Enriched Manipulation Action Semantics for Robot Execution of Time Constrained Tasks

Eren Erdal Aksoy, You Zhou, Mirko Wächter and Tamim Asfour

***Abstract*— This paper contributes to semantic representation of human demonstrated actions for robot execution of time constrained tasks. We propose a semantic action encoding method based on interactions between the subject and objects in the scene. Our semantic framework is enriched with a descriptive spatial reasoning method which leads to accurate segmentation and recognition of unique action primitives. The proposed framework can classify the segmented action primitives as periodic or discrete to allow robots to autonomously decide how to imitate observed actions at different temporal scales. We evaluated our framework on our new large manipulation action dataset which involves in total 70 demonstrations from 8 different action types. We conducted various experiments with our humanoid robot to evaluate the proposed framework.**

I. INTRODUCTION

In cognitive robotics, time perception plays a vital role, in particular to plan and imitate human demonstrated manipulation actions. To approximate the temporal lengths of observed and self-executed actions, the cognitive agent has to be equipped with accurate action segmentation and recognition capabilities. The main challenge is the extraction of inherent characteristics of a human action, because even the same action, e.g. a *pick&place* demonstration, can be performed with different objects by following different motions over variable amounts of time.

Existing methods in the literature tend to invest significant effort to capture the low-level appearance and motion characteristics of actions. Approaches based on action semantics, on the other hand, reveal the inherent characteristics of observed actions, which are invariant to manipulated object types, scene contexts, and followed motions over time.

In this work, we present a novel framework that captures the semantic representation of an action to allow robots to reason about executing observed actions at different temporal scales. The proposed framework is based on our previous semantic action encoding technique, Semantic Event Chains (SECs) [1], which stores patterns of spatial interactions (such as *Touch* and *Disjoint*) emerging between the subject and objects in the scene. These spatiotemporal patterns are invariant to trajectory, object type, and scene context variations. We, here, enrich the SEC concept with a set of more descriptive spatial relations including *inside/cover*, *on/under*, and *above/below*. The enriched SEC representation contributes to more accurate segmentation and recognition of action primitives such as *approach*, *lift*, or *withdraw*. In addition, the enriched SEC concept leads to the categorization of objects based on their roles in the manipulation. This finding can be employed by robots to estimate which objects are the most relevant for the task to be executed.

The proposed enriched semantic representation further lets robots apply additional reasoning on individual primitives. For instance, the robot can autonomously estimate the temporal length and also the type of each motion primitive. At this point, we apply our new trajectory sub-segmentation technique which computes local extrema, i.e. geometrical variations in the trajectory pattern (e.g. curves, straight lines), to identify the main intention in each primitive. By considering the distribution of all derived trajectory subsegments, our method can finally measure the similarity between two primitives and also explore whether the followed motion is periodic or discrete. The periodicity information helps robots to autonomously generate observed trajectories at different temporal scales without altering the characteristic features, such as the action speed. For instance, in the *stirring* action the agent can repeat the derived periodic pattern until meeting the given temporal constraints.

In this work, we also provide a novel manipulation action dataset recorded with a motion capture system. We applied our new enriched semantic perception framework to this dataset which involves in total 70 demonstrations from 8 different action types such as *stir* and *pick&place*. We further conducted various experiments on our humanoid robot to evaluate the proposed framework.

A. Related Work

There exists an extensive literature on topics related to action representation ([2], [3], [4]) and motion execution ([5], [6], [7], [8]) in computer vision and robotics.

Recent works by [2], [3], [4], among others, attempt to represent manipulation actions at the semantic level by considering the role of the manipulated objects. In [2], functional object categories were learned from activity graphs that encode spatiotemporal patterns of object-hand interactions. The previous work of [3] explored a reasoning method that generates semantic action rules by employing abstract hand movements, such as *moving*, *not moving* or *tool used* together with the object information. The work in [4] introduced an event parsing method based on stochastic event grammar by using binary spatial relationships (e.g. *touch* or *near*) between objects and agents in the scene. A more descriptive set of spatial relations was introduced in [9], [10], [11] for object manipulation tasks. Although those semantic ap-

*This work has been supported by the EU FET Proactive grant (GA:641100) TIMESTORM.

The authors are with the High Performance Humanoid Technologies Lab, Institute for Anthropomatics and Robotics, Karlsruhe Institute of Technology (KIT), Germany {eren.aksoy,you.zhou,waechter,asfour}@kit.edu

proaches boost the manipulation classification performance, none of them addresses extraction and execution of trajectory patterns of individual manipulation primitives.

In the context of motion execution, there are various promising methods such as Splines [6], Hidden Markov Models (HMMs) [7], Gaussian Mixture Models (GMMs) [8], Dynamic Movement Primitives (DMPs) [5], [12], [13]. Although these methods can encode and learn different complicated trajectory profiles, the motion type (e.g. *periodic* or *discrete*) must be provided in advance. Furthermore, the continuous trajectory must be initially segmented in the case of executing individual action primitives only. For the trajectory segmentation task, there are several methods which work either in a supervised manner (e.g. based on previously trained DMPs [14] or HMMs [15]) or use unsupervised techniques (e.g. by employing zero velocity crossings [16] or Principal Component Analysis [17]). Those methods, however, omit the action semantics and require either fine parameter tuning or a large training set.

Our research differs from previous works as we employ the manipulation semantics not only for action representation and identification but also for the tasks of trajectory segmentation and classification, all in one single framework. Compared to conventional HMM-based generative action recognition and temporal segmentation approaches, our SEC framework also obeys the Markovian assumption. The only difference is the definition of states which are observable in SECs and represent topological changes in the scene context.

