Embodiment-Specific Representation of Robot Grasping using Graphical Models and Latent-Space Discretization

Dan Song, Carl Henrik Ek, Kai Huebner and Danica Kragic

Abstract— **We study embodiment-specific robot grasping tasks, represented in a probabilistic framework. The framework consists of a Bayesian network (BN) integrated with a novel multi-variate discretization model. The BN models the probabilistic relationships among tasks, objects, grasping actions and constraints. The discretization model provides compact data representation that allows efficient learning of the conditional structures in the BN. To evaluate the framework, we use a database generated in a simulated environment including examples of a human and a robot hand interacting with objects. The results show that the different kinematic structures of the hands affect both the BN structure and the conditional distributions over the modeled variables. Both models achieve accurate task classification, and successfully encode the semantic task requirements in the continuous observation spaces. In an imitation experiment, we demonstrate that the representation framework can transfer task knowledge between different embodiments, therefore is a suitable model for grasp planning and imitation in a goal-directed manner.**

I. INTRODUCTION AND CONTRIBUTIONS

An important challenge in imitation learning [1] is the *correspondence problem* [2] due to the differences in embodiments between the teacher and the learner. Namely, direct copy of the demonstrated action may fail to achieve the goal of the demonstrated task, or even may not be feasible because the robot has different mechanical constraints. Several works have addressed the correspondence problem by constraining the imitation at a task space that is shared by the teacher and the learner. This common space can be either pre-specified by the user [3], or automatically identified using machine learning techniques [4]. In relation to robot arm movements, such a common space is usually the trajectory of the hand position and orientation in the Cartesian space, which is then reproduced by the robot solving the inverse kinematics [3].

However, identifying a common task space is difficult in the domain where the robot has to interact with the world: to grasp and manipulate objects. We may ask: *What is the common task space for* ***pouring*** *water into a cup?* Here, the robot has to consider not only the hand pose, finger configuration, but also the pose of the object, and its physical properties that determine if the object *affords* this task. Also, to firmly grasp an object for further manipulation, important control parameters such as the grasping force have to be considered. For the specific example of pouring, a good grasp would be the one that results in a stable manipulation of the objects during pouring, taking into account that the grasp should not be at a position so that the opening part is blocked.

To parameterize such semantic task constraints in a deterministic manner is hard. First, the task requirements can vary a lot with the task itself. For example, the constraint of a *hand-over* task is to leave enough free-space on the object so that it allows re-grasp. It is clearly described by a set of object and action variables that are different from those that define the pouring task. In addition, this task description may also be hand-specific. For example, human can apply power grasps to hand-over an apple, but a robot may fail with the same grasp type simply because it has a larger hand.

Our previous work [5] addressed already some of these challenges. We used a probabilistic graphical model – Bayesian Network (BN) – to encode such semantic task requirements for robot grasping. The network modeled the conditional distributions among a set of object and grasp related features that are hand-specific, together with the task requirements that have been introduced by human labeling. The initial results were very promising: the model allowed the robot not only to reason about high-level task representations, but also to make detailed decisions about which object to use and which grasp to apply in order to fulfill the requirements of the assigned task. However, the BN used in [5] models both discrete and continuous variables, which presents some limitations particularly in structure learning of the network. We therefore developed a novel multivariate discretization model presented in [6]. The model uses a non-linear dimensionality reduction technique to learn a low-dimensional latent representation of the observations. A mixture model is then learned to discretize the data allowing for a compact, generative representation of the data. The model is fully probabilistic and capable to facilitate structure learning from discretized data, while retaining the continuous representation.

The contribution of this paper is to create a fully probabilistic framework for embodiment-specific representation of robot grasping tasks. We do this by integrating the BN approach from [5] with the multi-variate discreitization model from [6]. The proposed approach is evaluated using human and robot object grasping examples in a simulated framework. We show that the two hands result in rather different network structures indicating potentially different conditional dependencies among the same set of task variables. Also, the conditional distributions in the individual variables turn to be hand specific. However, both models achieve good task classification, and represent the semantic task requirements in the continuous observation spaces.

D. Song, C.H Ek, K. Huebner and D. Kragic are with KTH – Royal Institute of Technology, Stockholm, Sweden, as members of the Computer Vision & Active Perception Lab., Centre for Autonomous Systems, www: `http://www.csc.kth.se/cvap`, e-mail addresses: {`dsong,chek,khubner,danik`}`@csc.kth.se`.

In an imitation experiment, we demonstrate that the proposed framework successfully transfers task knowledge between different hands and provides the means for grasp planning and imitation in a goal-directed manner. Compared with [5], [6], the current work extends the learning domain to a slightly more challenging dataset with more tasks, objects and embodiments.

