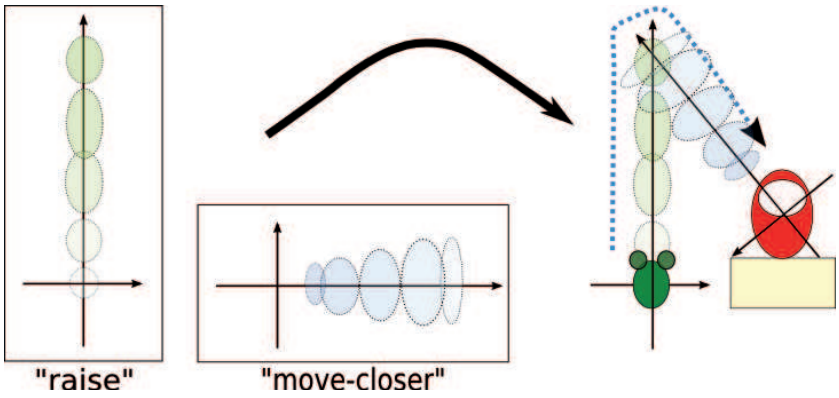


Fig. 5. Example of transformation in the combination of two HMMs, "raise" and "move-closer". Each dotted circle represents the variation of output probability distributions at each state of a left-to-right HMM. The direction of state transition is indicated by the color darkness. The intrinsic coordinate system of "move-closer" is transformed so that the its x-axis passes through both the landmark (the reference point of "move-closer") and the last position of the HMM regarding "raise". The dotted line represents the composite trajectory.

3.2 Generation of Motion Sequences by Composite HMMs

Here, we consider the problem of generating trajectories of sequential motions from composite HMMs. Suppose that a static image and the index of the trajector are given. As given in Section 2.2, we extract the candidate set of reference points, **R**. Our proposed method deals with two types of motion generation: (1) explicit instruction and (2) target instruction.

3.2.1 Explicit Instruction

The user requests the robot to move an object according to his/her instruction, which consists of a sequence of motions. Inputs from the user are the object ID and a sequence of verb-landmark pairs. The proposed method outputs the maximum likelihood trajectory that accomplishes the sequence.

Suppose a set of verbs, the intrinsic coordinate systems corresponding to verbs, and the HMM parameters corresponding to the verbs are given. Let $V = \{v_i \mid i = 1,2,...,|V|\}$ denote a set of verbs, λ_i denote HMM parameters corresponding to verb v_i, and k_i denote the index of intrinsic coordinate system corresponding to verb v_i. Let $(\mathbf{i}, \mathbf{r})$ denote a D-tuple of verb-landmark pairs as follows:

$$(\mathbf{i}, \mathbf{r}) = \left(i^{\langle 1 \rangle}, i^{\langle 2 \rangle}, ..., i^{\langle D \rangle}, r^{\langle 1 \rangle}, r^{\langle 2 \rangle}, ..., r^{\langle D \rangle}\right) \tag{10}$$

The candidate set of reference points, **R**, can be obtained from an image stream (c.f. Section 2.2). The method explained in Section 3.1 provides a composite HMM $\Lambda_D(\mathbf{i}, \mathbf{r})$.

The trajectory of Object r_{traj} corresponding to the verb-landmark pairs, (**i**, **r**), is obtained as follows:

$$\begin{aligned}\hat{\xi} &= \underset{\xi}{\text{argmax}}\, P\left(\xi \mid r_{\text{traj}}, Q_D(\mathbf{i}), \mathbf{r}, \mathbf{R}\right) \\ &= \underset{\xi}{\text{argmax}}\, P\left(\xi \mid \mathbf{x}_{\text{traj}}, Q_D(\mathbf{i}), \Lambda_D(\mathbf{i}, \mathbf{r})\right)\end{aligned} \tag{11}$$

where $Q_D(\mathbf{i})$ denotes the state sequence of the HMM corresponding to verb v_i, and $\mathbf{x}_{\text{traj}}$ denotes the initial position of the trajector. The method explained in (Tokuda et al., 1995) provides the maximum likelihood trajectory from the unknown state sequence $Q_D(\mathbf{i})$.

