General Recognition Models Capable of Integrating Multiple Sensors for Different Domains*

Karinne Ramirez-Amaro, Emmanuel Dean-Leon, Ilya Dianov, Florian Bergner and Gordon Cheng

***Abstract*— Allowing robots to recognize activities through different sensors and re-using its previous experiences is a prominent way to program robots. For this, a recognition method needs to be proposed such that is transferable toward different domains independently of the used input sources. One key component for such generalization is the definition of common representations. In this paper, we present a flexible system to extract symbolic representations of the perceived scenario which adapts to different sensors, such as cameras, multi-modal skin, and robot joint data. These symbolic representations are used to generate a semantic reasoning engine to transfer the obtained models among different domains. To validate our system, first, our robot learns basic activities from observing a video for the task *cutting bread*. The extracted symbolic representations are later used as previous experiences to the robot, to allow *on-line* segmentation and recognition of the Kinesthetically demonstrated activities for the new *packing oranges* scenario with an average accuracy of 83%, thus demonstrating the generalization of our method.**

I. Introduction

Autonomous robots are expected to recognize previously learned activities while interacting in new environments as efficient and reliable as possible. One of the major challenges is the different type of sensors that robots have. For example, some robots have access to videos of persons demonstrating everyday activities [1], [2], while other robots use information from virtual environments to learn and interpret the shown activities [3]. In some other cases, we would like to physically interact with the robot to Kinesthetically demonstrate the intended activities [4], [5]. Thus, making evident the need of a learning method to recognize activities independently of available sources of information. A prominent way to tackle this problem is using abstract or symbolic representations to obtain models that are transferable among different robots in different scenarios [6].

The recognition of activities and its transferability among different input sources or domains convey several challenges. For example, Fig. 1 shows at least four different sources of information and scenarios, e.g. single videos, multiple videos, virtual environments and robotic sensors, where different activities are demonstrated. Then, a general system should extract common representations allowing the system to transfer and adapt the learned models among different input data. For instance, if the robot learns to identify the activity *reaching* from videos, then the robot should also recognize this activity via Kinesthetic demonstrations even when a different scenario is perceived and without a training phase, i.e. the obtained models should be transferable when the semantic representations are learned for activities.

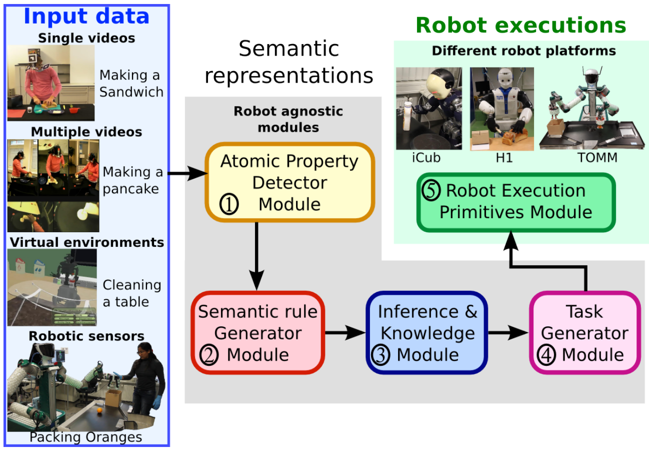


Fig. 1. Overview of our approach to segment and recognize the demonstrated activities using different input data, e.g. videos, robotic sensors, etc.

We propose a hierarchical approach to extract the meaning of demonstrations by means of symbolic and semantic representations. The lowest level of our hierarchical method finds the *relevant* information from the demonstrations from multiple sensors. This obtained information represents the input to the highest level, which infers the demonstrated activities using the automatically extracted semantic representations. The focus of this paper is mainly to demonstrate the flexibility of our system through the obtained general representations to handle multiple types of sensors, e.g. videos or/and robotic sensors (robot skin, robot joints, cameras). Hence, our robot TOMM can recognize the demonstrated activities from external videos or through its own sensors via Kinesthetic demonstrations. This implies that the acquired semantic-based models are robust to different variations on the demonstrations, which is typically the case when two different scenarios are considered, i.e. *sandwich making* and *packing oranges*, even on different robots (e.g. iCub, REEM-C and TOMM, see Fig. 1). This represents another advantage of our system, the generality of the obtained models that can be reused in different scenarios and multiple robots.