II. Models

A Bayesian Network is a directed graphical model which exploits conditional dependencies in the data in order to learn an efficient factorization of the joint distribution in the data,

$$p(X_1, \ldots, X_N) = \prod_{i=1}^{N} p(X_i|\pi_i), \tag{1}$$

where X_i represents variables and π_i its parents in the network. The model is defined by a set of *parameters* defining each conditional model and by the *structure* of the vertices representing the conditional dependencies. Learning both structure and parameters from both continuous and discrete data poses a significant challenge. Most algorithms for structure learning only work with discrete variables therefore a pre-discretization step is necessary [7].

In [6] we developed a method capable of learning an intermediate discrete representation of a high-dimensional, continous observation space. In specific we apply techniques from generative dimensionality reduction – the Gaussian Process Latent Variable Model (GP-LVM) [8]. Its sparse variational formulation [9] provides both efficient learning of the latent space and the initial clusters for the subsequent discretization. Due to space limit, we refer the readers to [8], [9], [6] for detailed formulations of sparse GP-LVMs.

In this paper we improve the discretization model by incorporating an additional prior that encourages the location of the states to be sparse. In other words, we want a representation where each of the cluster centers are well separated in the latent space. To do so, we propose a prior over the discretization centers $\mathbf{U} = \{u_1, u_2, \ldots, u_M\}$, which are the inducing points of the sparse GP-LVM. This prior penalizes the L_1 norm of the off-diagonal elements in the inner-product matrix computed between the inducing points,

$$p(\mathbf{U}|\theta_U, \beta_U) = \mathcal{N}(\sqrt{D(\mathbf{U}, \theta_U)}|0, \beta_U^{-1}), \tag{2}$$

$$D(\mathbf{U}, \theta_U) = \sum_{ij}^{M} \lambda_{ij} k_u(u_i, u_j, \theta_U), \lambda_{ij} = \begin{cases} 0 & i = j \\ 1 & i \neq j \end{cases} .$$

If $k_u(u_i, u_j)$ is a smooth monotonically decreasing function with respect to $||u_i - u_j||$ the distribution will encourage a representation with well separated clusters. The parameters β_U and θ_U control the strength of the prior and the smoothness of k_u respectively. Here we use a radial basis function where θ_U controls the width of the function that also relates to the strength of the prior. Including the above prior into the method presented in [6] we are able to further improve previous results.

Once we have acquired a discrete version of the observations, we use a greedy search algorithm to find the structure, or the directed acyclic graph (DAG), in a neigborhood of graphs that maximizes the network score (Bayesian information criterion [10]). The search is local and in the space of DAGs, so the effectiveness of the algorithm relies on the initial DAG. As suggested by [11], we use another simpler algorithm, the maximum weight spanning tree [12], to find an oriented tree structure as the initial DAG. We assume the task class variable is the 'cause' of the systems thus the root node of the network.

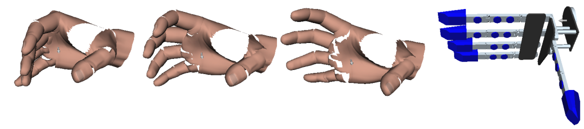


Fig. 1. Randomly sampled Eigengrasp preshapes of the human hand, and the preshape of Armar hand.

Inference

A trained BN defines an efficient factorization of the joint distribution of the observations. By converting the acyclic graph into a tree, the junction tree algorithm [13] allows efficient inference on the marginal distribution of any variable(s) conditioned on observations of others. The output of the inference is a multinomial distribution for variable X_i over each of its discrete states u_{ik} while the observation of the rest of the network $\mathbf{V}_i$ is at the state $\mathbf{v}_j$,

$$\mu_{ijk} = p(X_i = u_{ik}|\mathbf{V}_i = \mathbf{v}_j). \tag{3}$$

We will now describe how we can recover a continuous estimate of variable X_i in its original observation space $\mathbf{Y}$ from this distribution.

Each point of $\mathbf{x}_i$ on the latent space $\mathbf{X}$ defines a distribution over the observed data space $\mathbf{Y}$ through the GP that models the generative mapping. Therefore in order to acquire a continuous estimate in $\mathbf{Y}$ we need to determine a distribution over the latent space $\mathbf{X}$ associated with the multi-nominal distribution μ_{ijk}. In order to achieve this we first learn a parametric mixture model with the location of the inducing points as the mixture centers. In specific we use full-covariance Gaussian basis functions to define a mixture model with M components (discrete states),

$$p(\mathbf{x}_i) \propto \prod_{k=1}^{M} \lambda_k \mathcal{N}(\mathbf{x}_i|u_{ik}, \Sigma_{ik}), \tag{4}$$

and learn its parameters of the mixture model using the standard EM approach. The multinomial distribution output μ_{ijk} from the network defines a distribution over the inducing points u_{ik}. We use this distribution to specify the coefficient for the learned mixture components to create the following *conditional* mixture model,

$$p(\mathbf{x}_i|\mathbf{v}_j) \propto \prod_{k=1}^{M} \mu_{ijk} \mathcal{N}(\mathbf{x}_i|u_{ik}, \Sigma_{ik}). \tag{5}$$

We can then sample from the above distribution in order to find locations over the latent space which corresponds to our continuous estimate.

