Using multiple probabilistic hypothesis for programming one and two hand manipulation by demonstration

R. D. Zöllner and R. Dillmann
Universität Karlsruhe (TH)
Institute of Computer Design and Fault Tolerance
Karlsruhe, D-76128, Germany, P.O. Box 6980
zoellner@ira.uka.de, dillmann@ira.uka.de

Abstract—This Paper presents improvements done to a Programming by Demonstration (PbD) system in order to handle complex one and two hand manipulations. In order to do this, functional roles were added to the system's knowledge base. According to them a probability density function expressing the relationship between the manipulated objects has been set up. Since one object can fulfill several functional roles in different contexts, multiple hypotheses are considered. This enables the system to detect in a more reliable way the goals and the subgoals of a human demonstrated task. Further it is pointed out how these goals can be reached by setting up a sequence of elemental actions, how these are generated and represented symbolically. Such a representation is important in order to build up complex tasks consisting of several subtasks and skills. Finally, an experimental setup is presented in which household tasks like laying a table, pouring a glass of water, handling work tools can be understood, learned and generalized by the PbD system.

I. INTRODUCTION

Developing service and personal robots able to perform useful and complex tasks in different environments like households, offices and public environments requires high demand on these robots in terms of interactivity and learning capabilities since these systems are used by non expert persons. Also the perception abilities have to be on a high level in order to understand and deal different situations and intentions of humans in the environment.

The operational field of such robots is spread and dynamically, therefore programming these systems requires at least the ability of adapting parameters but also teaching new skills and task. This can be done on various levels which imply different requests on the interaction, perception and knowledge base of the system.

In the near past so called “easy robot programming systems” were presented, with different ability in terms of perception, interaction and learning. Regarding only on the systems which try to form humans, all of them have very hard constraints in terms of understanding human actions and are restricted on Pick & Place operations.

The following article describes the improvements made on an existing PbD system in order to extend its capabilities form handling only Pick & Place actions to more general manipulation like serving a glass of water performed with one or two hands. The next section II presents related approaches that have been developed by others and earlier works that have been carried out at our institute. Then the perception of human demonstration and some aspects of modeling objects are briefly presented in section III, since these are used for the developed methods. A two step fragmentation and analyze method for segmenting a demonstration into sets of key points, based on probabilistic hypothesis is described in section IV. Then based on this hypothesis set the building of actions sequences and their symbolic representation is pointed out in section V. Finally in section VI the experimental setup used for the evaluation and the future work is presented.

II. RELATED WORK

Several Programming systems and approaches based on human demonstrations have been proposed during the past years. Many of them address special problems or a special subset of objects only. An overview and classification of the approaches can be found in [1].

Often, the analysis of a demonstration takes place observing the changes in the scene undertaken by the user. These changes can be described using relational expressions or contact relations [6], [5], [8]. For generalising of a single demonstration mainly explanation based methods are used [7], [4]. Those allow for an adequate generalisation taken from only one example (One-Shot-Learning). Approaches based on One-shot-learning techniques are the only ones feasible for end users since giving many similar performing examples is an annoying task.

Besides a derivation of action sequences from a user demonstration direct cooperation with users has been investigated. Here, user and robot reside in a common work cell. The user may direct the robot on the basis of a common vocabulary like speech or gestures [12], [11], [10]. But this allows for rudimentary teaching procedures only.
A general approach to programming by demonstration together with the processing phases that have to be realized when setting up such a system is outlined in [2]. A discussion of sensor employment and sensor fusion for the action observation can be found in [9], [14], [3]. Here, basic concepts for the classification of dynamic grasps is presented as well.

III. PERCEPTION OF HUMAN DEMONSTRATION

For the better understanding in this section a brief overview about the used sensors, the sensor processing methods and abstraction is given. For details please refer to [13].

A. Sensory Setup

For acquiring human demonstration a vast sensorial input is needed. Since this work is restricted on manipulation tasks the used sensors are focusing on hand and finger actions. For this purpose a data glove combined with a magnetic tracker is used in order to gather 6 DOF hand positions and 22 DOF hand poses by measuring the finger joint angles. Furthermore a glove with 12 tactile sensors was developed to collect force values applied to manipulated objects during the manipulation.

![Fig. 1. Used perception sensor setup containing two datagloves with force magnetic position sensors and an active stereo vision system.](image)

For understanding complex manipulation tasks the information about the hand actions is not enough, in fact the produced effects by manipulating objects in the environment has to be implemented too. For realizing this objective an active stereo camera head is tracking the working environment observing objects and hands.

B. Processing Sensory Data

Several sensor fusion and signal preprocessing methods have been implemented in order to extract robust information from sensory data [3]. Furthermore statistical learning methods like neuronal networks and support vector machines were trained in order to classify static and dynamic grasps [13].