This paper presents in Sec. II the related work. Sec. III describes the extensions of our semantic-base system. Sec. IV presents the obtained semantic rules and their transference to Kinesthetic demonstrations followed by the conclusions.

* This work has received funding from the European Communitys Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 609206.

Faculty of Electrical and Computer Engineering, Institute for Cognitive Systems, Technical University of Munich, Germany {karinne.ramirez, dean, ilya.dianov, florian.bergner, gordon}@tum.de

II. Related work

Learning and understanding different activities in different scenarios can greatly improve the transference and generalization of acquired knowledge among different robotic platforms. One learning method to teach robots new activities is via demonstrations [4]. These techniques are mainly used to allow robots to imitate the motions from the demonstrator via low-level copying of human trajectories to map the observed task movements to robots [7], however in order to get the models typically, a manual labeling of the demonstrated activities needs to be performed in advance. Thus, the need for the automatic segmentation and recognition of the demonstrated activities is desired [8], especially if the same activity is demonstrated by different persons leading to small or large variations depending on the expertise of the demonstrator who is manipulating the robot.

In order to obtain general models for the recognition of activities, different levels of abstraction need to be determined to extract meaningful information from the produced task to obtain *what* and *why* a certain task was recognized [9], [10]. This is done using hierarchical approaches, which recognize high-level activities with complex temporal structures [6]. Such approaches are suitable for a semantic-level analysis between humans and/or objects which can be modeled using object Affordances to anticipate/predict future activities [11], or using Decision Trees to capture relationships between motions and object properties [10], or using Graphical Models to learn functional object-categories [12].

For example, [13] suggests to use a library of OACs (Object-Action Complexes) to segment and recognize an action using preconditions and effects of each sub-action which enables a robot to reproduce the demonstrated activities. However, this system requires a robust perception system to correctly identify object attributes which are obtained off-line. Yang et. al. [14] introduced a system that can *understand* actions based on their consequences, e.g. split or merge. Nevertheless, this technique needs a robust active tracking and segmentation method to detect changes in the manipulated object, i.e. the consequences of the action. Aksoy et. al. [2] presented the called *Semantic Event Chain* (SEC) to determine interactions between hand and objects, expressed in a *rule-character* form and it is extended to incrementally learn semantics of manipulated actions [15].

The need for a flexible and general learning component is evident especially when a physical interaction with a robot is expected. An ideal solution would be a scenario where the movements of the robot are segmented and recognized while Kinesthetically demonstrating a new process. This is a challenging topic of research of recent years; in addition, from an industrial point of view, these methods are still not robust enough [16]. Therefore, in this paper, we offer our first attempt to solve these problems.

III. Semantic-based Hierarchical Method

We present the extension of our system which adapts to new robot data, allowing the automatic segmentation, recognition, and labeling while demonstrating activities.

A. System description

This system has been firstly introduced in [10], where our iCub humanoid robot extracts the meaning of human activities from videos using our proposed hierarchical approach. Later, we extended our system by including the activity recognition of both hands at the same time as presented in [17]. After that, we further extended our system to add robustness to different demonstration styles during the *cutting the bread* task especially for co-manipulated activities [18]. From our previous work, we obtained semantic-based models from observing human demonstrations using videos as input sources for cooking scenarios, as depicted in Fig. 2. The main goal and contribution of this paper are to demonstrate the generalization and transferability of the obtained semantic-based models when having different input data, e.g. robotic information. Hence, the learned semantic models remain the same since we aim to validate their robustness during online demonstrations when the system has no knowledge about the activities different persons will Kinesthetically demonstrate in a new scenario, i.e. no data from this scenario or robot sensors has been used to build the semantic-based model.

A key component of our system is the definition of common representations to adapt to different incoming data either videos or robotic information, thus permitting the re-usability of previously learned semantic-based models. With this new information, the system is able to segment and recognize Kinesthetically demonstrated behaviors on the robot by reusing its past experiences. Then, the obtained activities are used to create tasks, which can later be used to execute the learned activities by the robot.