III. Data Generation

Tab. I shows the features used in this work. The features describing each grasp are divided into three sub-sets: *object features* (O) from the object representation, *action features* (A) from the planned grasps, and *constraint features* (C) resulting from the complementation of both, i.e. the hand-object configuration. Each grasp was visualized in GraspIt! to a human tutor who associated it with a task label (T).

Two hand models are used in the experiments: the human 20 degrees-of-freedom (DoF) hand, and the Armar 11 DoF hand [14]. The database includes in total 48 objects covering 6 object classes (8 models per class). Each object class includes 4 different object shapes each of which is scaled to 2 sizes – small and average. Four tasks are labeled: *hand-over, pouring, tool-use* and *dish-washing*. Compared to previous work we include the new task, *dish-washing*. In summary, the current approach extends [5], [6] with a more challenging dataset and a new robot hand showing the scalability of the framework.

Note that the human hand has an Eigengrasp pre-configuration *egpc* as one of the action variables, whereas the Armar hand does not. Human hand is high-dimensional, but not all of the DoFs are indepently controlled. Therefore we use the idea of [15] to define random preshape configurations of the hand in the 2D *eigen grasp* space (i.e. *egpc*). The two dimensions of *egpc* roughly represent the levels of finger spreading and finger flexion respectively. A detailed implementation on *egpc* can be found in [5]. For the Armar hand the spreading component is missing, and the four fingers opposing the thumb can only flex and extend. We therefore do not implement random samples in preshape configuration for Armar hand, and the hand always starts at a preshape while all the DoFs are at zero, i.e. the fingers are fully extended (see Fig. 1).

Fig. 2 shows the schematic of the data generation process. To extract those features for each hand, we first generate grasp hypotheses using the grasp-planner BADGr [16], and evaluate them as scenes of object-grasp configurations in a grasp simulator, GraspIt! [17]. BADGr includes extraction and labeling modules to provide the set of variables presented in Tab. I. The interested reader is referred to [5], [16] for more details on the feature extraction. We emphasize that the grasp representation does not have to be non-redundant, e.g. *cvex* and *ecce* are allowed variables to both represent object shapes. Such an "over-representation" of the featured variables allows us to use BNs to identify the importance of, and dependencies between these variables in the scenarios of robot grasping tasks.

TABLE I

Used features with their dimensionality D (for continuous) and number of states N after discretization.

	Name	D	N	Description
T	$task$	-	4	Task Identifier
O_1	*obcl*	-	6	Object Class
O_2	$size$	3	8	Object Dimensions
O_3	$cvex$	1	4	Convexity Value $[0, 1]$
O_4	$ecce$	1	4	Eccentricity $[0, 1]$
A_1	dir	4	15	Approach Direction (Quaternion)
A_2	pos	3	12	Grasp Position
A_3	$upos$	3	8	Unified Spherical Grasp Position
A_4	$fcon$	11/20	6	Final Hand Configuration (Armar/Human)
A_5	$egpc$	2	8	Eigengrasp Pre-Configuration (only Human)
C_1	coc	3	4	Center of Contacts
C_2	$fvol$	1	4	Free Volume

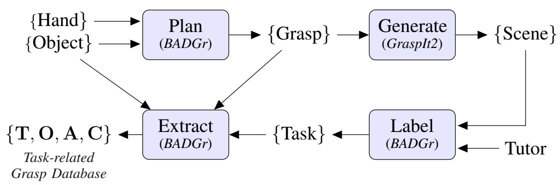


Fig. 2. Schematic diagram for generating task-related grasp database.

IV. Results

A. Experiment I: Structure Learning

The first experiment is to evaluate the network structure. Fig. 3 shows the results of learned DAGs for Armar (left) and human (right) hands. We note that learning the structure from data reveals complicated relationships among these variables, which will otherwise be very difficult to encode by human experts. An initial inspection of the DAGs associated with the different hands confirm our intuitive notion of the dependency relations between the variables. For example, the three action features – dir, $upos$ and pos – are connected with each other because the unified spherical grasp position $upos$ is directly derived from the grasp position pos and the hand orientation dir with respect to the object. And the object class *obcl* determines the three object features $ecce$, $size$ and $cvex$.

We also noticed significant differences in the conditional structures between the two hand models. For instance, Armar hand has pos directly conditioned on $ecce$, whereas human hand does not. The reason might be that the Armar hand has limited kinematics configuration, therefore, when the object is quite eccentric, most stable grasps will have to be generated in the position around the side of an eccentric object, for example, on the handle of a hammer.

Also for human hand, center of contact coc has two parents $task$ and *obcl*, these links are both missing in the Armar hand. This is again explainable when consider the embodiment difference of the hands. The human hand has much more DoFs, and more flexible control in its pre-configuration (the random samples in the 2D Eigengrasp space $egpc$). This allows much more variation in its finger contacts with the object compared with those from Armar hand. As a result, coc which quantified this richer variation allows the learning algorithm to discover its potential relations with the object categories and the task requirements. Similar arguments also apply to the differences in connections around $fvol$, and $fcon$.