B. Contribution

The main contributions of this paper are listed as follows:

- We introduce a new manipulation action dataset, which has eight action types with in total 70 different demonstrations. Each action has at least 5 different versions demonstrated at different temporal scales using various objects. Compared to existing human motion capture datasets, we, for the first time, provide not only human motions but also motions and models of objects present in the scene. The entire dataset is publicly available.
- We extend our existing semantic action perception framework introduced in [1] with a more descriptive spatial reasoning method which introduces additional spatial relations such as *inside/cover*, *on/under*, and *above/below*. This improvement leads to more accurate action classification results and also object categorization based on their roles in the manipulation.
- The action perception framework also provides semantic segmentation and classification of trajectory information. This allows robots to perceive the temporal duration of individual action primitives and also to distinguish between different motion profiles (e.g. *periodic* or *discrete* segments) of the same action.
- We, for the first time, introduce the term *temporal scaling* which allows the robots to autonomously reason about how to generate an observed action movement in time constrained tasks. We finally evaluate the proposed method on a humanoid robot.

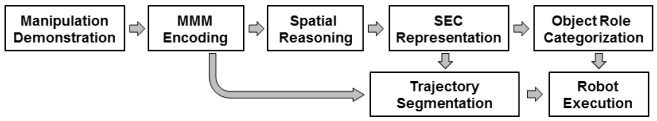

Fig. 1. Block diagram of the algorithm.

II. METHOD

In the following, we will provide a detailed description of our method, block diagram of which is shown in Fig. 1.

A. Manipulation Demonstration & MMM Encoding

The framework starts with the human demonstration of manipulation actions. In this work, we investigate eight different manipulation actions: *stir*, *pick&place*, *put in*, *take down*, *put on*, *drink*, *pour*, and *cut*. Fig. 2 shows sample frames from each manipulation type. All actions were demonstrated at least five times with various objects at short or long temporal scales. In Fig. 3, sample frames from three stirring demonstrations are depicted to highlight the degree of intra-class variation.

All demonstrations were recorded by a marker-based motion capture system working at 100 Hz. During demonstrations, we captured motions of the human subject and objects in the scene. Using manually created object models, $6D$ pose information were then estimated from the tracked object markers. Recorded demonstrations were finally encoded with the Master Motor Map (MMM) [18], which is a generic mapping framework allowing the transfer of whole-body human motions to different embodiments, e.g. humanoid robots. The entire motion data set, including MMM representations and object models, is publicly available as a part of the KIT whole-body human motion database [19].

B. Spatial Reasoning

This section elaborates on how to extract frame-wise spatial relationships between human body segments (e.g. the left hand) and objects (e.g. a bowl) in the scene. The aim here is to represent the observed scene as a set of symbolic predicates that describe the world state.

Given a manipulation recording, we first represent the scene as a set of point clouds, $\mathcal{P} = \{p_1, \cdots, p_n\}$, where n is the total number of tracked entities in the scene and p_i defines the point cloud of an object or a human body segment. Each object point cloud is generated from

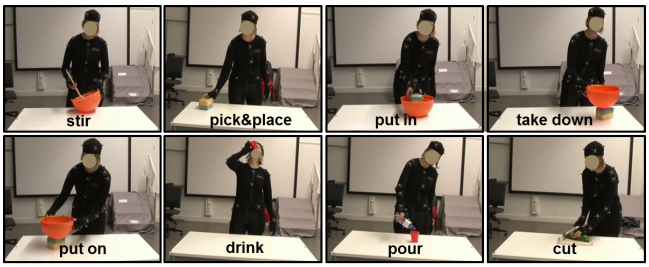

Fig. 2. A sample original frame for each manipulation type.

Fig. 3. Three different demonstrations of the stirring action.

a given object model. At each time instant, we compute an oriented bounding box around each point cloud in the world coordinate system to be further used to compute spatial relations between each cloud pair, i.e. $\mathcal{R}\ (p_i, p_j)$. Fig. 4 shows the point cloud representation of a sample scene together with the computed oriented bounding boxes of manipulated objects.

We introduce a rule-based system that defines eight basic spatial relations as follows:

Touch: indicates the direct contact relation between two neighboring clouds when the minimum distance d between the clouds is less than a predefined threshold τ; that is, $\mathcal{R}_{touch}\ (p_i, p_j) = 1$, if $d < \tau$.

Disjoint: represents spatially separated clouds and is the complement of the relation *touch*.

Inside: defines whether a cloud p_i is, at least partly, inside another cloud p_j. This predicate initially requires the relation $\mathcal{R}_{touch}$ to be set as a precondition. In this case, we check if any point in p_i falls into the oriented bounding box of p_j. To finally confirm the *inside* relation, we additionally compare the number of points in both clouds, for instance, $\mathcal{R}_{inside}\ (p_i, p_j)$ is set if $|p_i| < |p_j|$. In Fig. 4, the spoon (i.e. the red bounding box) is detected as being *inside* the bowl (i.e. the black bounding box). Note that they also fulfill the precondition, i.e. $\mathcal{R}\ (spoon, bowl) = 1$.

Cover: corresponds to the symmetrical counterpart of the relation *inside*. In the same example in Fig. 4, the bowl *covers* the spoon.

On: indicates that one point cloud is on top of the other. We first check whether $\mathcal{R}_{touch}$ and $\mathcal{R}_{inside}$ are detected as true and false, respectively. Next, we compute if any point in p_i lies in the horizontal projection, i.e. XY-space, of the oriented bounding box of p_j. We finally check both cloud positions along the Z-axis to confirm that the position of cloud p_i is higher than that of p_j. For instance, in Fig. 4, the bowl (i.e. the black bounding box) is detected as being *on* the table (i.e. the blue bounding box).