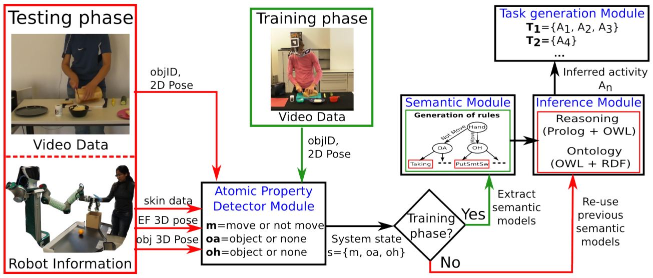


Fig. 2. Overview of our proposed system to handle robot and human information in a common framework.

Fig. 2 shows four modules of our system, *atomic property detector module*, *semantic module*, *inference module*, and *task generator module*. Since we are proposing a hierarchical approach, then we need to define two levels of abstraction.

- The first one, *low-level*, converts the perceived continuous data into symbolic representations, i,e. atomic hand motions (m) such as: *move, not move* and *tool use*, as well as *basic* object properties, e.g. *ObjectActedOn* (a_o) and *ObjectInHand* (o_h). This information is obtained in the *atomic property detector module*, and it is considered as the *state of the system* (s).
- Whereas, the second, *high-level*, handles the difficult problem of interpreting the perceived data into meaningful classes, e.g. *reach, take,* or *cut,* by extracting

semantic rules (*semantic module*) which are used in our reasoning engine (*inference module*).

Then, the sequence of recognized activities are consecutively stored to define tasks and this is done in the *task generator module*. We consider the following vocabulary in the rest of the paper. `Motions` represent atomic movements of end-effectors. `Activities` are semantic descriptions of *motions* and object properties, e.g. *reach, take, cut*, etc. `Tasks` are ordered combinations of *activities*, for example, the task "place orange in box" is composed of the sequence of *activities*, $\{$*take, putSomethingSomewhere* and *release*$\}$.

B. Atomic Property Detector Module

In our previous work [1] we used videos as input information, which means that the *atomic property detector module* received as input the object ID and the 2D positions of the tracked objects including the analyzed hand of the human (see Fig. 2). In this work since we are considering different sources of information, such as multi-modal skin, robot joints, and visual information, we needed to enhance the *atomic property detector module* to handle this new and different data in order to extract *low-level* features.

We adapt this module to consider human or robot motions end-effector (EF) to segment the continuous motions [10]:

- *Move*: the EF is moving, i.e. $\dot{x} > \varepsilon$.
- *Not move*: the EF stops its motion, i.e. $\dot{x} \rightarrow 0$,

where $\dot{x}$ is the EF velocity. Furthermore, we also need to detect the following properties of objects with the help of a simple vision system [19]:

- *ObjectActedOn* (o_a): the EF is moving towards an object, i.e. getting closer to the object, $d(x_{ef}, x_{o_i}) = \sqrt{\sum_{i=1}^{n}(x_{ef} - x_{o_i})^2} \rightarrow 0$.
- *ObjectInHand* (o_h): the object is in the EF location, i.e. o_h is currently manipulated, i.e. $d(x_{ef}, x_{o_i}) \approx 0$.

where $d(\cdot,\cdot)$ is the distance between the EF position (x_{ef}) and the position of the detected object (x_{o_i}) from a common coordinate frame. Note, that the robot cameras have an egocentric view of the scene. Therefore, it is possible to have occlusions, for example when the robot hand is grasping an object which is not longer visible in the camera due to occlusions with the hand and the arm of the robot (see Fig. 4). In such cases, the above properties can not be obtained leading to failures in the recognition similar to [2], [13], [14]. To deal with this problem, we use the information of the proximity and force sensors of the artificial skin attached to the robot [19]. Then, an object that was previously seen, which is not visible anymore, has the *ObjectInHand* property if the average value of the proximity and force sensors attached to the robot's palm is above a certain threshold, this means that the object has been grasped and this can be detected even when the arm moves to another location. Hence, we exploit all available sensors of our robot TOMM.

Since we expect some noise on the signals of the robot and visual features from the cameras, we implement the Butterworth filter to smooth the obtained velocities and distances. The output of this module determines the current *state of the system* (s), which is defined as the triplet $s = \{m, o_a, o_h\}$. Then, we used the perceived *state of the system* (s) to obtain the semantic rules from the *semantic and knowledge module*.