B. Experiment II: Task Classification

In this section we evaluate the learned network by their task classification performance. The performace is evaluated based on the testing data that also covers all the object

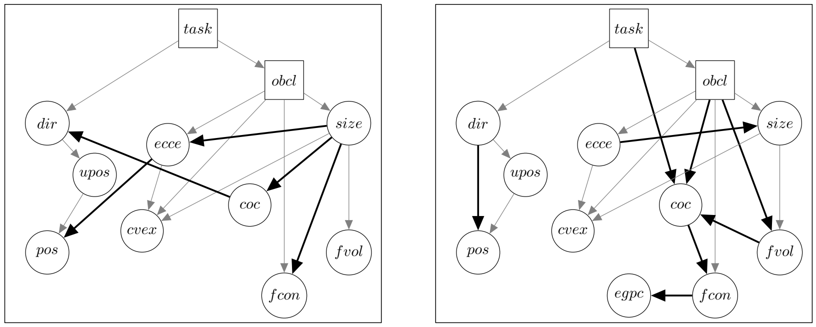


Fig. 3. Experiment I: The resulting DAGs by applying structural learning on (left) Armar hand, (right) human hand data. The differences in DAGs are highlighted by thick arrows. Square nodes represent discrete variables and circled nodes continuous.

	O	A	C	O, A	O, C	A, C	O, A, C
Armar Hand	0.00 0.32 0.33 0.35	0.15 0.09 0.33 0.43	0.00 0.02 0.31 0.67	0.24 0.28 0.26 0.22	0.00 0.37 0.33 0.30	0.33 0.11 0.26 0.30	0.30 0.26 0.20 0.24
	0.00 0.46 0.00 0.54	0.02 0.63 0.02 0.33	0.00 0.00 0.57 0.43	0.02 0.78 0.00 0.20	0.00 0.50 0.00 0.50	0.04 0.46 0.22 0.28	0.02 0.76 0.00 0.22
	0.00 0.00 1.00 0.00	0.00 0.30 0.59 0.11	0.00 0.00 0.78 0.22	0.00 0.00 0.93 0.07	0.00 0.00 1.00 0.00	0.00 0.11 0.89 0.00	0.00 0.00 0.93 0.07
	0.00 0.00 0.13 0.87	0.02 0.07 0.13 0.78	0.00 0.04 0.33 0.63	0.02 0.02 0.13 0.83	0.00 0.13 0.13 0.74	0.07 0.04 0.26 0.63	0.02 0.02 0.13 0.83
Human Hand	0.24 0.35 0.12 0.29	0.54 0.42 0.00 0.04	0.23 0.62 0.15 0.00	0.44 0.28 0.20 0.08	0.28 0.30 0.14 0.28	0.51 0.39 0.07 0.03	0.47 0.28 0.16 0.09
	0.00 0.46 0.00 0.54	0.01 0.94 0.00 0.05	0.01 0.95 0.04 0.00	0.01 0.75 0.00 0.24	0.00 0.49 0.00 0.51	0.01 0.90 0.01 0.08	0.01 0.80 0.00 0.19
	0.62 0.00 0.38 0.00	0.29 0.71 0.00 0.00	0.00 0.51 0.49 0.00	0.28 0.00 0.72 0.00	0.05 0.00 0.95 0.00	0.07 0.44 0.49 0.00	0.04 0.00 0.96 0.00
	0.07 0.00 0.02 0.91	0.38 0.33 0.00 0.29	0.13 0.76 0.11 0.00	0.06 0.01 0.02 0.91	0.01 0.04 0.07 0.88	0.33 0.26 0.07 0.34	0.03 0.05 0.06 0.86

Fig. 4. Experiment II: Confusion matrices for task classification given different observations spaces: permutations of O, A, C features. For each 4×4 matrix, from left to right (top to down), the 3 tasks are: *hand-over, pouring, tool-use, dish-washing*.

classes. The data size is one quarter of the training cases.

As shown in Fig. 4, this task classification is based on the inference of $task$ variable given observation of different set of other variables that form a complete permutation of the 3 feature sub-sets: O, A and C. For object features O, we assume that the object is unknown, therefore object class information $obcl$ is not observed. This is to simulate the real-world scenarios where recognizing object categories from its raw features is still a hard problem for robot sensor systems.

Comparing the task classification given different observation spaces (different columns), we see that for both hands, the object and action features (O, A) result in quite good task classification on the last 3 tasks: *pouring, tool-use* and *dish-washing*; particularly for the Armar hand, the accuracy are 78%, 93% and 83% respectively. When the two constraint features $fvol$ and coc are also observed (O, A, C), we observe overall improvements for human hand, but not so much for Armar. This can be explained by the differences in DAGs where Armar hand has less conditional dependencies discovered with the two constraint variables.