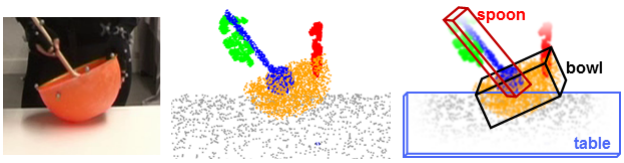

Fig. 4. Extraction of spatial relations from a sample scene. Left: Human demonstrator and manipulated objects together with attached markers. Middle: 3D point cloud representation. Each color represents a unique entity. Right: Sample oriented bounding boxes computed for manipulated objects.

Under: is considered as the symmetrical counterpart of the relation *on*. In Fig. 4, the table is *under* the bowl.

Above: defines whether a cloud p_i is above another disjoint cloud p_j. We here apply the same reasoning method employed to compute the relation *on* but with a different precondition. In this case, the predicate *above* requires the relation $\mathcal{R}_{disjoint}$ to be true as an initial condition. In Fig. 4, the spoon is estimated as *above* the table.

Below: corresponds to the symmetrical counterpart of the relation *above*.

Due to sake of simplicity, in the computation of those predicates, we introduce some hard constraints, such that human body segments (e.g. hands) cannot have relations other than *touch* or *disjoint* with any other object in the scene.

C. SEC Representation

Once the spatial relations between object pairs are computed at each time instant, we represent the scene by a graph, in which nodes refer to the tracked object or human body segment and edges indicate the spatial relation between two nodes. By employing an exact graph matching method, we discretize the continuous graph sequence into decisive main graphs, i.e. "key frames", each of which represents a topological change in the scene. All extracted main graphs form the core skeleton of the SEC, which is a matrix where rows are spatial relations (e.g. *touch* or *inside*) between object pairs and columns describe the scene configuration, i.e. a world state, when a new main graph occurs.

Fig. 5 depicts the SEC representation of a stirring demonstration with sample key frames. For instance, the second row of the SEC represents the spatial relations between the spoon and the bowl, i.e. $\mathcal{R}\ (spoon, bowl)$. Note that although there exist more object pairs, the SEC only encodes those pairs that produce at least one relational change, e.g. from *touch* to *disjoint*. All other pairwise relations are static and therefore irrelevant (e.g. between the left and right hands). Note that for the sake of simplicity, the symmetrical counterparts of pairwise relations (e.g. $\mathcal{R}\ (bowl, spoon)$) are omitted.

Details of the SEC concept were described in [1] which also provides a method to measure the semantic similarity between two event chains by comparing rows and columns of SECs using sub-string search and counting algorithms.

D. Object Role Categorization

After encoding the manipulation with the SEC matrix, we categorize SEC graph nodes, i.e. objects, according to their

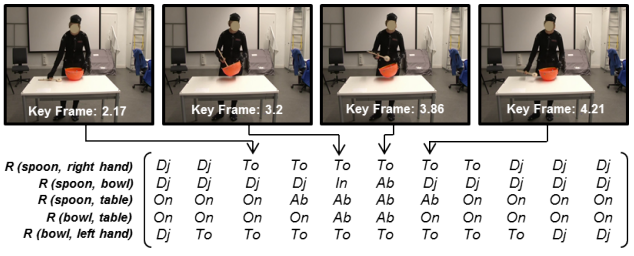

Fig. 5. The SEC representation of a sample stirring action. Detected spatial relations are *Touch (To)*, *Disjoint (Dj)*, *Inside (In)*, *Above (Ab)*, and *On*.

roles in the action. For this process, we employ the method in [20], which assumes each manipulation is composed of the three main elements: *manipulator*, *primary* and *secondary object*. The *manipulator* is the main actor, e.g. a hand, which plays the main role by frequently interacting with objects in the scene. The *primary object* is the tool that is actively used by the *manipulator*. All other objects directly interacting with the *primary object* are named *secondary objects*. We further extend these categories with the background set that represents the supporting surface, e.g. table.

In Fig. 5, SEC nodes for the right hand, spoon, bowl, and table are respectively categorized as *manipulator*, *primary object*, *secondary object*, and *background*. We store the four detected categories as *object quadruples* to be employed to smooth the SEC matrix. In this sense, we remove all SEC rows that do not involve any of those *object quadruples*.

E. Trajectory Segmentation

So far, we encoded descriptive spatiotemporal features of manipulations at the semantic level together with the role of manipulated objects. We will now enrich this semantic representation with a detailed trajectory information required for the robot execution. The entire trajectory processing approach is summarized in Fig. 6 on a sample stirring action and has the following steps:

1) Semantic Trajectory Segmentation: Each column of the SEC matrix highlights a unique world state and a transition from one column to the next indicates an individual manipulation primitive. For instance, the transition from the second to third SEC columns in Fig. 5 denotes the *approaching* primitive while the right hand is getting closer to the spoon, whereas the interval between the fifth and sixth columns describes the *stirring* movement primitive only. We, therefore, employ these decisive temporal points to capture the most crucial segment of the entire trajectory.

For this purpose, we apply a reasoning method which searches for the relational changes only between the *manipulator* and *primary object* or between the *primary* and *secondary objects*, if there exists any. For instance, a *pick&place* demonstration (see Fig. 2) does not involve any secondary object. Hence, we consider the trajectory segment only when the *manipulator* touches the *primary object*, i.e. the interval of $(Disjoint\ Touch\ Disjoint]$. In the case of, for instance, a stirring demonstration, we inspect the semantic segment while the *primary* and *secondary objects* are interacting, i.e. having an *inside* relation. Fig. 6 shows the MMM representation of the *primary object* motion, i.e. spoon. The gray box corresponds to the temporal border of the semantic trajectory segment in which the spoon is *inside* the bowl.