3.2.2 Target Instruction

Next, we consider the problem of obtaining the optimal index sequence of verb-landmark pairs, $(\hat{\mathbf{i}}, \hat{\mathbf{r}})$, which affords the maximum likelihood trajectory $\hat{\xi}$ from initial position $\mathbf{x}_{\text{traj}}$ to the goal position x_G. Most motion planning methods (see e.g., (Latombe, 1991)) do not provide linguistic expressions that explain generated trajectories. In contrast, the proposed method can generate trajectories consisting of learned motions labeled by verb indices. Therefore, the proposed method allows a robot to explain generated trajectories by natural language expressions if it has a speech interface. For example, the robot can generate confirmation utterances such as "The robot will put Object A on Object B, then raise Object A, is it OK?" Generating such confirmation utterances is desirable from the viewpoint of safety since the user can judge whether the motion is appropriate or not before the planned motion is performed.

In Target Instruction mode, the user requests that the robot move an object to a goal. Inputs from the user are the object ID and goal position x_G. The proposed method outputs the maximum likelihood sequence of verb-landmark pairs, $(\hat{\mathbf{i}}, \hat{\mathbf{r}})$, and the maximum likelihood trajectory $\hat{\xi}$. We obtain $(\hat{\xi}, \hat{\mathbf{i}}, \hat{\mathbf{r}})$ by conditioning the right side of Equation (11) with x_G and then adding $(\mathbf{i}, \mathbf{r})$ to the search arguments:

$$(\hat{\xi}, \hat{\mathbf{i}}, \hat{\mathbf{r}}) = \underset{\xi, \mathbf{i}, \mathbf{r}, D}{\text{argmax}}\, P\left(\xi \mid \mathbf{x}_{\text{traj}}, \mathbf{x}_G, Q_D(\mathbf{i}), \Lambda_D(\mathbf{i}, \mathbf{r})\right),$$

where the number of combined HMMs, D, is a search depth parameter.

3.3 Recognition of Motion Sequences by Composite HMMs

The recognition of sequential motions by reference-point-dependent HMMs can be formalized as the problem for obtaining the maximum likelihood probabilistic model for trajectory ξ under the condition where a lexicon of verbs $L_v = \{v_i, \lambda_i, k_i \mid i = 1,2,...,|V|\}$ is given. The maximum likelihood index sequence of the verb-landmark pairs, $(\hat{\mathbf{i}}, \hat{\mathbf{r}})$, is searched through the following equation:

$$\left(\hat{\mathbf{i}},\hat{\mathbf{r}}\right) = \underset{\mathbf{i},\mathbf{r},D}{\operatorname{argmax}} P(\xi \mid \mathbf{i},\mathbf{r},D,\mathbf{R}) = \underset{\mathbf{i},\mathbf{r},D}{\operatorname{argmax}} P(\xi \mid \Lambda_D(\mathbf{i},\mathbf{r}))$$

4. Simulation Experiments

4.1 Experimental Setup

We first conducted simulation experiments for evaluating the proposed method. The simulator consists of a graphical interface and a mouse. Virtual objects shown on the screen can be manipulated by dragging them with a mouse.

In the learning phase, a user was asked to teach motions. The trajectories were recorded and used for training probabilistic models. The trajectories for the following seven verbs were collected.

raise, move-closer, move-away, rotate, place-on, put-down, jump-over

For each verb, the number of training samples was 15. Those motions were taught by the user in the learning phase beforehand, and they were constant throughout the motion generation experiments.