When only object features are observed, both hands have good classification on *dish-washing* task with slight confusion with *tool-use*. This is because in the labeled grasp data, the objects that are good for *dish-washing* are all the mugs and glasses, and one particular knife model (the kitchen knife). But the *pouring* task is never confused with *tool-use* because no tool objects affords pouring, and the observed object features could clearly differenciate the tools from the container objects. However, the grasps that are good for *pouring* is often confused with *dish-washing* becasue many pourable objects are also applicable to be dish-washed.

The *hand-over* task is often confused with others even when most features are observed (column O, A, C). This is expected as grasps that are good for *hand-over* are in many cases also likely to work well for the other three. This indicates that our classification of task might need a hierarchical structure rather than the flat class association we use here.

When comparing the confusion matrices between two hand models, we see that in any observation conditions, the performance over task classification has very different profiles in different hands. This means a variable that is strong in task description for one hand might be weak for another, again supporting the idea of embodiment-specific representation for grasping tasks.

C. Experiment III: Inference on Unified Grasp Position

In Experiment II we showed that the two hand BNs have different but good performances in task classification. The goal of Experiment III is to examine i) if both models could successfully encode task constraint in the continuous space of object observation, and ii) if this constraint is hand-dependent. Notice that the constraint of a given task is often encoded by a combination of multiple features, e.g. one should not grasp this object from this position pos, in this orientation dir, and with this joint configuration $fcon$. However due to space limit and for the purpose of

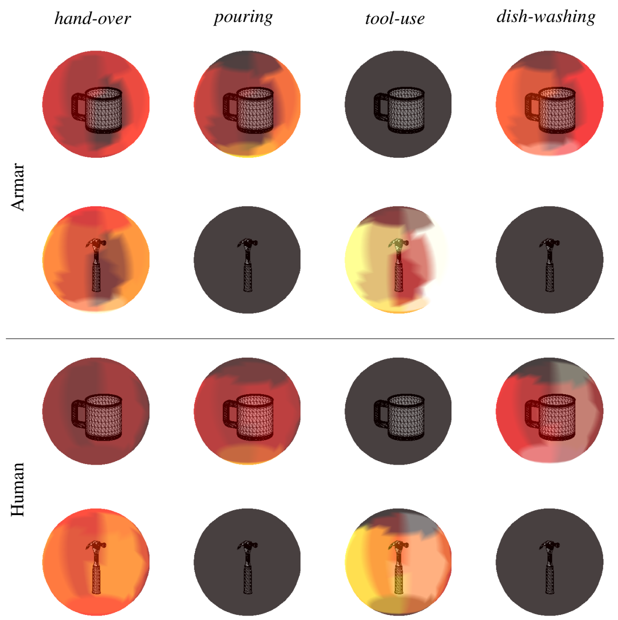


Fig. 5. Experiment III: Likelihood maps of the unified grasp position given tasks and object features $P(upos|T, O)$. The top panel is for Armar hand and bottom panel for human hand.

easier evaluation by the readers, we choose the unified grasp position $upos$ which combines the absolution grasp position pos and approach vector of the hand as an intuitive variable to visualize the task constraint.

For each hand, we sample 625 points on the unified sphere (where $upos$ is located) around the object. As shown in Fig. 5, for each sampled point, the likelihood is obtained given the 4 tasks, and the object features for 2 unknown objects: a mug and a hammer, i.e. $P(upos|T, O)$. The top panel shows the results for the Armar hand, and the bottom for the human hand.

We see that, for both hands, the model sucessfully rules out the mug for *tool-use*, and the hammer for *pouring* and *dish-washing* tasks. For *pouring*, the mug can not be grasped from the top as it will block the opening; similarly, when using the hammer as a tool, the grasp should avoid the head of the hammer as it is the functional part. To wash the mug, the preferred grasps indicated by the network are clearly from side or bottom. This is because the mugs usually need to be placed upside-down in the dishwasher, so grasping from top is not so convenient for this task.

When comparing the likelihood maps between the two hands, we have very interesting observations. In general maps are different for the two different embodiments even though they all model the task constraints in a similar way. In a specific case of *hand-over* the hammer, Armar hand has quite low likelihood on the side of the hammer that is facing the head of the hammer. Thinking closely, we understand that grasping from this position is particularly difficult for the Armar hand because the fingers might contact the sharp edges on the hammer head, resulting in unstable configuration. Similar situation is also for grasping from the top approaching the hammer head. On the contrary, human hand has much more uniformed distribution around the hammer.

This experiment again demonstrated the strength of the proposed framework: by modeling the embodyment-specific task space using a probabilistic network, we have learned not only the affordances of the objects based on its basic 3D features, but also the robot's own motor capability.

