Extracting Semantic Rules from Human Observations

Karinne Ramirez-Amaro[1], Michael Beetz[2] and Gordon Cheng[1]

***Abstract*— With the recent advancements of sensory technologies (such as Kinect), perceiving reliably basic human actions have become tenable. If robots were to learn or interact with humans in a meaningful manner, the next foreseeable challenge to face robotic research in this area is toward the semantic understanding of human activities - enabling them to extract and determine higher level understanding. In this paper, we present a new methodology that account for the extraction of observed human behaviors with an estimation of the intended activities, follow by the automatic generation of action rules for the synthesis of robot behaviors. Furthermore, we will show the enhancement of the semantic representation with our reasoning system. It is important to mention that the obtained rules are preserved even when different kinds of kitchen scenarios are observed. In order to test the robustness of our results, we used three different kitchen activities: making a pancake, making a sandwich and setting the table. Moving beyond the state-of-the-art in imitation learning, ontology of behavioral rules from human observations can provide more powerful tools for the robots to learn from humans.**

I. Introduction

Programming-by-demonstration (PbD) [1] is a powerful and well-established mechanism used widely in the robotics community to teach robots new activities. Nevertheless, one significant challenge of this teaching technique, is to correctly identify and answer the question: *what to imitate?* [2]. The work by Billard et. al. [3], proposed an interesting approach to identify a general policy for learning *relevant* features of the task, in other words, the authors identified what to imitate from a movement by detecting the time-invariants of the demonstrator. A recent approach employ the idea of a library of *dynamic motion primitives* (DMPs), which enables the generalization of DMPs to new situations [4]. This approach takes into account perturbations and includes feedback [5]. A different approach to encode observed trajectories is presented by Takano et. al. [6], where a mimesis model based on the Hidden Markov Models (HMMs) is presented to segment and generate motions trough imitation. Nevertheless, more of these early approaches focused only at the trajectory level, i.e., in the Cartesian and Joint spaces. Which means that they are able to obtain *relevant* parameters to identify and reproduce similar motions to those of the demonstrator, but the system (robot) will not be able to extract the meaning of the motion.

Then, *what do we want the robot to imitate?*: a) similar motion or b) the meaning of the motion. To a large extent, the above-mentioned approaches have been successful in resolving the first issue. However, less effort has been made to answer the second question. Regarding this issue, one noticeable work presented by Kuniyoshi et. al. [7], proposed a solution to map between the continuous real world events and the symbolic concepts. In a similar work [8], a (partially) symbolic representation of manipulation strategies to generate robot plans based on pre- and post- conditions is presented. Nevertheless, those frameworks are not able to either reason about the intentions of the users or extract the meaning of the action. Another work that address the latter problem, is presented here [9], where a logic sub-language is presented to learn specific-to-general event definitions by using manual correspondence information.

In this paper, we will present our first approach to successfully identify and extract the meaning of human motions by automatically generating rules that define and explain human motions. Those rules will be preserved even in different scenarios. Later, we introduce our reasoning engine based on a ontology semantic representation in order to infer new relationships between actions and objects. An example of our proposed framework can be depicted in Fig. 1. For this approach the acquisition of knowledge represents a key factor. To this end, several sources have been proposed, e.g. the web information and natural-language instructions [10] or annotated videos [11]. We will use annotated videos as our source of knowledge information.

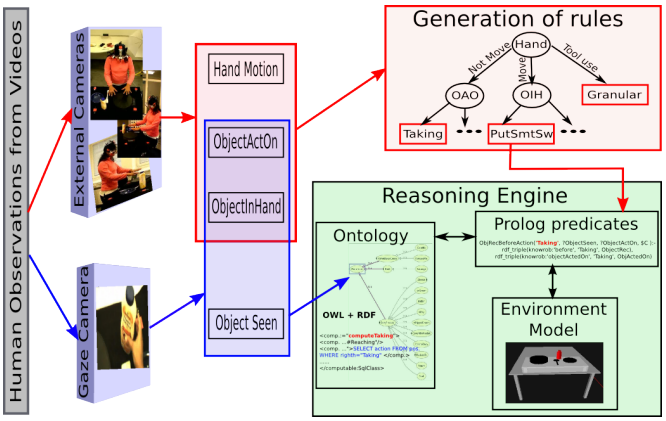

Fig. 1. This figure shows the overview of the approach proposed in this work.