C. Semantic and Knowledge Module

In this work, *the semantics of human activities* refers to find meaningful relationships between hand motions and object properties to infer activities performed by demonstrators. In order to infer the demonstrated activities, we present a two-step semantic-based approach. First, we extract the *low-level* features from the perceived environment (e.g. signals from the sensors handled by the *atomic property detector module*), and as a second phase we automatically generate compact semantic rules to *deterministically* infer the robot activities from the demonstrations (*semantic and knowledge module*). The obtained semantic representations are enhanced with the help of a knowledge-based ontology, which allows a better generalization toward different scenarios.

To obtain the semantic rules, we use the C4.5 algorithm [20] to compute a decision tree (T). This algorithm learns the target function c by selecting the most useful attribute (A) to classify as many training samples (S) as possible. These training samples are composed of the *state of the system* (s), which consists of hand or EF motion segmentation (m) and the object properties (o_a or o_h) obtained from the *atomic property detector module*. The selection of the most useful attribute (A) is done using the information gain measure:

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

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 samples for S. Similar to our previous approach [17], we use the target concept c to recognize basic robot activities:

$$\begin{aligned} &Class\ c : ActivityRecognition : S \rightarrow \{Reach,\ Take,\ Release, \\ &\quad Put\_Something\_Somewhere,\ Idle,\ GranularAct\} \qquad (2) \end{aligned}$$

where *GranularActivity* represents the set of activities that depend on the context. Therefore, to identify such kind of activities a second step is needed as explained in [17].

From the learned decision tree (T) we will obtain *if-then rules*, which are human readable. These rules are enhanced by including them in our knowledge and reasoning engine. In this work, the knowledge-based is defined by an ontology representation expressed in the Web Ontology Language (OWL), and the reasoning is based on Description Logics (DL) such as Prolog queries. We use KnowRob [21] as our baseline ontology and we mainly extended two branches of the KnowRob ontology: *TemporalThings* and *SpatialThings*, where the first one contains the important subclasses of *Actions* and the second describes abstract spatial concepts such as places and *object* classes. An example of the produced Prolog query is given in eq. (5).

D. Task generator module

To learn new tasks we require the input from the user via GUI. With this GUI, we can indicate the robot when a certain task will start and finish. First, we define the name of the task that is going to be Kinesthetically demonstrated. After that, we specify that this task has finished. Then, all the inferred activities will be retrieved by the system and stored as part of the learned task[1], e.g:

$$
\begin{aligned}
T_1 &= \{A_1, A_2, A_3\} \\
T_2 &= \{A_2\}
\end{aligned}
\tag{3}
$$

At this point, the user can delete any activities, if any, that were incorrectly detected by the *semantic and knowledge module* and save the new sequence of activities. This module is also connected to the knowledge-based to store the learned sequence of tasks. The task generation is important since, after the Kinesthetic teaching, the system can later retrieve the learned tasks and execute them in a loop.

IV. Results

We aim to demonstrate the generalization of our obtained semantic models, therefore we present our results in three parts. The first part explains the automatic extraction of semantic models from human demonstrations by observing videos of *making sandwiches*. The second part shows the re-usability of the obtained models even when different input information is used from Kinesthetic demonstrations, for a new scenario of *packing oranges*. Finally, we present the execution of our robot for the demonstrated tasks.

A. Results of extracting semantic rules

In order to extract semantic rules, we randomly select one participant that demonstrates a *sandwich making* from a kitchen data set[2]. Then, we obtained a decision tree using the information of the ground-truth[3] data of the analyzed subject. As an intermediate step, we adjust the ground-truth and use a higher object class information using our previously described ontology. For example, from a sandwich scenario, it is very likely to have objects such as bread and knife. Then, in order to obtain a general tree we use as training input the highest class of objects from our ontology, i.e. *Something*. For example, $(bread \ \& \ knife) \in Something$. Then, the new training samples (S) are:

$$\{Move,\ Something,\ None\} \tag{4}$$

We split the training and testing data as follows: the first 60% of the trails are used for training and the rest 40% for testing, similar to [10]. Then, we obtained the tree $T_{sandwich}$ shown in the top part of Fig. 3 (magenta box) to infer *basic* human activities defined in eq. (2) using general classes of the objects, i.e. *Something*.