D. Experiment IV: Goal-directed Imitation

Finally we would like to demonstrate the application of the proposed framework in the scenarios of goal-directed imitation. The experiment is implemented using the human hand model as the demonstrator, and the Armar hand as the imitator. The goal is to imitate the demonstrator performing the *pouring* (demo 1) and *dish-washing* (demo 2) tasks using a mug (see Tab. II), and the *hand-over* (demo 3) and *tool-use* (demo 4) tasks using a hammer (see Tab. III). The object images in step 1 and step 2.1 shown in both tables are presented with same scale, so the size of the objects can be compared. We use $\mathbf{o}^H, \mathbf{a}^H, \mathbf{c}^H$ to indicate the human demonstrated object, action and constraint features respectively, $\mathbf{o}, \mathbf{a}, \mathbf{c}$ to represent the instances of the features of the Armar hand.

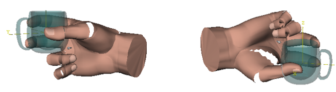
The process of the imitation consists of two major steps: step 1 for task recognition, and step 2 for object or action selection, the same way as we outlined in [5]. Briefly, in step 1, the robot uses the human hand-specific network to recognize the demonstrated task $\hat{t}^H$ based on maximum-log-likelihood estimation $L^H(\ t \mid \mathbf{o}^H, \mathbf{a}^H, \mathbf{c}^H)$, where L^H denote log-likelihood using human network. In step 2, given this recognized task $\hat{t}^H$ as the goal, the robot choose the object among the ones in the scene, and then select the most compatible grasp on the chosen object to achieve the task. Object and action selection has been formulated as the Bayesian decision problems, where a reward function is a weighted combination of their *task affordance* represented by the likelihood function L^R and the similarity to the demonstration S. The weight λ is a high-level control input to define the imitation requirements. Due to space limit, we refer the reader to [5] for the detailed formulation of the Bayesian decision problem and the confidence-based similarity metric.

Tab. II and III present the results of the imitation experiment. The bar plots on the right side of the tables show the log-likelihood values for step 1, and the reword functions in step 2.1 and 2.2. We see that in all four demonstrations, the robot could correctly recognize the tasks, even though there might be potential confusion with *hand-over* task (in demo 1 2 and 3). In demo 3, we find an interesting result where the grasping on the hammer has returned the zero probability for *tool-use*, and low but non-zero probability for *pouring* and *dish-washing*. Apparently unintuitive, but the result is consistent with what we have observed previously [5]: since we assumed unknown object, the inference was only based on observation of object $size$, $cvex$ and $ecce$ features, the network has confused a hammer with other container objects like bottles and glasses.

TABLE II

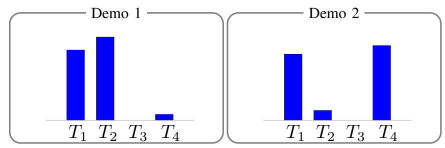
EXPERIMENT IV: GOAL-DIRECTED IMITATION ON *'pouring, dish-washing'* TASKS.

Scenes	Objective Functions
Step 1 Human demonstration: recognize task $\hat{t}^H$	$L^H(\,t \mid \mathbf{o}^H, \mathbf{a}^H, \mathbf{c}^H)$

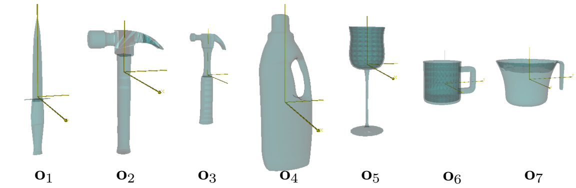


Demo 1   Demo 2

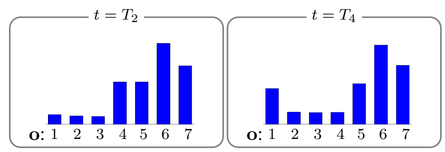
T_1 = *hand-over*
T_2 = *pouring*
T_3 = *tool-use*
T_4 = *dish-washing*



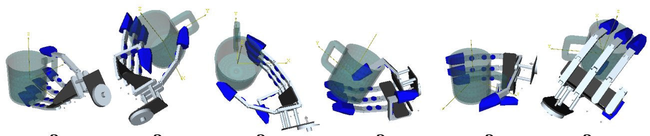
Step 2.1 Select object $\mathbf{o}^*$: matching $\hat{t}^H$, and also similar to $\mathbf{o}^H$ — $L^R(\,\mathbf{o} \mid t) \cdot 0.5 + \, S(\mathbf{o},\, \mathbf{o}^H \mid t) \cdot 0.5$



$\mathbf{o}_1$ $\mathbf{o}_2$ $\mathbf{o}_3$ $\mathbf{o}_4$ $\mathbf{o}_5$ $\mathbf{o}_6$ $\mathbf{o}_7$



Step 2.2 Select action $\mathbf{a}^*$: matching $\hat{t}^H$, and also similar to $\mathbf{a}^H$ — $L^R(\,\mathbf{a} \mid t, \mathbf{o}_6) \cdot 0.5 + \, S(\mathbf{a},\, \mathbf{a}^H \mid t) \cdot 0.5$