II. Identification of Human Motions

Over the past years new methods of classifying human motions have been proposed, for example: Conditional Random Fields (CRF) [12], Dynamic Time Warping [13], or

[1] Faculty of Electrical Engineering, Institute for Cognitive Systems, Technical University of Munich, Germany ramirezk@in.tum.de and gordon@tum.de

[2] Institute for Artificial Intelligence, University of Bremen, Germany beetz@cs.uni-bremen.de

with Classification and Regression Trees [14]. These earlier techniques realise on generation of trajectories depending on the location of the objects, and if a different environment is being analyzed then trajectories will altered completely, thus, new models have to be acquired for the classification.

In this work, we propose a new method to recognize the human activities based on an abstract layer. This abstraction method does not directly attempt to classify human activities, but rather, it infer the activities based on the observed human motions and the information of the object of interest. To achieve this goal, we will combine information from the environment and information of the human motions. We employ annotated video information[1] to extract the primitive human motion and objects information.

Three primitive human motions are labeled in the videos:

- Move: Defines any motion of the hand.
- Not Move: Means that the hand is not moving.
- Tool-Use: Represents a more complex motion, which involves two objects, one is used as a tool and the second is the object that receives the action, for example: pouring or cutting something.

Additionally, the information of the objects involved in the activity is also considered. The possible labels are:

- Object Acted On: It means that the hand is attracted towards an object, in other words is the object that is going to be manipulated.
- Object In Hand: Defines the object that is physically in the hand, i.e. the object which is being currently manipulated.
- Object Seen[2]: Represents the object that the human is looking at.

In the remainder of this paper we will refer to low-level human Activities (such as: Reach, Take, Release, etc) as a set of high-level human motions (i.e. Move, Not Move and Tool Use).

III. Automatic Generation of Rules

In this work we propose two levels of abstraction: the *high-level*, which describes generalized actions such as: move, not move or tool use, and the *low-level* abstraction, which represents the basic human activities, such as: reach, take, release, etc. Our technique uses the information from the *high-level* abstraction, to infer the *low-level* activities. The inference rules are obtained from a decision tree (see Fig. 1, red box) based on the C4.5 algorithm [15]. Decision trees represent a very reliable technique to learn top-down inductive inference rules because of its robustness to noisy data. Also they can be represented as sets of *if-then* rules to improve human readability. The central core in the C4.5 algorithm is to select the most useful attribute to classify as many examples as possible by using the information gain measure:

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{S} Entropy(S_v) \quad (1)$$

where $Values(A)$ is the set of all possible values of the attribute A, and $S_v = s \in S | A(s) = v$ as a collection of examples for S, and the entropy is defined as:

$$Entropy(S) = \sum_{i=1}^{c} -p_i log_2 p_i \quad (2)$$

where p_i is the probability of S to belong to class i.

IV. Reasoning Engine

With the use of the reasoning engine certain facts can be derived, these facts are not explicitly expressed in the ontology or in the knowledge base. For example the rules obtained from the decision trees. These rules will generate new individuals and new relationships between individuals (objects-properties). Those object properties will be obtained from the definition of new computables. Then, the created new instances and new relationships will be added into the ontology, as part of the inferred knowledge base.

The reasoning engine presented in this work uses the Web Ontology Language (OWL), which is an action representation based on logic description as Prolog queries. We used KnowRob [12] as the base line ontology and we incorporated new relationships between objects and actions, and defined new activity classes These relationships provide us the first perspective view of what the human sees while executing an activity. Such information can be obtained from the head-mounted gaze camera and this represents a good source of information, because from this camera, it is possible to experience what the user is focusing his/her attention during performing certain activity.

The contributions of this work regarding the reasoning engine are:

- Description of a new model for the semantic environment, for the pancake-making scenario.
- New classes on the ontology: ClosingABottle, OpeningABottle, Pouring, Flipping, etc.
- Definition of new SQL computables[3] properties: computeObjectActedOn, computeDetectedObject, etc.
- New object properties such as: detectedObject, ObjectActedOn, OnObject, etc.
- New prolog predicates such as: objSeenBeforeAction, actionObjSeen, etc.