Next, we tested the accuracy of the obtained tree $T_{sandwich}$ using the remaining 40% of the data set to validate the robustness of the obtained rules. The obtained results show that $c(n_{sandwich\_test}(t))$ was correctly classified for 92.57% of the instances using as input information manually labeled data, i.e., during the *off-line* recognition.

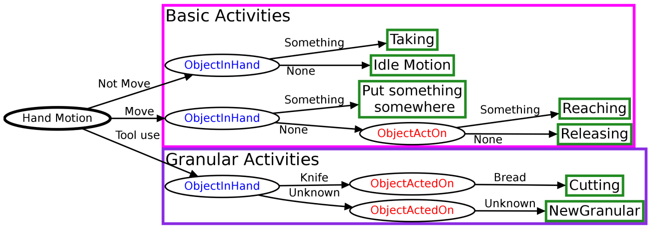


Fig. 3. Tree obtained from the sandwich making scenario $(T_{sandwich})$.

We enhanced our obtained decision tree by including the obtained *if-then rules* in our Prolog reasoning system, e.g. for the activity of *put something somewhere*:

$$
\begin{aligned}
& inferAct(+Motion, +Obj\_AO, +Obj\_IH, ?Activity) :- \qquad (5)\\
& == (Motion, fiadOntology : Move), \\
& rdf\_has(Obj\_IH, fiadOntology : objectInHand, ObjC\_IH), \\
& rdfs\_subclass\_of(ObjC\_IH, fiadOntology : 'Something'), !, \qquad (6)\\
& Activity = 'PutSomethingSomewhere'.
\end{aligned}
$$

where + indicates the input to the system, ? represents the infered activity and *ObjC_IH* is the class of the object with the property of *ObjectInHand*.

The next important step is to extend our system from *off-line* to *on-line* recognition using a Color-Based technique as presented in [17]. Therefore, we use as input the data obtained from the *atomic property detector module* (see Fig. 2) using a different participant than the one used for training. In this case, we perceive the object detected (e.g. bread), which leads to the *state of the system*: $\{Move,\ Bread,\ None\}$[4]. This, however, does not affect the performance of our system, since from eq. (6) we ask if this new object with the property of *ObjectInHand* belongs to the class *Something*. Then, leading to the generalization of our system to untrained objects. Therefore, the result of recognizing activities from both hands at the same time is around 90.64% by re-using the obtained semantic rules $T_{sandwich}$.

B. Robot Kinesthetic teaching results

The next challenge is to test our obtained semantic rules in a completely new environment for a new task of *packing oranges*. To further test the robustness of $T_{sandwich}$, we design the following experiment. Since we enhance the *atomic property detector module* to discretize the continuous information from robotic sensors, we assume that the inference rules obtained in $T_{sandwich}$ will also work when a person is Kinesthetically showing a robot similar activities in a new environment. Therefore, for this experiment there is no training phase, thus allowing the robot to re-use the inferences that it learn from previous experiences as shown in Fig. 4. Then we expect that the *inference module* automatically segments

[1] The generation of new tasks is further explained in the video accompanying this paper.

[2] The used data set is publicly available at the following link http://www.ics.ei.tum.de/ics-data-sets/cooking-data-set/

[3] The ground-truth was manually labeled by a person considered as an expert since this person received a training session.

[4] Note that the perceived *state of the system* can be different than the one used during training, e.g. eq. (4)

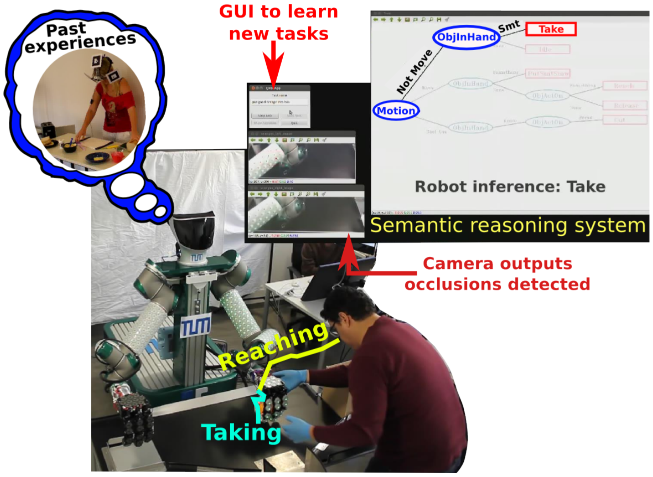


Fig. 4. Kinesthetic demonstrations with our robot TOM. The demonstrated activities are segmented and recognized during the demonstration of the activities for the task of *packing oranges*.