$\mathbf{a}_1$ $\mathbf{a}_2$ $\mathbf{a}_3$ $\mathbf{a}_4$ $\mathbf{a}_5$ $\mathbf{a}_6$

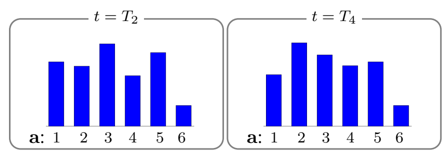
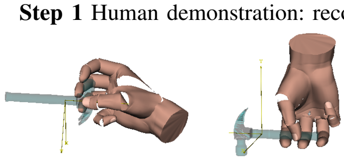


TABLE III

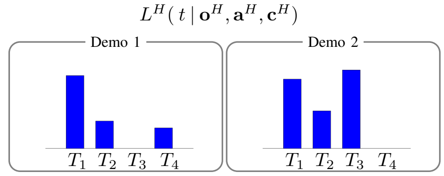
EXPERIMENT IV: GOAL-DIRECTED IMITATION ON *'hand-over, tool-use'* TASKS.

Scenes	Objective Functions
Step 1 Human demonstration: recognize task $\hat{t}^H$	$L^H(\,t \mid \mathbf{o}^H, \mathbf{a}^H, \mathbf{c}^H)$

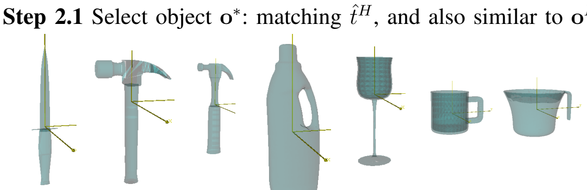


Demo 3   Demo 4

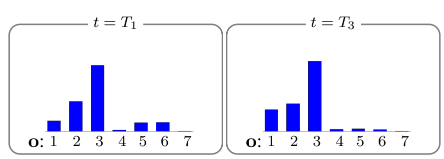
T_1 = *hand-over*
T_2 = *pouring*
T_3 = *tool-use*
T_4 = *dish-washing*



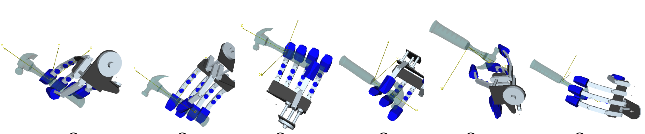
Step 2.1 Select object $\mathbf{o}^*$: matching $\hat{t}^H$, and also similar to $\mathbf{o}^H$ — $L^R(\,\mathbf{o} \mid t) \cdot 0.5 + \, S(\mathbf{o},\, \mathbf{o}^H \mid t) \cdot 0.5$



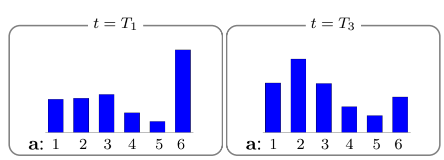
$\mathbf{o}_1$ $\mathbf{o}_2$ $\mathbf{o}_3$ $\mathbf{o}_4$ $\mathbf{o}_5$ $\mathbf{o}_6$ $\mathbf{o}_7$



Step 2.2 Select action $\mathbf{a}^*$: matching $\hat{t}^H$, and also similar to $\mathbf{a}^H$ — $L^R(\,\mathbf{a} \mid t, \mathbf{o}_3) \cdot 0.5 + \, S(\mathbf{a},\, \mathbf{a}^H \mid t) \cdot 0.5$



$\mathbf{a}_1$ $\mathbf{a}_2$ $\mathbf{a}_3$ $\mathbf{a}_4$ $\mathbf{a}_5$ $\mathbf{a}_6$



In step 2.1, the robot is able, in all four demonstrations, to choose among seven objects the one that matches the goal of the task $\hat{t}^H$ and at the same time is also similar to the object used by the human hand. In Tab. II we see the network preferred the smaller mug $\mathbf{o}_6$ that is similar size to the mug in the demonstration in both *pouring* and *dish-washing* tasks. In *dish-washing* task, the knife $\mathbf{o}_1$ has almost as high reward value as the glass $\mathbf{o}_5$. This is because one kitchen knife in the knife category affords *dish-washing*.

Finally in step 2.2, the robot successfully selected the grasp hypotheses that satisfy the requirements on task affordance and grasp similarity. In *pouring* task, grasp $\mathbf{a}_6$ has lowest ranking, which is obvious as three fingers block the cup opening. Grasp $\mathbf{a}_5$ is a very natural configuration for the *pouring*. But it is ranked as the second best grasp because compared to $\mathbf{a}_3$, $\mathbf{a}_5$ is less similar to the demonstrated grasp. Similar behaviors have been observed in other 3 demonstrations.