As a result of the processed sensory data the following high level information is extracted and passed to analyzing methods of the demonstrated task:

- trajectory and velocity of hand and objects movements
- contact points and force progression during a manipulation
- static grasps (the hand pose remains unchanged during a object is grasped)
- dynamic grasps (a grasped object is move by changing the hand pose e.g. unscrewing a lid)

C. Environment model

Because the presented system learns form one single demonstration an adequate environment representation has to be chosen. Therefore a frame based representation like Minsky’s of objects and environment has been implemented in which object types, including geometrical features, grasping configurations etc. are stored. This representation was sufficient for handling Pick & Place actions but considering detection of common tasks with various kinds of operations more information has to be included in the object representation to unresolved ambiguities. Hence a set of functional roles was added to the object types.

Like shown above a Cup has beside his geometric and manipulative features three different functional roles which can occur. Some of the roles are fixed e.g. a cup can be manipulated and others like fill in and pour out are can be computed from the actual state of the cup. These state depended roles can change over the time, giving the system the opportunity to react more reliable in specific situations.

Due to the fact that human demonstration in terms of manipulation produce changes in the environment, these have to be modeled in an adequate way. Therefore the state of the whole environment is represented by a set of
different relations. So far two different kinds of relations are distinguished:
- unity relations (object related) \{type of, functional roles, consists of,\}
- two dimensional relations (geometric relations) \{is above, is under, near, part of ...\}

An environment representation for a certain point on the timeline is done by the Conjunctive Normal Form (CNF) of all fulfilled relations. In [4] an information theory based algorithm is presented for selecting a context relevant subset of relations in order to reduce computational time for calculating actual states.

IV. ANALYZE AND FRAGMENTATION

After processing sensory data, the next step consists in fragmenting the demonstration into time related key points, which are serving for triggering the evaluation of elemental actions. This is done in two phases; first segmentation is performed according to grasp and ungraps actions and second a probabilistic segmentation function is used for generating hypothesis of further key points. One main difference between these two segmentation methods is the fact that the first one segments the preformed actions of each hand independently and the second one is depending of geometrical and object feature and role information.

A. Grasp segmentation

Since this segmentation part was described in [13] here only a brief overview is given. A threshold based algorithm using force values and hand velocities and poses is used for selecting key points. These are triggering the static and dynamic grasp classifiers in order to get the grasp type. As a result of this fragmentation method for each hand a trace of grasp and ungrasp phases is generated.

This step of fragmentation is sufficient for detecting and representing Pick & Place tasks but it lacks the facility of detecting coordination of both hands and tool handling tasks. Therefore a second step has been introduced in the system.

B. Probabilistic segmentation

For detecting more then Pick & Place action there is need to detect possible interaction between two objects or between one or two hands and objects without the restriction of grasping or ungrasping an object. Looking for instance at the task of “pouring a glass of water” grasping the glass with the left hand and the bottle with the right hand can be easily detected. But the understanding of the relationship between the two objects and consequently the two hands has to be performed. In terms of “pouring out water” there is a relation between the bottle and a glass without any contact. Since the sensorial input about the running water is missing the knowledge about the functional role of the objects, the object types (included in the world states) and of course the movement descriptions have to compensate this lack.

1) Role dependent probability density function : A probability density function \(P(o_1, o_2)\) (1) has been set up in order to describe the probability of some (generic) relations between two objects \(o_1\) and \(o_2\).

\[
P(o_1, o_2, r) = f_s(\psi, r, t) \cdot e^{-\frac{d(o_1, o_2)}{\sigma}} \quad (1)
\]

\[
\tau(t_2, r) > 0 \text{ and } 0 < f_s(\psi, r, t) \leq 1
\]

The function \(d\) denotes the distance between two objects:

\[
d(o_1, o_2) = \|pos(o_1) - pos(o_2)\|^2 \quad (2)
\]

\(\tau(t_2, r)\) adapts \(P(o_1, o_2)\) (1) to the object type \(t\) and its role \(r\). For now \(\tau(t_2, r)\) is implemented as an empirical discrete function.

\[
f_s(\psi, r, t) = (1 - k \cdot e^{-\frac{\psi^2}{\sigma^2}}); 0 < k \leq 1 \quad (3)
\]

In order to include the dependency of the density function \(R\) the term expressed in (3) has been setup. The function \(f_s(\psi, r, t)\) reacts on variation of the velocity \(v\) of moved object by taking the first derivation of the velocity. Again the term \(\tau(t_2, r)\) is used to adapt \(f_s\) according the objects properties like type and role. The factor \(k\) is constant and denotes a damping of \(f_s\).

Figure 2 shows a scenario with different vessels like a glass, a cup, a bowl and some objects like a saucer, a bottle lid and a wood workpiece which don’t serve the role “pour in”, assumed for \(P\). The graphs are showing the variation of the probability density function \(P\) according to the distance of the bottle to the objects glass and in the second picture to the cup. Hence the distance to all vessels is bigger than in the first picture, all probabilities are scaled down. Further for the non vessel types saucer, workpiece and bottle lid very small probability values are computed.

2) Hypothesis generation according to the role correlation: As mentioned an objects role is determined by a set of possible roles.