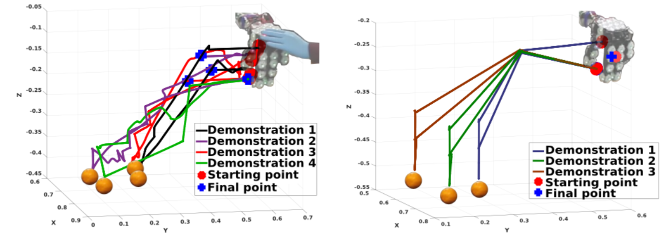
and infers the demonstrated activities *on-the-fly* by re-using the learned semantic models.

Our proposed demonstration has been successfully implemented in our robotic platform Tactile Omni-directional Mobile Manipulator (TOMM), see Fig. 4. TOMM is composed of two industrial robot arms (UR-5) covered with artificial skin, two Allegro hands from SimLab also covered with our artificial skin and 2 cameras on its fixed head used to obtain the 3D position of target objects [19].

In this experiment, additionally to the robot sensor information, we require the visual detection of oranges. For this visual detection, we implement a stereo vision method using two of the robot head cameras. For each camera, we apply the color blob detection technique to recognize oranges. Obtaining a list of detected oranges and their global positions with respect to the robot torso frame. The 3D position of detected oranges has an average error of 0.02m, however as shown in Fig. 4 the recognition of oranges may fail which is compensated by using the information of the proximity and force sensors of the robot's palm which can sense the object.

To quantitatively validate the robustness and generalization of our system, we tested our semantic models $T_{sandwich}$ with different variations on the Kinesthetic demonstrations for the *packing oranges* scenario performed by two different participants[5]. A total of four demonstrations[6] are considered and our system is able to infer Kinesthetically demonstrated activities *on-the-fly*. For these four demonstrations, the position of the oranges in all the experiments is randomly selected. Fig. 5 depicts the obtained trajectories after the Kinesthetic demonstrations where these trajectories are automatically segmented into meaningful activities (see Fig. 4).

By analyzing the obtained trajectories of the robot it is possible to notice some disturbances and variations on these trajectories. For example, from Fig. 5 we can observe that the trajectories obtained from the second demonstration (purple line) and the fourth demonstration (green line) present more variations than the other trajectories. This exemplifies the need for robust algorithms to correctly recognize the demonstrated activities. The overall results[7] of the segmentation and recognition of the demonstrated activities is shown in Table I, where the average of recognition is 83.15%. The following link presents a video with an example of the *on-line* segmentation and recognition: https://youtu.be/o0wfEgzu0mA



a) Demonstrated activities by humans b) Executed activities by TOMM

Fig. 5. a) Obtained trajectories from the Kinesthetic demonstrations on the robot. b) Executed trajectories by the robot of the learned activities. When demonstrating motions on the robot, we expect to see similar trajectories as b) and not as noise as the one obtained from the actual demonstrations a).

TABLE I
RECOGNITION FROM KINESTHETIC DEMONSTRATIONS

Packing oranges	Accuracy of recognition (%)
Kinesthetic experiment #1	85.63
Kinesthetic experiment #2	70.02
Kinesthetic experiment #3	84.88
Kinesthetic experiment #4	92.06

C. Robot execution of the learned tasks

Before the Kinesthetic demonstration starts, the user needs to tell the system that a new task is demonstrated. Then, the user proceeds with the guided demonstration. Once, the demonstration of the new task is finished, the user needs to indicate via GUI the ending of the task. Then, the task generated by the user is automatically stored in the knowledge-based, to allow its later retrieval. In our experiment one user choose the following tasks:

$$T_1\{Pick\_Fruit\} = [Reach(something)] \tag{7}$$

$$T_2\{Place\_Fruit\_inBox\} = [Take(something), \tag{8}$$

$$Put(something, somewhere), Release(something)]$$

where $something = Orange$ and $somewhere = Box$ are going to be instantiated during execution using visual information. Then, if a user wants the robot to perform the learned tasks, he/she just needs to ask the system for the available tasks, which in this case are two, see eq. (7-8). Since our system also saves the order of the learned tasks, the user just needs

[5] One participant was a robotic expert and the other non-expert. We are planning to extend this study to a larger group of participants.