2) Trajectory Sub-segmentation: In this section, we continue with the sub-segmentation of the semantic segment extracted in the previous step. Note that, for the sake of simplicity, at this point we only consider semantic segments that represent the action descriptive primitives (e.g. *stirring* or *moving*), although all other primitives, such as *approaching*, *grasping*, or *withdrawing* are also correctly detected.

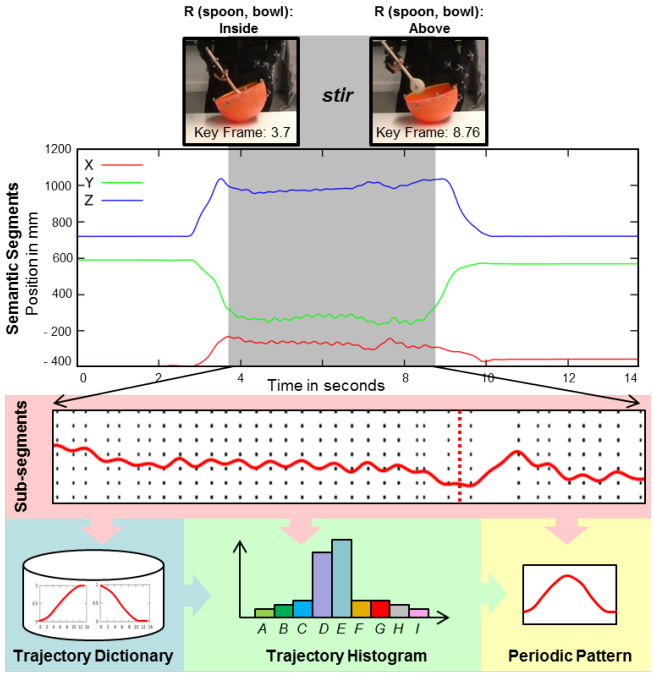

Fig. 6. Trajectory segmentation process.

Once the semantic segment of the low-passed (smoothed by a Gaussian filter) trajectory ($\mathcal{T}$) is explored, we further compute all local minima and maxima points (m_i) in each dimension separately. Those local extrema define geometrical variations in the trajectory pattern (e.g. curves, straight lines), which help to identify the main intention in the action. We consider each trajectory fragment between two consecutive extrema as one subsegment ($\mathcal{S}^i$), i.e. $\mathcal{S}^i = \mathcal{T}_{[m_i, m_{i-1}]}$.

Inspired from the work of [21], we also apply a persistence measure to remove noisy extrema. The persistence value of two consecutive minimum and maximum points corresponds to the amplitude differences (i.e. $|\mathcal{T}_{m_i}| - |\mathcal{T}_{m_{i-1}}|$) which is then compared with an empirically defined threshold to select characteristic points. The red box in Fig. 6 indicates the final subsegment borders of the semantic segment in the gray box. Black dashed lines here represent local extrema and the red dashed line indicates a single local maximum point removed due to the poor persistence measure.

3) Trajectory Dictionary Generation: After deriving the characteristic subsegments, we measure the motion similarities between subsegments ($\mathcal{S}^i$) in the spatiotemporal domain. Our main intent here is to create a dictionary which stores only unique trajectory subsegments.

To compare two subsegments, we employ Dynamic Time Warping (DTW) [22] with the L_2-norm. DTW computes the best alignment between two trajectory subsegments which may vary in time or speed. The DTW approach implemented using dynamic programming warps the time axis iteratively until finding the minimum distance (an optimal match) between the corresponding subsegments. The cost of best alignment between two subsegments $\mathcal{S}^1 = \langle s_1^1, \cdots, s_r^1 \rangle$ and $\mathcal{S}^2 = \langle s_1^2, \cdots, s_q^2 \rangle$ is recursively computed by:

$$\mathcal{D}(\mathcal{S}_i^1, \mathcal{S}_j^2) = \delta(s_i^1, s_j^2) + min \left\{ \begin{array}{l} \mathcal{D}(\mathcal{S}_{i-1}^1, \mathcal{S}_{j-1}^2) \\ \mathcal{D}(\mathcal{S}_{i-1}^1, \mathcal{S}_j^2) \\ \mathcal{D}(\mathcal{S}_i^1, \mathcal{S}_{j-1}^2) \end{array} \right\}, \quad (1)$$

where $\delta(s_i^1, s_j^2)$ is the Euclidean distance between samples s_i^1 and s_j^2. The final DTW distance between two subsegments is then given by $\mathcal{D}(\mathcal{S}_{|\mathcal{S}^1|}^1, \mathcal{S}_{|\mathcal{S}^2|}^2) = \mathcal{D}(\mathcal{S}_r^1, \mathcal{S}_q^2)$.

When the first observed motion trajectory is subsegmented, we store the first subsegment directly in the dictionary. The next subsegment is then compared with the one in the dictionary by employing the DTW method. The dictionary is updated with this new subsegment if the DTW distance is higher than a certain threshold. We continue this operation with all the other subsegments in each dimension parsed from all 70 motions in the dataset. Note that at this step the subsegment amplitudes are normalized.

4) Trajectory Histogram Representation: Next, we convert the detected semantic trajectory segment into a histogram representation. For this purpose, each trajectory subsegment is mapped to its closest counterpart in the dictionary. A histogram, showing the frequency of each dictionary element in the semantic trajectory segment, is then computed for each XYZ-dimension separately. We compute the final histogram ($\mathcal{H}$) by concatenating three individual histograms as $\mathcal{H} = [\mathcal{H}_x, \mathcal{H}_y, \mathcal{H}_z]$. The histogram $\mathcal{H}$ has the size of $1 \times 3k$, where k is the length of the dictionary.