The verbs were successfully learned in the experiment. Fig. 6 shows the examples of the training samples. In this figure, the thick arrows represent the trajectories of an object manipulated by the user. The thin arrows represent the x- and y-axes of the estimated type of the intrinsic coordinate system. The type name is shown at the lower right of each illustration. The types of the intrinsic coordinate system are defined as follows:

- C_1: A coordinate system with its origin at the landmark position. C_1 is a translated camera coordinate system. The x-axis is inverted in case the x-coordinate of the original position of the trajector is negative after translation.
- C_2: An orthogonal coordinate system with its origin at the landmark position. The direction of the x-axis is from the landmark towards the trajector.
- C_3: A translated camera coordinate system with its origin at the original position of the trajector.
- C_4: A translated camera coordinate system with its origin at the center of the image.

After probabilistic models were trained, we carried out motion generation simulation experiments. Two types of motion experiments were performed: for target instruction and for explicit instruction.

In the target instruction experiment, the user requested the robot to move an object in a camera image. The inputs were object ID and a goal point, and the outputs from the robot were both a sequence of verb-landmark pairs and a trajectory to the goal. We used 64 grid points as goal points. The search depth parameter D was set as $D = 3$. Therefore, the estimated motion sequence consisted of up to three HMMs.

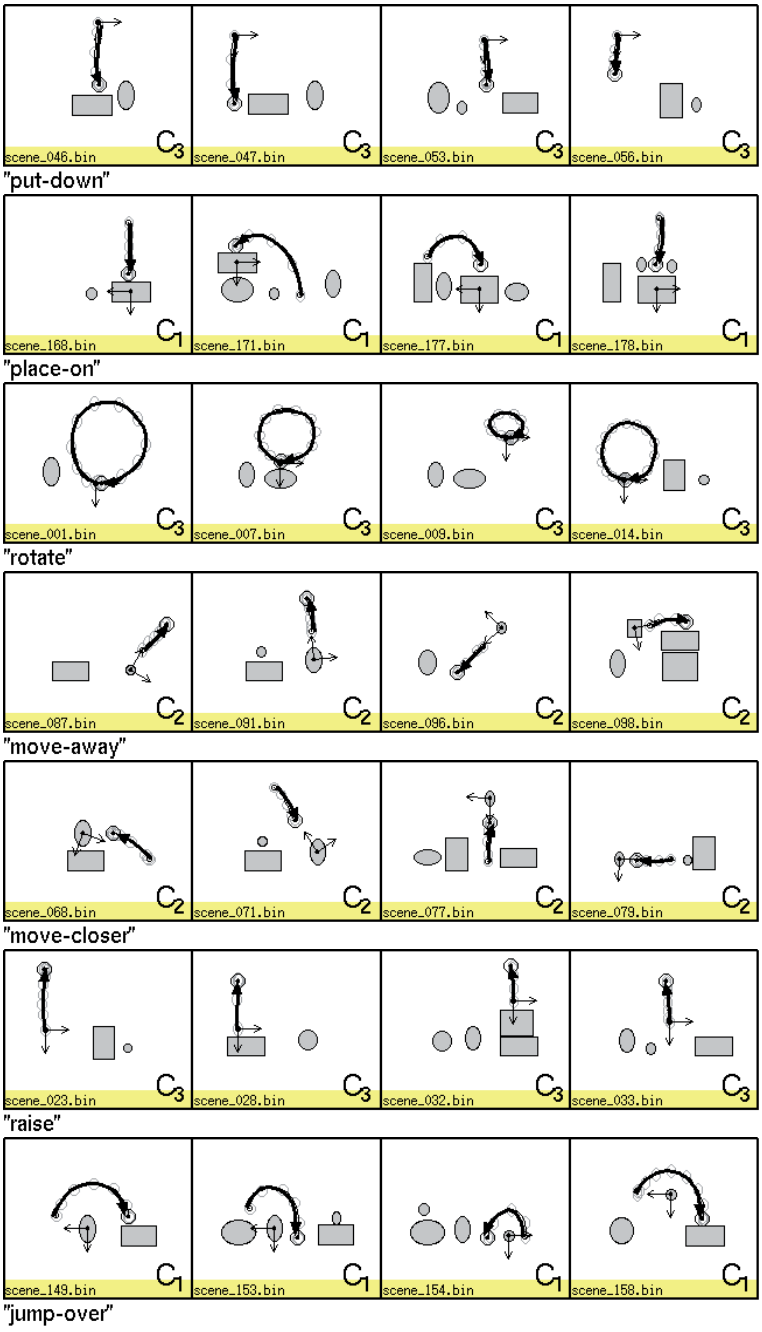


Fig. 6 Examples of training samples.

4.2 Result (1): Motion Generation

4.2.1 Explicit Instruction