V. Conclusion

We have proposed a unified probabilistic framework to represent the embodiment-specific grasping tasks. The framework consists of a discrete Bayesian network and the sparse GP-LVM-based multi-variate discretization method. The Bayesian network models the task constraint through conditional distributions among a set of task, object, action and constraint variables. The discretization model provides compact, efficient data representations that allow fast learning and inference for the Bayesian network. With the simulated data from a human and a robot hand, we have shown that the grasping tasks are hand-specific, and the differences are reflected both in the conditional (in)dependencies between the representation variables (network structure), and in the probabilistic distributions of individual variables. However, both models perform well in task classification and representation of the underlying constraints.

We also showed that the hand-specific task representation can provide a unified framework for many aspects in scenarios of goal-directed grasp imitation. Not only can the robot recognize the intention of the human demonstration, but it can also reason in the low-level feature space of the object and grasp actions conditioned on the high-level task requirements. As a result, the robot can make automatic decisions that satisfy multiple user requests, for example, task affordance and grasp similarity.

Though in this paper, the proposed framework was only experimented with one grasp planner [16], we want to emphasize that it is not limited to any specific grasp planning systems. Several grasp planners can provide different representations of grasps and objects, and together with a human-provided task information, we could obtain similar task constraint models for each hand. In the cases the two grasp planners can provide similar grasp-related variables, we expect that the model trained on one planner could be used to infer task information on the other. This is to be tested in one of the next steps in the future research.

In addition, there are also some limitations in discretization model that need further research. Currently, the number of discrete states are manually chosen to satisfy a trade-off between refined data representation and complexity of BNs. In the future, we would also like to learn this hyper parameter automatically from data.

Finally, we plan to test this framework in grasp planning and execution in real robot platforms where sensorimotor uncertainty is more prominant. We believe this will further exemplify the benifits of using a probabilistic model capable of dealing with uncertainty in real-world applications.

Acknowledgments

This work was supported by EU IST-FP7-IP GRASP and Swedish Foundation for Strategic Research.

References

[1] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, “Robot Programming by Demonstration,” in *Springer Handbook of Robotics*, 2008, pp. 1371–1394.

[2] C. L. Nehaniv and K. Dautenhahn, *The Correspondence Problem*. MIT Press, 2002, pp. 41–61.

[3] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, “Learning and generalization of motor skills by learning from demonstration,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2009.

[4] S. Bitzer, I. Havoutis, and S. Vijayakumar, “Synthesising novel movements through latent space modulation of scalable control policies,” in *the 10th International Conference on Simulation of Adaptive Behavior*, 2008.

[5] D. Song, K. Huebner, V. Kyrki, and D. Kragic, “Learning Task Constraints for Robot Grasping using Graphical Models,” in *IROS*, 2010.

[6] D. Song, C.-H. Ek, K. Huebner, and D. Kragic, “Multivariate discretization for bayesian network structure learning in robot grasping,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, May 2011, to appear.

[7] L. D. Fu, “A Comparison of State-of-the-art Algorithms for Learning Bayesian Network Structure from Continuous Data,” Master’s thesis, Biomedical Informatics, Vanderbilt University, December 2005.

[8] N. D. Lawrence, “Probabilistic Non-linear Principal Component Analysis with Gaussian Process Latent Variable Models,” *Journal of Machine Learning Research*, vol. 6, pp. 1783–1816, November 2005.

[9] M. Titsias, “Variational Learning of Inducing Variables in Sparse Gaussian Processes,” in *Artificial Intelligence and Statistics*, 2009.

[10] G. Schwarz, “Estimating the Dimension of a Model,” *Annals of Statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[11] P. Leray and O. Francois, “BNT Structure Learning Package: Documentation and Experiments,” Université de Rouen, Tech. Rep., 2006.

[12] C. Chow and C. Liu, “Approximating Discrete Probability Distributions with Dependence Trees,” *IEEE Transactions on Information Theory*, vol. 14, no. 3, pp. 462–467, 1968.

[13] S. Lauritzen and D. Spiegelhalter, “Local computations with probabilities on graphical structures and their application to expert systems,” *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 157–224, 1988.

[14] T. Asfour, K. Regenstein, P. Azad, J. Schrder, A. Bierbaum, N. Vahrenkamp, and R. Dillmann, “Armar-iii: An integrated humanoid platform for sensory-motor control,” in *Proceedings of the 7th IEEE-RAS International Conference on Humanoid Robots*, 2006.

[15] M. Ciocarlie, C. Goldfeder, and P. Allen, “Dexterous Grasping via Eigengrasps: A Low-dimensional Approach to a High-complexity Problem,” in *RSS 2007 Manipulation Workshop*, 2007.

[16] K. Huebner, “BADGr - A Toolbox for Box-based Approximation, Decomposition and GRasping,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems: Workshop on Grasp Planning and Task Learning by Imitation*, 2010, **URL:**http://www.csc.kth.se/∼khubner/badgr/.

[17] A. T. Miller and P. K. Allen, “GraspIt! A Versatile Simulator for Robotic Grasping,” *IEEE Robotics and Automation Magazine*, 2004.