5) Periodicity Measure: Trajectory histograms are used for two main purposes: First, we can compare trajectory (motion) profiles of different demonstrations since each histogram acts as a descriptive feature vector. For this purpose, we compute the Bhattacharyya distance ($d_\mathcal{B}$) which measures the statistical separability between two histograms as:

$$d_\mathcal{B} = -\ln \sum_{i=1}^{k} \sqrt{\mathcal{H}_1(i) \times \mathcal{H}_2(i)} \quad . \quad (2)$$

Second, we employ the histogram representation to investigate whether the trajectory segment follows a periodic pattern or not. This is a very important contribution of our framework, which is required for autonomous generation of periodic motions in different temporal lengths. In order to measure the periodicity, we search for which dictionary elements are frequently observed in the histogram. For this purpose, we assign a label to each dictionary element, which converts the trajectory segment into a string stream. We finally apply standard substring search methods to explore the most repetitive string part.

For the example given in Fig. 6, the trajectory subsegments in the red box are represented as a stream of "$\cdots DEDEDEDEDEDDCDDEDED \cdots$", where D and E are unique dictionary elements. The repeating cycle is then estimated as "DE" which corresponds to the two long histogram bars in the green box in Fig. 6, i.e. the curve shape shown in the yellow box in Fig. 6. The final periodicity value ρ is measured as $\rho = \frac{\eta|\gamma|}{|\chi|}$, where η is the number of repetitions, $|\gamma|$ and $|\chi|$ are the lengths of the repetitive string and the entire stream, respectively. The periodicity value was finally measured as 0.73 for the semantic trajectory segment shown in the gray box in Fig. 6.

F. Robot Execution

The periodicity measure is an important feature that allows robots to autonomously explore whether an action segment, i.e. motion primitive, can be executed at different temporal scales without altering any characteristic feature, e.g. speed. If the periodicity measure is high, the robot can simply repeat the detected periodic subsegment to imitate actions at longer or shorter temporal lengths in a given time constrained task. In the case of having non-periodic actions, the motion can be executed, for instance, by altering the velocity component to meet the temporal constraints.

In this respect, we use Dynamic Movement Primitives (DMPs) [5] in order to generate segmented trajectories. A big benefit of using DMPs is that several features including speed, goals and start positions can be changed by adjusting the parameters to make the planner task-specific.

III. RESULTS

A. Semantic Manipulation Similarities

We applied the proposed enriched semantic action perception framework to all 70 demonstrations in our new manipulation action dataset introduced in section II-A.

As described in section II-C, we first encoded all manipulations by enriched event chains. We then measured the semantic similarities between 70 SEC matrices by employing the method in [1]. Fig. 7 (left) shows the class-wise average SEC similarities. The high diagonal similarity values indicate that our SEC-based action representation approach can successfully capture the semantic similarities even though demonstrations have high intra-class variations (see Fig. 3).

In Fig. 7 (left), we have almost 63% similarity between the actions *put in* and *put on*, which is very reasonable since only the last spatial relation is different (i.e. *inside* versus *on*). We also have 64% semantic similarity between the actions *stir* and *pour*. This is because in both actions almost all primitives are the same, except the one which introduces an additional relation *inside* only in the action *stir*. This

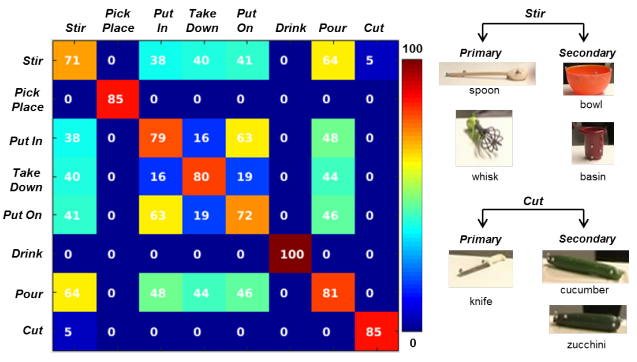

Fig. 7. Left: Computed average SEC similarities for each action type. Right: Estimated sample object categories for actions *stir* and *cut*.

TABLE I

CLASS-WISE F-SCORES: COMPARISON OF THE ORIGINAL SEC METHOD [1] WITH THE NEW ENRICHED SEC (*eSEC*) REPRESENTATION.

	Stir	*Pick Place*	*Put In*	*Take Down*	*Put On*	*Drink*	*Pour*	*Cut*
SEC [1]	0.42	0.95	0.54	0.0	0.0	0.0	0.0	0.0
eSEC	**0.89**	**1.0**	**0.85**	**1.0**	**0.75**	**1.0**	**0.93**	**1.0**

is a very important finding showing that both actions *stir* and *pour* are, to some degree, similar at the semantic level. Once we, however, include the action dynamic, for instance the trajectory profiles, the similarity will dramatically drop since the action *stir* is periodic whereas the motion *pour* is rather discrete as detected in Table II. We here emphasize that observing such a slightly high similarity between different action types is not a limitation. This is rather a feature of our semantic perception framework since we, at this very high symbolic level, consider only the interaction between objects in the scene. At the next level (see section III-B) relevant trajectory information will be additionally incorporated for the sake of accuracy.

As explained in section II-D, the framework can also categorize manipulated objects considering their roles in the manipulation task. Fig. 7 (right) shows estimated *primary* and *secondary objects* in actions *stir* and *cut*. For instance, the *spoon* and *whisk* were estimated as the *primary* tools for *stirring* ingredients inside the *bowl* or *basin*. Such a reasoning solely based on object roles plays a vital role for humanoid robots to explore object affordances and even to replace missing objects with the most appropriate one as proposed in [23].