The explicit instruction experiment was carried out in the same simulation environment. Fig. 7 shows two examples of motion generation: "jump object 1 over object 2, then put object 1 down, and move object 1 closer to object 4" and "jump object 2 over object 1, jump object 2 over object 1 again, and then place object 2 on object 5". The inputs for the two cases were as follows:

	Trajector	Motion sequence
(a)	Object 1	<jump-over, 2><put-down, no landmark><move-closer, 4>
(b)	Object 2	<jump-over, 1><jump-over, 1> <place-on, 4>

Fig. 7 shows that the three motions were combined smoothly. To examine this result quantitatively, we plotted the evolution of position, velocity, and acceleration in case (a) in Fig. 8. Fig. 8 verifies that the composite trajectory of velocity and that of acceleration were continuous.

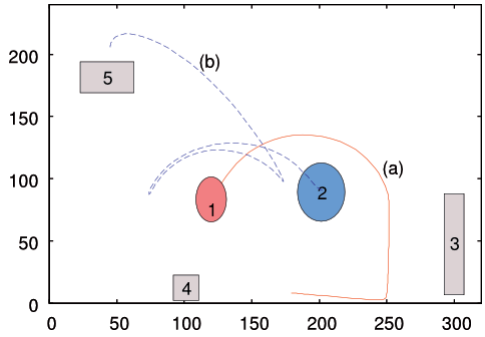


Fig. 7. Explicit instruction

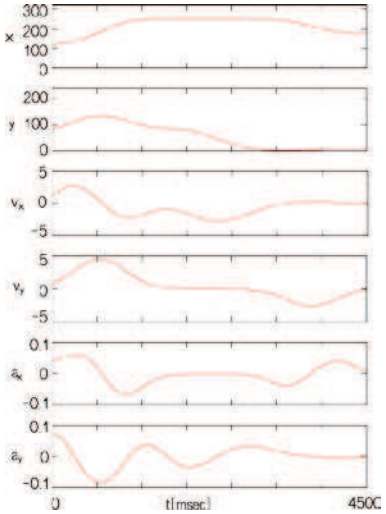


Fig. 8. Evolution of position, velocity, acceleration under condition (a) in Fig. 7.

4.2.2 Target Instruction

The simulation environment is shown in Fig. 9. There were five objects in the environment (depicted by numbered boxes and circles). Object 1 was used as the trajector. Fig. 9 shows the examples of maximum likelihood trajectories output by the proposed method. In the figure, bracketed pairs represent the estimated sequences of verb-landmark pairs. For example, <place-on, 2><move-closer, 5> signifies "place object 1 on object 2, and move object 1 closer to object 5."

From Fig. 9, we can see that the proposed method combined two motions smoothly rather than independently. In particular, although <move-away, 2> and <move away, 2><jump-over, 4> share <move-away, 2>, the trajectories do not overlap with each other. This is probably due to the large variation for the last position in the learned probabilistic model of "move-away." In other words, a part of the trajectory of <move-away, 2> was curved to smoothly combine <move away, 2> and <jump-over, 4>, but the likelihood of the resultant trajectory was still high because of its large variance.