The reasoning engine enabled with new capabilities, that help to infer new information and integrate information from external sources such as: the semantic environment model, and the data base (MySQL) where the object information is stored.

[1]This manual segmentation represents a first step towards an automatic functional motion segmentation and will act as a baseline or ground truth for the automatic segmentation.

[2]This object was not taken into account for the generation of rules, but it was used for the reasoning engine as an addition to the ontology.

[3]Computables, are used to obtain the semantic relationships on demand, instead of importing everything from the ontology.

V. Task Examples and Experimental Setup

In order to test the robustness of the generated rules in different scenarios, we use three real-world scenarios: making a pancake, setting the table and making a sandwich. These three activities have different levels of complexity and they involve different objects. This is explained in the next sub-sections.

A. Making pancakes

In our first scenario, we recorded videos of humans making a pancake. This scenario allows to analyze the transitions between sequence of motions into activities, and activities into tasks. These recordings contain one human performing the action nine times. The human motions are captured by three cameras located in different positions (see Fig. 2 top). Additionally, the subject was wearing a head mounted camera, to record his/her gaze (see Fig. 2 bottom).

Fig. 2. Top: shows one view of the 3 external cameras and some examples of human activities using the right hand: 1) Reach, 2) Hold, 3) Pour, 4) Flip, 5) Put smt-smw and 6) Release. Bottom: illustrates the gaze recordings.

B. Setting the table

The second experimental set up, uses videos from the TUM Kitchen Data Set, which contains observations of four subjects setting a table (see Fig. 3). The subjects are performing the actions in a natural way. Some subjects perform the activity like a robot would do, transporting the items one-by-one, other subjects behave more natural and grasp as many objects as they can handle. We could notice certain variations during the executed tasks, which include actions executed in different order.

Fig. 3. The subject is performing the setting the table activity. Example of activities for the right hand: 1) Take, 2) Reach, 3) Put something-somewhere, 4) Open-drawer, 5) Release.

C. Making a sandwich

As a final scenario, we recorded a more complex activity, which is making a sandwich. These recordings also contain the information of three external cameras and the gaze camera. Fig. 4 shows the action which contains several objects and different activities.

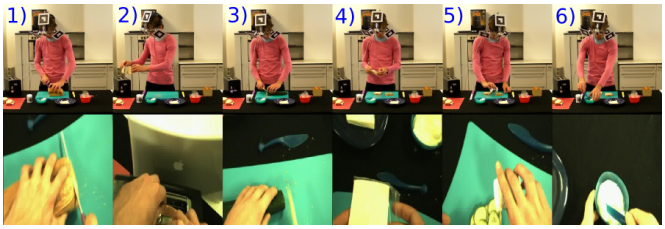

Fig. 4. This figure depicts the action of making a sandwich. Top: shows one view of the three external cameras.A subset of the main activities executed with the right hand are shown: 1) Cut, 2) Put smt-smw, 3) Cut, 4) Unwrap 5) Sprinkle, 6) Spread. Bottom: illustrates the output of the gaze camera.

VI. Results

This section will introduce the obtained results into two subsections. The first will show the rules extracted from human observations and the second will present the new features of the reasoning engine.

A. Automatic generation of rules

The weka data miming software was used to generate the decision tree [16]. We use the labeled information of the motions and objects from the complete pancake-making videos to build the decision tree, which will contain the rules to infer the human activities. The obtained tree is shown in Fig. 5. We would like to stress that a similar tree is obtained if we used as training set the labeled information obtained from the setting-the-table or sandwich-making actions.

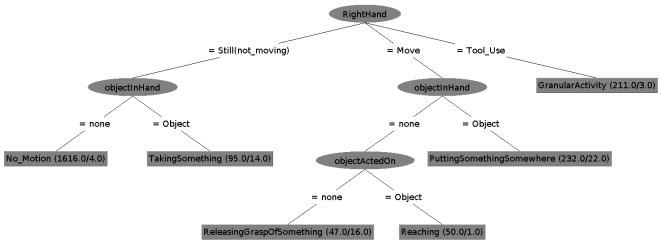

Fig. 5. This figure shows the tree obtained from the pancake making action.