In order to show the main contribution of our new spatial reasoning method explained in section II-B, we classified demonstrated actions based on their semantic similarities and compared with the one computed from the original SEC method [1] which involves only three types of spatial relations: *Touch*, *Disjoint*, and *Absence*. For this purpose, we first computed semantic similarities between all 70 samples and then applied a simple one-nearest-neighbor classifier. In order to compute the classification accuracy, we measured the class-wise F-scores. Table I shows the major improvement of our new enriched SEC method in contrast to the original SEC concept [1]. This result suggests that the more descriptive and structural the semantic representation, the greater the perception capabilities of the robot.

B. Semantic Trajectory Similarities

In this section, we continue with boosting the semantic event chains with descriptive trajectory information. In this sense, we applied our proposed trajectory sub-segmentation method (section II-E) to the entire 70 actions in the dataset.

As clarified in section II-E.1, the framework first derived the crucial semantic trajectory segments by only considering the spatiotemporal interactions between the estimated *manipulator*, *primary* and *secondary objects* in an unsupervised manner. Those explored semantic segments were further subsegmented to form a trajectory dictionary. Fig. 8 shows

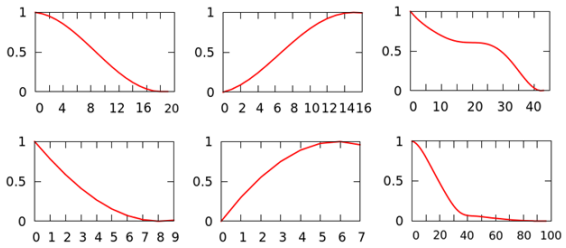

Fig. 8. Sample normalized subsegments from the trajectory dictionary.

some sample subsegments from the generated trajectory dictionary which stores in total 20 unique subsegments after processing all 70 demonstrations. As the figure indicates, there is a significant difference between each subsegment due to the success of the DTW method.

Next, we measured the pairwise Bhattacharyya distances between the computed 70 trajectory histograms. Fig. 9 (left) depicts the average Bhattacharyya distances between different demonstrations of each action type. Furthermore, we computed the DTW distance between the raw semantic trajectory segments without applying the sub-segmentation step. Fig. 9 (right) shows the average class-wise DTW distances. The first impression that Fig. 9 (left) conveys is that our proposed histogram-based trajectory representation method has a more homogeneous distribution, such that low distance values appear on the main diagonal whereas much higher distances emerge across different action types. This finding suggests that the proposed subsegment histograms can directly be employed to reveal the similarity between the two trajectory patterns. As Fig. 9 (right) depicts, such a comparison is, however, not feasible if we consider the DTW distance between raw semantic trajectory segments. This is a very crucial contribution in order to use the memory in a more efficient way. It is because robots can now autonomously detect whether the currently observed trajectory pattern is different than those of previously obtained demonstrations of the same action type and, hence, store only highly different and unique motions.

For instance, in Fig. 9 (left) stirring demonstrations have less intra-class variations, hence less distance due to observing similar movements. This is, however, not the case for the *pick&place* action type since followed motions are not goal directed and vastly vary from demonstration to

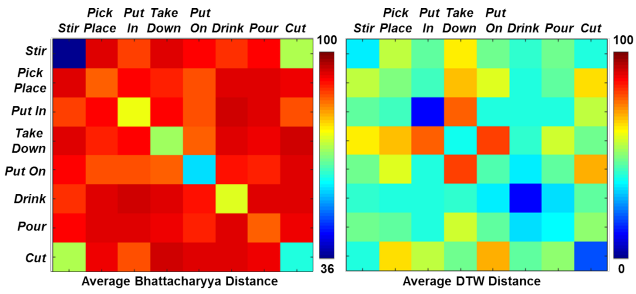

Fig. 9. Comparison of class-wise trajectory distances.

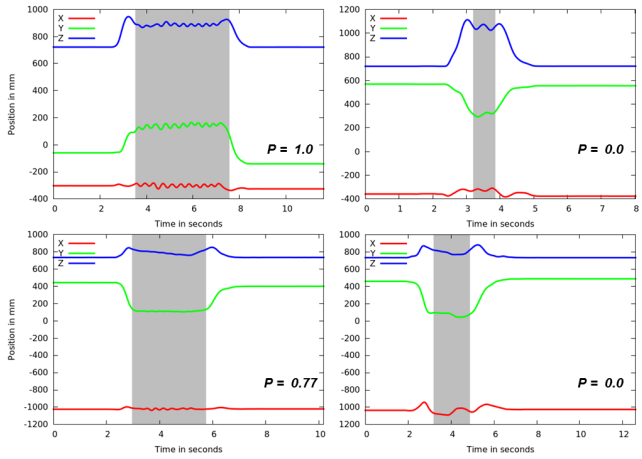

Fig. 10. Different demonstrations for actions *stir* (Top) and *cut* (Bottom). In contrast to samples on the left, motions on the right last for shorter period of time, hence the periodicity measure (P) is lower.

demonstration. Note that we also obtained a relatively small distance between *stir* and *cut* demonstrations as both have a similar periodic motion as shown in Fig. 10.

As claimed in section II-E.5, the proposed trajectory sub-segmentation method also allows us to measure the trajectory periodicity. Table II shows the average periodicity values for each action type. As expected we obtained quite high periodicities only for the actions *stir* and *cut*. The reason of having slightly low values, for instance around 0.5 out of 1.0 for the action *cut*, is that some demonstrations last for a very short period of time, hence, the length of the repeating cycle is not enough to detect the periodicity. Fig. 10 depicts sample trajectory profiles for the *stir* and *cut* demonstrations. Both motions on the right have less lifespans in contrast to those on the left, therefore, no periodicity was estimated.