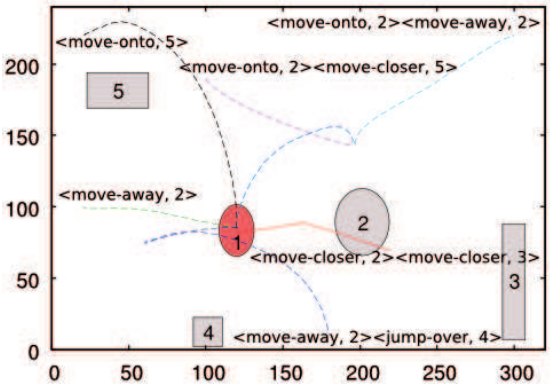


Fig. 9. Generated motions in Target Instruction mode for manipulating object 1. Numbered boxes and circles represent objects. Bracketed pairs represent the estimated sequence of verb-landmark pairs. Specifically, <place-on, 2><move-closer, 5> means "place object 1 on object 2, and move object 1 closer to object 5".

5. Physical Experiments

5.1 Experimental Setup

The experiments were conducted by using a PA-10 manipulator manufactured by Mitsubishi Heavy Industries with seven degrees of freedom (DOFs). The manipulator was equipped with a BarrettHand, a four-DOF multifingered grasper. The user's movements were recorded by a Bumblebee 2 stereo vision camera at a rate of 30 [frame/s]. The size of each camera image was 320 × 240 pixels. The left-hand side image of Fig. 2 shows an example shot of an image stream, and the right-hand side image of the figure shows the internal representation of the image stream. All the motion data used for learning and recognition were obtained from physical devices. In addition, motion generation results were examined in an environment using the manipulator and physical objects such as puppets and toys.

The difference of parameter setup between the simulation and physical experiments was the number of training samples, L. In physical experiments, L was set to 9 for each motion.

5.2 Motion Generation

5.2.1 Explicit Instruction

Fig. 10 shows an example trajectory generated by the proposed method. The solid line represents the trajectory generated in the explicit instruction mode. The input for the explicit instruction mode was as follows:

Trajector	Motion sequence
Object 2	<move-away, 1><jump-over, 4><move-closer, 4>

From **Fig. 10**, we can see that the proposed method generated an appropriate trajectory. To support this, the manipulator is shown performing the generated trajectory, as shown in **Fig. 11**.

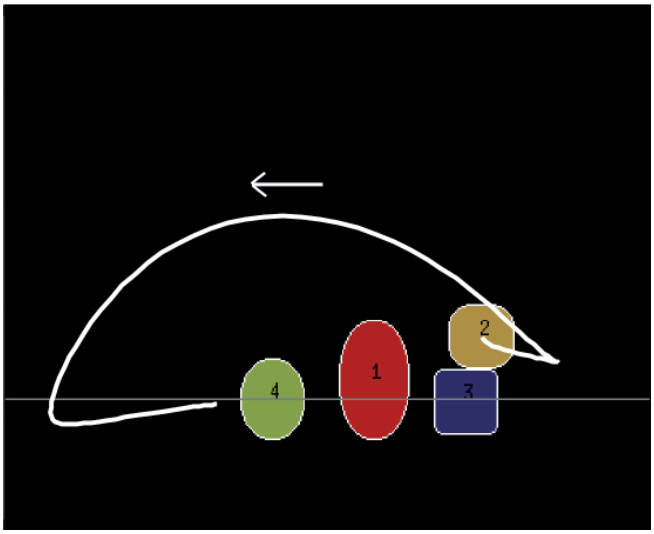
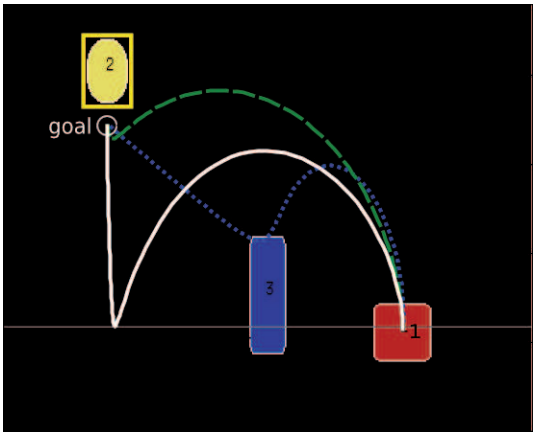


Fig. 10. Generated trajectory in the explicit instruction mode.