Let \(R_i = (r_1, r_2, ..., r_n)^t\) and
be the role vectors of two objects \( o_1 \) and \( o_2 \), where the indices correspond to the same role e.g. for the role “manipulable” the index value is \( i = 1 \).

The correlation between the two role vectors is expressed by the covariance matrix:

\[
C_{1,2} = \text{Cov}(R_1, R_2)
\]

And the covariance vector

\[
CV_{1,2} = \text{diag}(C_{1,2}) = (c_1, c_2, ..., c_n)^T
\]

contains the correlation coefficients according to the associated roles.

The result of multiplying \( CV \) with the probability density function \( P \) is a set of density functions \( H \) which denote possible relations of two objects according to their context roles.

\[
H(o_1, o_2) = CV_{1,2} \ast P(o_1, o_2, c_i)
\]

In figure 3 the glass can fulfill both roles “pour in” and “pour out”, since it is half-full. Hence two hypothesis for this situation are generated one for the scenario that one will pure in water in a glass (left picture) and one for the situation that one will pour out water from the glass to some other vessels. Next section points out how the hypothesis are resolved.

V. SYMBOLIC TASK REPRESENTATION

In order to describe high level complex task a performed human demonstration is represented as a sequence of elemental operations (EO’s). Since two hand tasks are considered the EO sequences for the left and right hand are conjunct by a two hand EO’s. Examples for EO’s are grasp or ungrasp actions, dynamic grasp acions (srewing...), several kinds of move actions (liner, circular ...), tool handling actions (pouring in ... ) etc. Each EO has a pre- and a post-condition which enables their instantiation. In order to optimize the system in terms of computational time proving this condition is triggered by the fragmenta-

A. Triggering elemental operation

The triggering EO’s pre-condition is performed according different aspects like:

- **Time point of evaluating EO’s condition.** Hereby key points on the timeline are determined form both the grasp segmentation and the probabilistic segmenta-

tion. Because of the multiple hypothesis of the probabilistic segmentation too many key points are generated since the role of one object is unique during performing a EO’s. For this fact the role hypothesis is stored in generated key points serving for the next processing step.

- **Selection of possible EO’s to be proved according to the generated hypothesis.** This is done by selecting the \( m \) (in the actual implementation \( m=3 \)) maximum
role possibility which are greater than a lower probability bound.

- **Searching for two hand EO’s.** Here the assumption is made that during a two hand EO both hands are interacting with one (not necessary the same) object. Hence a search of both (left and right) key point sets is preformed in order to find pairs of corresponding key points. Only on these points two hand EO’s are evaluated.

B. Symbolic pruning: generation of Macro Operators

In order to generate descriptions of complex tasks a symbolic pruning of the generated sequence is performed. This is done by setting up a grammar, in which the EO’s are terminal symbols. According to this grammar a sequence of EO’s are pruned to a Macro operator (MO). Pre- and post-conditions are generated form the EO’s conditions expressed by a Disjunctive Normal Form (DNF). MO’s and EO’s can be pruned to other high level MO’s.

VI. EXPERIMENTAL RESULTS AND FUTURE WORKS

For experiments various objects has been modeled and used in order to teach the system several EO’s and higher level tasks.

![Set of objects used for demonstration](image)

Fig. 4. Set of objects used for demonstration

Figure 4 shows the used objects for programming the system different tasks. In table II the most relevant tasks which have been learned are summarized.

<table>
<thead>
<tr>
<th>Task</th>
<th>Used EO’s / Macro operators</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pick &amp; Place</td>
<td>Grasp, Ungrasp, Move (several kinds)</td>
</tr>
<tr>
<td>Laying Table</td>
<td>Several Pick &amp; Place</td>
</tr>
<tr>
<td>Screwing</td>
<td>Unscrewing (a Bottel lid, a screw)</td>
</tr>
<tr>
<td>Pouring out water</td>
<td>“Pour Out”, Move</td>
</tr>
<tr>
<td>Coordinated opening a bottle</td>
<td>Pick, Place, Unscrew, Move</td>
</tr>
<tr>
<td>Pour a glass of water</td>
<td>Pick &amp; Place, Unscrew, Pour Out, Move</td>
</tr>
</tbody>
</table>

Implementing the methods described in this paper some lacks related to the computational time has been discovered. Analyzing complex tasks with many involved objects can take several minutes. This lack will be improved in future works. Other future work will deal with the mapping of two hand task to a humanoid robot. Till now mapping strategies were performed only on our “one arm” robot *ALBERT*.

VII. CONCLUSION

This paper describes how an existing programming by demonstration system has been improved by introducing probabilistic strategies in order to analyze of the human demonstration. It was shown that the insertion of knowledge in terms of functional role of manipulated objects is necessary and following multiple hypothesis during the analyze leads to a flexible and more robust recognition. Further it was pointed out how the made improvements are enabling the recognition from several simple actions like “pouring water” up to coordinated two hand manipulations like “opening a bottle”.

VIII. ACKNOWLEDGMENT

This Project has been partially supported by the BMBF Project MORPHA and the SFB588 funded by the German Research Association.
IX. REFERENCES