C. Robot Execution

In this section, we describe how the enriched semantic action information can be further employed by the robot to autonomously execute actions at different temporal scales.

The execution phase was evaluated on the humanoid robot ARMAR-IIIb [24] which has 43 DOFs and is equipped with position, velocity, and force-torque sensors. In all experiments, we assumed that the high-level action plan was provided in advance and objects were already grasped before the execution since the grasping and action planning are not in the core of this work.

In the first experiment, the robot was asked to imitate one of the observed *stirring* actions at different temporal lengths. Fig. 11 (a) shows a sample frame from a human demonstration. In Fig. 11 (b) the right hand trajectory encoded in the MMM format is depicted. Vertical dashed lines indicate the estimated semantic segment borders while the spoon is *inside* the bowl. This segment takes approximately 5 seconds and represent the actual stirring movement. The gray box shows the detected periodic pattern which is highlighted in Fig. 11 (c). In the execution phase, the robot ARMAR-IIIb chooses the whisk instead of the spoon (see Fig. 11 (d)) since in the stirring demonstrations both objects shared the same role as explored in Fig. 7 (b). Finally, the robot selects a periodic DMP and repeats only the detected periodic pattern at the same speed until meeting the given temporal length, which is 190 seconds as shown in Fig. 11 (e). In the case of having a stirring plan with a lower speed, the *stirring* action is generated with less repeating cycles to meet the temporal constraint which is 120 seconds as depicted in Fig. 11 (f).

TABLE II

CLASS-WISE AVERAGE PERIODICITY MEASURES.

Stir	*Pick Place*	*Put In*	*Take Down*	*Put On*	*Drink*	*Pour*	*Cut*
0.67	0.0	0.0	0.0	0.0	0.0	0.0	**0.5**

The next experiment covers the robot execution of the *pick&place* task. The top row of Fig. 12 shows the semantic segments of the human right hand motion. Vertical dashed lines show the border of the segment *replacing* which starts when the hand touches the bowl and ends when the bowl is released after 1.88 seconds. Since this segment is not detected as periodic, ARMAR-IIIb selects a discrete DMP and slows down to generate the same *replacing* motion at a longer temporal scale, e.g. 18.8 seconds as desired by the planner. Note that the robot again applies the object replacement. In this case, the sponge is used instead of the bowl since both performed the same role in the *pick&place* scenarios demonstrated by human subjects.

These results indicate that the robot ARMAR-IIIb can autonomously segment the demonstrated action and select the most appropriate DMP type for the execution. ARMAR-IIIb can further adjust the required repeating cycles to meet temporal constraints if the action is periodic. Otherwise, the robot selects altering other characteristic features, such as the action speed. Consequently, the robot can autonomously decide how to execute actions at different temporal scales. See the supplementary movies showing the entire robot execution of different *stir* and *pick&place* actions.

IV. CONCLUSIONS

In this work, we addressed the problem of temporal segmentation and execution of human demonstrated manipulation actions by humanoid robots. Unlike the conventional appearance- or motion-based approaches, our proposed framework relies on the action semantics and allows robots to autonomously classify the derived motion primitives as periodic or discrete movements. We evaluated our framework on a new large manipulation action dataset. Our findings derived from conducted experiments on the robot ARMAR-IIIb suggest that it is possible to autonomously execute the perceived action segments at different temporal scales.

REFERENCES

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

[2] M. Sridhar, G. A. Cohn, and D. Hogg, "Learning functional object-categories from a relational spatio-temporal representation," in *Euro. Conf. on Artificial Intelligence*, 2008.

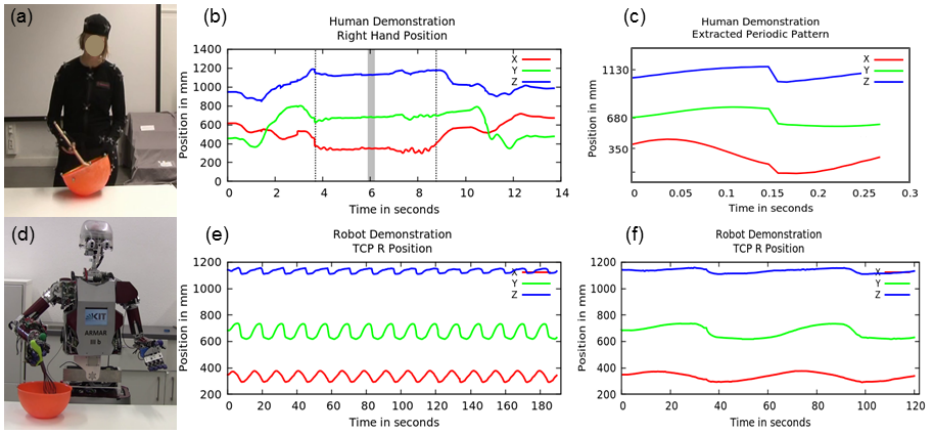

Fig. 11. Robot execution of the action *stir*. (a) Human demonstration. (b) Right hand position. Vertical dashed lines indicate the semantic segment border. The gray box show the periodic region. (c) Extracted periodic pattern. (d) Robot execution. (e) Robot TCP R position for the robot execution lasting 190 seconds. (f) Robot executes the same action in 120 seconds but at a lower speed.