[6] Note that the data from the robot Kinesthetic demonstrations was not used to improve in any way the semantic models $T_{sandwich}$.

[7] The ground-truth is obtained from visual information of the robot cameras, used by each participant to segment and label the taught activities.

to send the execution command to the robot to perform the learned sequence of tasks and activities. For the robot execution, we use a simple state-machine to command robot primitives. In this case, an operational position control was implemented on the robot where the desired position of the hand is defined by the position of the orange. The orientation of the robot hand is constant and it is defined by the initial hand configuration [19], [22]. The trajectories obtained from our robot after executing the learned tasks and activities for *packing oranges* is shown in Fig. 5b), where the orange locations are randomly assigned.

Most of the recognition systems are designed to fit perfectly the studied task, however, most of these systems can not easily allow different input sources [6], [13]. Then, the main advantage of our system is its levels of abstraction which allows developing a general method to transfer the learned models from different domains independently of the used sensors. For instance, our perception module permits the use of different input sources, such as: single videos [10], multiple videos [23], virtual environments [3], and robotic sensors which bootstrap the learning process.

The key components and contributions of our system are: a) The robust adaptation of our symbolic representations to discretize the continuous data from different types of sensors. b) A system that can automatically segment and recognize activities from Kinesthetic demonstrations by reusing the previous experiences applied in a new domain. c) We propose a teaching by demonstration system[8] that can handle different variations on the demonstrations.

V. Conclusions

Finding general recognition models capable of integrating multiple sensors for different domains is a challenging problem. To address this, we proposed a method that uses symbolic and semantic representations, enhanced with a knowledge-based, to automatically segment and recognize the demonstrated activities. In this paper, we demonstrate the robustness and generalization of our obtained semantic models for the recognition of activities using either videos or Kinesthetic demonstrations in different scenarios such as *making sandwiches* or *packing oranges*. Our presented framework has an accuracy of recognition of around 83% when the human is Kinesthetically showing the desired activities to a robot. It is important to highlight that the recognition model was obtained for the *sandwich making* scenario, thus no training phase was performed for the new scenario. Our presented system is adaptable to different input sensors and variations on the demonstrated activities allowing the robot to re-use its previous experiences.

References

[1] K. Ramirez-Amaro, M. Beetz, and G. Cheng, "Transferring skills to humanoid robots by extracting semantic representations from observations of human activities," *Artificial Intelligence. DOI:10.1016/j.artint.2015.08.009*, 2015.

[2] E. E. Aksoy, A. Abramov, J. Dörr, K. Ning, B. Dellen, and F. Wörgötter, "Learning the semantics of object-action relations by observation." *I. J. Robotic Res.*, vol. 30, no. 10, pp. 1229–1249, 2011.

[3] K. Ramirez-Amaro, T. Inamura, E. Dean-Leon, M. Beetz, and G. Cheng, "Bootstrapping Humanoid Robot Skills by Extracting Semantic Representations of Human-like Activities from Virtual Reality," in *Humanoids, IEEE/RAS*. IEEE, November 2014.

[4] R. Dillmann, T. Asfour, M. Do, R. Jäkel, A. Kasper, P. Azad, A. Ude, S. R. Schmidt-Rohr, and M. Lösch, "Advances in Robot Programming by Demonstration." *KI*, vol. 24, no. 4, pp. 295–303, 2010.

[5] S. Calinon, F. D'halluin, E. L. Sauser, D. G. Caldwell, and A. G. Billard, "Learning and reproduction of gestures by imitation: An approach based on Hidden Markov Model and Gaussian Mixture Regression," *IEEE Robotics and Automation Magazine*, vol. 17, no. 2, pp. 44–54, June 2010.