Fig. 11. Sequential photographs of the manipulator executing the trajectory shown in Fig. 10.

5.2.2 Target Instruction

For the target instruction mode, the top three trajectories are shown in **Fig. 12**. The trajector ID was set to 1, and the goal position used is indicated in the figure. In the figure, the solid, broken, and dotted lines represent the best, second-best, and third-best trajectories, respectively. In addition, the top three verb-landmark pairs and the log likelihood are shown in the figure.



1.	(solid line)	<jump-over, 3> <move-closer, 2>	-16.45
2.	(broken line)	<jump-over, 3>	-18.66
3.	(dotted line)	<place-on, 3> <move-closer, 2>	-25.06

Fig. 12. Generated trajectories in the target instruction mode.

5.3 Motion Recognition

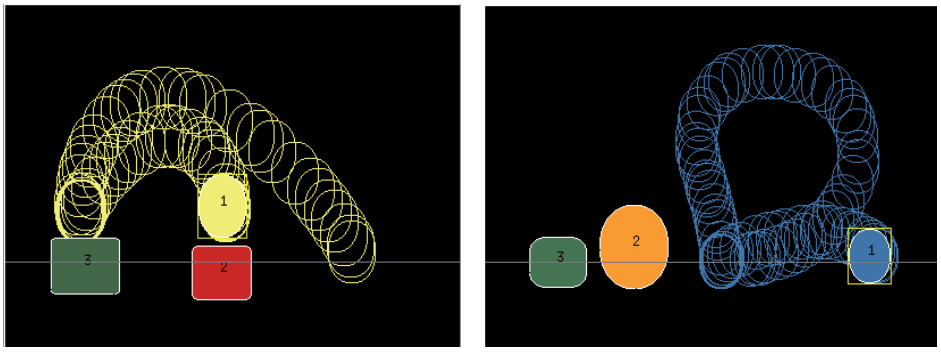
The user was presented with six pairs of randomly chosen verbs, and performed the motions sequentially. The manipulation trajectories and the positions of the static objects were recorded to obtain a test set. For each pair, five different object settings were given. Therefore, the size of the test set was 30.

Fig. 13 illustrates example trajectories in the test set. In the figures, the top three recognition results for each scene are shown. The bracketed pairs and numbers represent the estimated sequences of verb-landmark pairs and the log likelihood, respectively.

We can see that a correct recognition result was obtained for the left-hand figure of Fig. 13. On the other hand, the correct recognition result for the right-hand figure does not have the maximum likelihood. This is considered to be due to the fact that the trajectory in the training set always starts from pause ($\dot{\mathbf{x}}_I(0) = \mathbf{0}$), and therefore, the composite HMMs contain states representing pauses between motions. However, the two motions are consecutively performed. Therefore, the likelihood of such trajectory given the combined HMMs was smaller than the likelihood of the trajectory given an HMM.

Table 1 shows the number of correctly recognized samples. The column labeled "*n*-best" stands for the number of correct answers contained in the top *n* recognition results. The accuracy of 1-best, 2-best, and 3-best recognition results are 63%, 83%, and 87%, respectively.

In the table, we obtain an accuracy of 80% (12/15) for sequences (1), (3), and (4). This is reasonable since we have obtained an accuracy of 90% for the recognition of single motions in preliminary experiments. However, we obtain an accuracy of 47% (7/15) for sequences (2), (5), and (6), which contains at least one C_2 verb. This result also supports the fact that the approximation (Equation (9)) deteriorated the recognition accuracy.