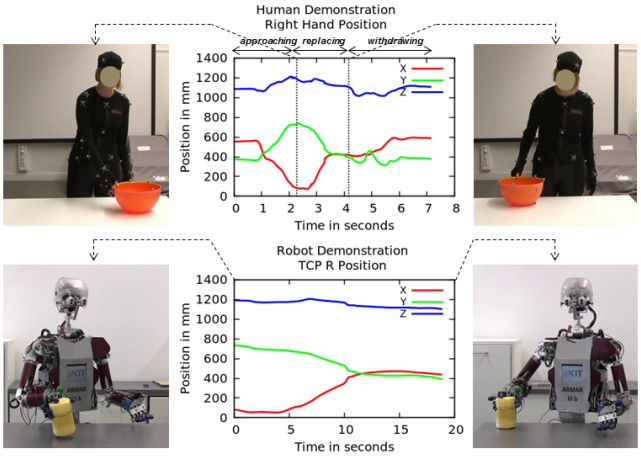

Fig. 12. Robot execution of the action *pick&place*. Top and bottom rows show the human and robot executions, respectively. Vertical dashed lines in the first row indicate the semantic segment border, which is executed by the robot at a longer temporal scale.

[3] 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 *IEEE-RAS International Conference Humanoid Robots*, October 2013.

[4] M. Pei, Z. Si, B. Yao, and S.-C. Zhu, "Video event parsing and learning with goal and intent prediction," *Computer Vision and Image Understanding*, vol. 117, no. 10, pp. 1369–1383, 2013.

[5] J. A. Ijspeert, J. Nakanishi, and S. Schaal, "Movement imitation with nonlinear dynamical systems in humanoid robots," in *Proc. 2002 IEEE Int. Conf. Robotics and Automation*, 2002, pp. 1398–1403.

[6] A. Ude, "Trajectory generation from noisy positions of object features for teaching robot paths," *Robotics and Autonomous Systems*, vol. 11, no. 2, pp. 113 – 127, 1993.

[7] D. Lee and Y. Nakamura, "Stochastic model of imitating a new observed motion based on the acquired motion primitives," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2006, pp. 4994 –5000.

[8] S. Calinon, F. Guenter, and A. Billard, "On learning, representing, and generalizing a task in a humanoid robot," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 37, no. 2, pp. 286 –298, 2007.

[9] K. Zampogiannis, Y. Yang, C. Fermüller, and Y. Aloimonos, "Learning the spatial semantics of manipulation actions through preposition grounding," in *IEEE International Conference on Robotics and Automation, ICRA*, 2015, pp. 1389–1396.

[10] S. Panda, A. H. A. Hafez, and C. V. Jawahar, "Learning support order for manipulation in clutter," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 809–815.

[11] W. Liang, Y. Zhao, Y. Zhu, and S.-C. Zhu, "What is where: Inferring containment relations from videos," in *25th International Joint Conference on Artificial Intelligence (IJCAI)*, 2016.

[12] T. Kulvicius, K. J. Ning, M. Tamosiunaite, and F. Wörgötter, "Joining movement sequences: Modified dynamic movement primitives for robotics applications exemplified on handwriting," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 145–157, 2011.

[13] T. Luksch, M. Gienger, M. Mühlig, and T. Yoshiike, "A dynamical systems approach to adaptive sequencing of movement primitives," in *Proceedings for the conference of ROBOTIK*, 2012.

[14] F. Meier, E. Theodorou, F. Stulp, and S. Schaal, "Movement segmentation using a primitive library," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011.

[15] D. Kulic and Y. Nakamura, "Scaffolding on-line segmentation of full body human motion patterns," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 2860–2866.

[16] A. Fod, M. J. Matarić, and O. C. Jenkins, "Automated derivation of primitives for movement classification," *Autonomous Robots*, vol. 12, no. 1, pp. 39–54, 2002.

[17] J. Barbič, A. Safonova, J.-Y. Pan, C. Faloutsos, J. K. Hodgins, and N. S. Pollard, "Segmenting motion capture data into distinct behaviors," in *Proceedings of Graphics Interface*, 2004, pp. 185–194.

[18] Ö. Terlemez, S. Ulbrich, C. Mandery, M. Do, N. Vahrenkamp, and T. Asfour, "Master motor map (MMM) - framework and toolkit for capturing, representing, and reproducing human motion on humanoid robots," in *Int. Conf. on Humanoid Robots*, 2014, pp. 894–901.

[19] C. Mandery, O. Terlemez, M. Do, N. Vahrenkamp, and T. Asfour, "Unifying representations and large-scale whole-body motion databases for studying human motion," *IEEE Transactions on Robotics*, vol. 32, no. 4, pp. 796–809, 2016.

[20] E. E. Aksoy, M. J. Aein, M. Tamosiunaite, and F. Wörgötter, "Semantic parsing of human manipulation activities using on-line learned models for robot imitation," in *Proc. of IROS*, 2015, pp. 2875–2882.

[21] F. Despinoy, D. Bouget, G. Forestier, C. Penet, N. Zemiti, P. Poignet, and P. Jannin, "Unsupervised trajectory segmentation for surgical gesture recognition in robotic training," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 6, pp. 1280–1291, 2016.

[22] E. Keogh and A. C. Ratanamahatana, "Exact indexing of dynamic time warping," *Knowledge and Information Systems*, vol. 7, no. 3, pp. 358–386, 2005.

[23] A. Agostini, M. J. Aein, S. Szedmak, E. E. Aksoy, J. Piater, and F. Wörgötter, "Using structural bootstrapping for object substitution in robotic executions of human-like manipulation tasks," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2015, pp. 6479–6486.

[24] T. Asfour, K. Regenstein, P. Azad, J. Schroder, A. Bierbaum, N. Vahrenkamp, and R. Dillmann, "Armar-iii: An integrated humanoid platform for sensory-motor control," in *IEEE-RAS International Conference on Humanoid Robots*, 2006, pp. 169–175.