[6] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: A review." *ACM Comput. Surv.*, vol. 43, no. 3, pp. 16:1–16:43, 2011.

[7] D. Lee and Y. Nakamura, "Motion Recognition and Recovery from Occluded Monocular Observations," *Robotics and Autonomous Systems*, vol. 62, no. 6, pp. 818–832, 2014.

[8] A. L. P. Ureche, K. Umezawa, Y. Nakamura, and A. Billard, "Task Parameterization Using Continuous Constraints Extracted From Human Demonstrations." *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1458–1471, 2015.

[9] K. Ikeuchi, M. Kawade, and T. Suehiro, "Toward assembly plan from observation - Task recognition with planar, curved and mechanical contacts," in *IEEE/RSJ IROS*, vol. 3, Jul 1993, pp. 2294–2301.

[10] K. Ramirez-Amaro, M. Beetz, and G. Cheng, "Automatic Segmentation and Recognition of Human Activities from Observation based on Semantic Reasoning ," in *IEEE/RSJ IROS 2014*. IEEE, Sept 2014.

[11] H. S. Koppula and A. Saxena, "Anticipating Human Activities Using Object Affordances for Reactive Robotic Response." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 14–29, 2016.

[12] M. Sridhar, A. G. Cohn, and D. C. Hogg, "Learning Functional Object-Categories from a Relational Spatio-Temporal Representation." in *ECAI*, ser. Frontiers in Artificial Intelligence and Applications, M. Ghallab, C. D. Spyropoulos, N. Fakotakis, and N. M. Avouris, Eds., vol. 178. IOS Press, 2008, pp. 606–610.

[13] M. Wächter, S. Schulz, T. Asfour, E. Aksoy, F. Wörgötter, and R. Dillmann, "Action Sequence Reproduction based on Automatic Segmentation and Object-Action Complexes," in *Humanoids, IEEE/RAS*, 2013.

[14] Y. Yang, C. Fermüller, and Y. Aloimonos, "Detection of Manipulation Action Consequences (MAC)." in *CVPR*. IEEE, 2013, pp. 2563–2570.

[15] E. E. Aksoy, M. Tamosiunaite, and F. Wörgötter, "Model-free incremental learning of the semantics of manipulation actions." *Robotics and Autonomous Systems*, vol. 71, pp. 118–133, 2015.

[16] D. Massa, M. Callegari, and C. Cristalli, "Manual guidance for industrial robot programming." *Industrial Robot*, vol. 42, no. 5, pp. 457–465, 2015.

[17] K. Ramirez-Amaro, M. Beetz, and G. Cheng, "Understanding the intention of human activities through semantic perception: observation, understanding and execution on a humanoid robot." *Advanced Robotics*, vol. 29, no. 5, pp. 345–362, 2015.

[18] K. Ramirez-Amaro, E. C. Dean-Leon, and G. Cheng, "Robust semantic representations for inferring human co-manipulation activities even with different demonstration styles." in *Humanoids*. IEEE, 2015, pp. 1141–1146.

[19] E. Dean, K. Ramirez-Amaro, F. Bergner, I. Dianov, P. Lanillos, and G. Cheng, "Robotic technologies for fast deployment of industrial robot systems," in *42nd IEEE Industrial Electronics Conference (IEEE IECON2016). [Accepted]*. IEEE, October 2016.

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

[21] M. Tenorth and M. Beetz, "KnowRob: A knowledge processing infrastructure for cognition-enabled robots." *I. J. Robotic Res.*, vol. 32, no. 5, pp. 566–590, 2013.

[22] E. C. Dean-Leon, L. G. García-Valdovinos, V. Parra-Vega, and A. Espinosa-Romero, "Uncalibrated image-based position-force adaptive visual servoing for constrained robots under dynamic friction uncertainties." in *IROS*. IEEE, 2005, pp. 2983–2990.

[23] K. Ramirez-Amaro, E.-S. Kim, J. Kim, B.-T. Zhang, M. Beetz, and G. Cheng, "Enhancing Human Action Recognition through Spatio-temporal Feature Learning and Semantic Rules," in *Humanoids, IEEE/RAS*, October 2013.

[8] We validate our proposed system in public demonstrations such as the IROS 2015 conference presenting our robot Kinesthetic teaching method.