1. <place-on, 3> <place-on, 2> :	-22.04	1. <move-away, 2> :	-22.18
2. <jump-over, 2> <place-on, 2> :	-23.79	2. <rotate> <move-away, 2> :	-22.65
3. <place-on, 3> <move-away, 3> :	-28.79	3. <rotate> :	-25.06

Fig. 13. Examples of test set. The recognition results for each scene are shown below the corresponding figure. Left: "place Object 1 on Object 3, then place Object 1 on Object 2." Right: "rotate Object 1, and then move Object 1 away from Object 2."

Test set	1-best	2-best	3-best
(1) rotate + rotate	5	5	5
(2) move-away + move-closer	3	3	3
(3) place-on + place-on	3	5	5
(4) rotate + jump-over	4	4	4
(5) rotate + move-away	2	4	4
(6) move-closer + place-on	2	4	5
Total	19/30 (63%)	25/30 (83%)	26/30 (87%)

Table 1. Number of correctly recognized samples.

5. Discussion

5.1 Recognition Accuracy

Here, we discuss two causes for the deterioration in the recognition accuracy.

The first one is that sequential motions performed by users tend to be smoothly combined. However, the likelihood for such motions is not always high. This is because the trajectory

in the training set always starts from pause ($\dot{\mathbf{x}}_1(0) = \mathbf{0}$), and therefore, the composite HMMs contain states representing pauses between motions.
We assume that this problem can be solved by using pause HMMs. In this case, a sequence of HMMs comprising a motion HMM sandwiched between pause HMMs are trained. Furthermore, we can obtain a composite HMM by aligning the HMMs of pause, motion A, motion B, and pause, and thereby combining two motions.
Another problem is that Equation (9) does not consider the full covariance matrices, and so the rotation of coordinate systems is ignored. As stated above, this approximation deteriorated the recognition accuracy for the C_2 verbs. In the future work, we will perform the rotation of covariance matrices.

5.2 Collision Avoidance

The proposed method has a possibility of generating inappropriate trajectories leading to object collision. For instance, Fig. 9 shows that the trajectory of <move-closer, 2><move-closer, 3> runs through Object 2.
There are at least three solutions to this problem. The first is to select the maximum likelihood sequence of verb-landmark pairs among which no collision occurs. This is the simplest solution since the positions of all static objects are evident in camera images. Another solution is to slightly change the maximum trajectory so as to avoid collisions. The third one is to modify the output probability density functions of HMMs by setting them to 0 near the position of obstacles. The third method, however, will not work if there are many obstacles in the camera image.

5.3 Future Work

The proposed method can be applied for motion planning rather than manipulating objects. One example is a mobile robot that generates a path to a goal set by the user by combining learned motions and informs the user about it. Suppose a camera is placed on the ceiling of an office and a camera image, as shown in Fig. 9, is obtained. In this case, the robot can decompose an instruction such as "go to the president's room" into learned motions such as "move-forward" and "move-along."
Another application would be the prediction of motions. In this chapter, partial trajectories were not considered. However, if the motion recognition is performed with a part of the trajectory, the system can predict the landmark of the motion and help the user. For example, this could be used in a technique to unlock a door before the user grasps the door knob.

6. Conclusion

It is important for robots to be able to report their internal states to humans in a comprehensive manner in environments shared by both. For example, it is critical for the safety of people that the robots are able to communicate their next move.
In this chapter, we have presented a method to combine reference-point-dependent probabilistic models. The experimental results of the Target Instruction mode revealed that the proposed method successfully decomposed goal-oriented motions into learned motions. This indicates that the robot could decompose the given task into learned motions and then present the planned motions, in a manner easy for the user to understand. This is a

significant safety feature because if the machine can inform the user of the planned motions before executing them, the user can decide in advance whether they are safe or not.
Furthermore, the proposed method enables the recognition of sequential motions. One of the contributions of this work is the recognition of sequential object manipulation. In the future work, the proposed method would be applied to problems for the retrieval of motions from video data and action mining in ubiquitous computing.