From the above tree the following collection of rules are obtained:

$$if\ RightHand(Move)\ and\ ObjectInHand(None) \\ and\ ObjectActedOn(Object) \rightarrow Activity(Reaching) \quad (3)$$

$$if\ RightHand(Move)\ and\ ObjectInHand(None) \\ and\ ObjectActedOn(None) \rightarrow Activity(Realeasing) \quad (4)$$

$$if\ RightHand(Move)\ and\ ObjectInHand(Object) \\ \rightarrow Activity(PuttingSomethingSomewhere) \quad (5)$$

It is important to notice that the differentiation between "Reaching" and "Releasing" is very challenging because the Cartesian trajectories of both activities are very similar, but if we take into account the objects, as we proposed in this work, then the distinction is possible.

From Fig. 5 we can observed that the activities: Pour, Flip and Slide-out are inferred using the same rule:

$$if\ RightHand(Tool\_use) \rightarrow Activity(GranularActivity) \quad (6)$$

This means that these activities need more information in order to be correctly classified. Those activities does not represent human basic activities, and we will call them *granular activities* and their definition is beyond the focus of this document.

Now, using the obtained tree from the pancake-making activity, we are able to classify the activities involved in the sandwich-making activity, which is more complex and has more instances. The accuracy of the correctly classification of instances is 92.17% and the corresponding confusion matrix can be observed in Table I.

TABLE I
CONFUSION MATRIX FROM THE ACTIVITY OF SANDWICH MAKING

Actual Class \ Classified as	a	b	c	d	e	f
a)Reach	**316**	21	8	3	0	71
b)Take	6	**114**	34	28	0	0
c)PutSomethingSomewhere	13	14	**2034**	30	65	46
d)Release	31	1	6	**248**	0	208
e)Granular	0	4	57	0	**3862**	0
f)No_Motion	83	0	0	80	0	**2951**

Similar to the above procedure, we use the rules obtained from the pancake-making activity to infer the setting-the-table[4] activities. The accuracy of the correctly classification of instances is 91.58%.

The important contribution of these results is the definition of rules that make possible the inference of basic human activities in different scenarios with an accuracy above 90%. This presents the first step towards the generalization of those kinds of activities.

B. Reason engine results

First, we generate a new semantic environment model for the pancake making scenario (see Fig. 1, green box). Second, we define new SQL computables and classes. For example, we can ask the following: *what **object(s)** do I **see** before I **Reach** my **goal**?* . The prolog query will be[5]:

$$objSeenBeforeAction('Action', ?ObjectSeen, ?ObjGoal) :-$$
$$rdf\_triple(_{knowrob}:'before','Action', ?ObjectSeen),$$
$$rdf\_triple(_{knowrob}:'objectActOn','Action', ?ObjGoal).$$

where *Action* is replaced by *Reaching*, then the output will be: $?ObjectSeen = Spatula$ and $?ObjGoal = Spatula$ or $?ObjectSeen = Pancake$ and $?ObjGoal = Pancake$ or $?ObjectSeen = Spatula$ and $?ObjGoal = Pancake$, etc. This means that in most of the cases, we first look at the object that we will reach. This represents an important contribution because this new information will help to decrease the search space for the perception module, because we could focus on the object that the human is currently seeing and infer that, most probably, it will be the object that is going to manipulate in the close future. Further analysis on this topic is being consider as future work.

[4]The testing data set was not used during the training period.

[5]The presented prolog queries are simplified and they are used only for illustration purposes.

We could also infer from our system, the current activity that the subject is performing when the pancake mix is being manipulated. Also, it is possible to answer if the human focuses his attention to the manipulated object or he is seeing something else:

$$actionObjSeen(?Action,'ObjInHand', ?ObjSeen) :-$$
$$rdf\_triple(_{knowrob}:'objHand', ?Action,'ObjInHand'),$$
$$rdf\_triple(_{knowrob}:'detecObject', ?Action, ?ObjGoal).$$

where $ObjInHand = pancakeMix$ and the outputs are: $?Action = CloseBottle$ and $?ObjSeen = pancakeMix$ or $?Action = OpenBottle$ and $?ObjSeen = pancakeMix$ or $?Action = Pouring$ and $?ObjSeen = pancake$, etc. Therefore, we can observe that most of the time, the people look at the object that he/she is manipulating, but we could also notice that during certain actions, such as pouring, a new object appear (*pancake*) and we most likely will focus our attention on that new object. Those inference rules, could help the robot during the planing process.

VII. CONCLUSIONS

This paper presents two contributions:

- Automatic generation of rules from human observations. Those rules have the important characteristic that they can be used in different scenarios and the accuracy to correctly infer human activities is above 90%. This represents our approach to find rules that could generalize basic human activities.
- We show that by adding new capabilities into the reasoning engine, we will be able to compute new relationships between objects and actions. Therefore, the reasoning engine could be improved, taking into account information from the human gaze.

ACKNOWLEDGMENTS

K. Ramírez-Amaro is supported by a CONACYT-DAAD scholarship.

REFERENCES

[1] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, "Survey: Robot Programming by Demonstration," Handbook of Robotics, 2008.

[2] C. L. Nehaniv and K. Dautenhahn, Eds., *Imitation and Social Learning in Robots, Humans and Animals: Behavioural, Social and Communicative Dimensions*. Cambridge: Cambridge Univ. Press, 2007.

[3] A. Billard, Y. Epars, S. Calinon, S. Schaal, and G. Cheng, "Discovering Optimal Imitation Strategies," *Robotics and Autonomous System*, vol. 47, no. 2–3, pp. 67–77, 2004.

[4] A. Ude, A. Gams, T. Asfour, and J. Morimoto, "Task-Specific Generalization of Discrete and Periodic Dynamic Movement Primitives." *IEEE Transactions on Robotics*, vol. 26, no. 5, pp. 800–815, 2010.

[5] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Movement Imitation with Nonlinear Dynamical Systems in Humanoid Robots." in *ICRA*. IEEE, 2002, pp. 1398–1403.

[6] W. Takano and Y. Nakamura, "Humanoid robot's autonomous acquisition of proto-symbols through motion segmentation," in *Humanoid Robots, 2006 6th IEEE-RAS International Conference on*. IEEE, 2006, pp. 425–431.

[7] Y. Kuniyoshi, M. Inaba, and H. Inoue, "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, 1994.

[8] R. Jäkel, S. R. Schmidt-Rohr, M. Lösch, and R. Dillmann, "Representation and constrained planning of manipulation strategies in the context of Programming by Demonstration." in *ICRA*. IEEE, 2010, pp. 162–169.

[9] A. Fern, J. M. Siskind, and R. Givan, "Learning Temporal, Relational, Force-Dynamic Event Definitions from Video." in *AAAI/IAAI*, R. Dechter and R. S. Sutton, Eds. AAAI Press / The MIT Press, 2002, pp. 159–166.

[10] M. Tenorth, U. Klank, D. Pangercic, and M. Beetz, "Web-enabled Robots – Robots that Use the Web as an Information Resource," *Robotics & Automation Magazine*, vol. 18, no. 2, pp. 58–68, 2011.

[11] D. Gehrig, T. Stein, A. Fischer, H. Schwameder, and T. Schultz, "Towards Semantic Segmentation of Human Motion Sequences." in *KI*, ser. Lecture Notes in Computer Science, R. Dillmann, J. Beyerer, U. D. Hanebeck, and T. Schultz, Eds., vol. 6359. Springer, 2010, pp. 436–443.

[12] M. Beetz, M. Tenorth, D. Jain, and J. Bandouch, "Towards Automated Models of Activities of Daily Life," *Technology and Disability*, vol. 22, 2010.

[13] S. Albrecht, K. Ramirez-Amaro, F. Ruiz-Ugalde, D. Weikersdorfer, M. Leibold, M. Ulbrich, and M. Beetz, "Imitating human reaching motions using physically inspired optimization principles." in *Humanoids*. IEEE, 2011, pp. 602–607.

[14] D. Nyga, M. Tenorth, and M. Beetz, "How-models of human reaching movements in the context of everyday manipulation activities." in *ICRA*. IEEE, 2011, pp. 6221–6226.

[15] R. Quinlan, *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann Publishers, 1993.

[16] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," *SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10–18, 2009.