7. References

Breazeal, C. & Scassellati, B. (2002). Robots that imitate humans. *Trends in Cognitive Science*, Vol. 6, No. 11, pp. 481–487.

Haoka, T. & Iwahashi, N. (2000). Learning of the reference-point-dependent concepts on movement for language acquisition, *Proceedings of Pattern Recognition and Media Understanding*, Vol. 2000-105, pp. 39–46.

Inamura, T.; Toshima, I.; Tanie, H. & Nakamura, Y. (2004). Embodied symbol emergence based on mimesis theory, *International Journal of Robotics Research*, Vol. 23, No. 4, pp. 363–377.

Krüger V.; Kragic, D.; Ude, A. & Geib, C. (2007). The meaning of action: a review on action recognition and mapping, *Advanced Robotics*, Vol. 21, No. 13, pp. 1473–1501.

Kuniyoshi, Y.; Inaba, M. & Inoue, H. (1994). Learning by watching: extracting reusable task knowledge from visual observation of human performance. *IEEE Transactions on Robotics and Automation, Vol.* 10, No. 6, pp. 799–822.

Langacker, R.W. (1987). *Foundations of Cognitive Grammar: Theoretical Prerequisites*, Stanford University Press.

Latombe, J.C. (1991). *Robot motion planning (Kluwer International Series in Engineering and Computer Science, 124), S*pringer.

Ogata, T.; Murase, M.; Tani, J.; Komatani, K. & Okuno, H.G. (2007). Two-way translation of compound sentences and arm motions by recurrent neural networks. *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1858–1863.

Ogawara, K.; Takamatsu, J.; Kimura, H. & Ikeuchi, K. (2002a). Generation of a task model by integrating multiple observations of human demonstrations. *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*,

Ogawara, K.; Takamatsu, J.; Kimura, H. & Ikeuchi, K. (2002b). Modeling manipulation interactions by hidden Markov models, *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robos and Systems*, pp. 1096-1101.

Regier, T. (1996). *The Human Semantic Potential: Spatial Language and Constrained* Connectionism (Bradford Books), The MIT Press.

Sugita, Y. & Tani, J. (2005). Learning combinatoriality from the interaction between linguistic and behavioral processes, *Adaptive Behavior*, Vol. 13, No. 1, pp. 33-52.

Sugiura, K. & Iwahashi, N. (2007). Learning object-manipulation verbs for human-robot interaction, *Proceedings of the 2007 International Workshop on Multimodal Interfaces in Semantic Interaction*, pp. 32-38.

Takano, W.; Yamane, K. & Nakamura, Y. (2007). Capture database through symbolization, recognition, and generation of motion patterns, *Proceedings of the 2007 IEEE International Conference on Robotics and Automation*, pp. 3092-3097.

Tokuda, K.; Kobayashi, T. & Imai, S. (1995). Speech parameter generation from HMM using dynamic features. *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, pp. 660–663.

Advances in Robot Manipulators

Edited by Ernest Hall

ISBN 978-953-307-070-4
Hard cover, 678 pages
Publisher InTech
Published online 01, April, 2010
Published in print edition April, 2010

The purpose of this volume is to encourage and inspire the continual invention of robot manipulators for science and the good of humanity. The concepts of artificial intelligence combined with the engineering and technology of feedback control, have great potential for new, useful and exciting machines. The concept of eclecticism for the design, development, simulation and implementation of a real time controller for an intelligent, vision guided robots is now being explored. The dream of an eclectic perceptual, creative controller that can select its own tasks and perform autonomous operations with reliability and dependability is starting to evolve. We have not yet reached this stage but a careful study of the contents will start one on the exciting journey that could lead to many inventions and successful solutions.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821