Language for Learning Complex Human-Object Interactions

Mitesh Patel[1], Carl Henrik Ek[2], Nikolaos Kyriazis[3], Antonis Argyros[3], Jaime Valls Miro[1] and Danica Kragic[2]

***Abstract*— In this paper we use a Hierarchical Hidden Markov Model (HHMM) to represent and learn complex activities/task performed by humans/robots in everyday life. Action primitives are used as a grammar to represent complex human behaviour and learn the interactions and behaviour of human/robots with different objects. The main contribution is the use of a probabilistic model capable of representing behaviours at multiple levels of abstraction to support the proposed hypothesis. The hierarchical nature of the model allows decomposition of the complex task into simple action primitives. The framework is evaluated with data collected for tasks of everyday importance performed by a human user.**

I. INTRODUCTION & MOTIVATION

For a robot, learning of everyday human activities is a challenging problem. Imitation learning has gained much attention in the last decade and attracted considerable research [1]. Despite this, we are still a long way from having a robot working alongside humans and demonstrating the same competencies. This is due to the fact that human behaviours are inherently highly complicated and the limitation of various sensors to capture such complex behaviours. It therefore remains an open challenge how to model behaviour from sensor data.

In the area of grasping and manipulation of everyday objects there has been a growing interest in expressing tasks as a combination of meaningful subparts called *Action Primitives* (APs) [2]. Research done on human motion and other biological movements postulates that movement behaviour consists of simple *APs:* atomic movements that can be combined and sequenced to form complex behaviours [3], [1], [4]. For example, as shown in Fig. 1 the task of *pouring water from a mug* could be decomposed into the sequence of APs such as `approach-grasp-lift-tilt-untilt-place back-release object-retreat (the arm to its initial position)`, where the AP cannot be decomposed further. Arguments raised in the field of neuroscience [5] reinforces that human actions are composed of APs similar to human speech where utterances of words are broken down into phonemes. Hence the use of a grammar based on APs is an attractive approach to represent tasks performed by a human. The use of APs allows for a "symbolic" description of complex actions. This is in accordance with the idea that a human task recognition process may be considered as an understanding of sequential human behaviours which, in its turn, consists of interpreting a sequence of action primitives [6]. Along with the advantage of a top-down approach (complex tasks decomposed into APs), this also enables bottom-up approach whereby APs can be shared to construct different task sequences.

A challenging part of detecting and recognising grasping and manipulation related tasks is the representation of the noisy sensory data. For a robot to learn these tasks, it is important that the task sequence presented to the robot is complete, with minimal loss of information. In real scenarios, sensor limitation and other environmental factors, makes this very challenging. Given the inherent level of uncertainty in the sensors, it is difficult to model these tasks in a deterministic manner. Stochastic or probabilistic models are the techniques of choice that researchers have explored to represent the possible uncertainties involved.

Our main contribution in this paper is to exploit a temporal probabilistic model, *Hierarchical Hidden Markov Model* (HHMM) capable of representing and learning grasp and manipulation related complex human activities. The model builds upon alphabets of APs which can be combined in different order to compose and describe complex human tasks. The hierarchical nature of the framework allows the decomposition of a typical task into different APs which are learned by the model at different levels of the hierarchy. An example shown in Fig. 1 is a decomposition of a pouring task into sequence of APs. The APs provide the necessary tool to describe a task as a sequential combination similar to the natural language description. The proposed framework is capable of learning this grammar at different levels i.e. the action primitives are learned and inferred by observing the hand-object interaction and their motion in the cartesian space whereas the abstract level tasks are inferred by learning the sequence of APs.

II. RELATED WORK

Learning by imitation is one of the many approaches that have been used by roboticists to represent human motion. Khansari-Zadeh and Billard [7] used a learning method called *Stable Estimator of Dynamical Systems (SEDS)*, to learn the parameters of a time invariant dynamical system

[1]M. Patel and J. V. Miro are with Faculty of Engineering and IT, University of Technology Sydney (UTS), NSW 2007, Australia. (`mitesh.k.patel@student.uts.edu.au, Jaime.VallsMiro@uts.edu.au`)

[2]C. H. Ek and D. Kragic are with KTH - Royal Institute of Technology, Stockholm, Sweden, as members of the Computer Vision & Active Perception Lab., Centre for Autonomous Systems, www: http://www.csc.kth.se/cvap. (`(chek, dani)@kth.se`)

[3]N. Kyriazis and A. Argyros are with Institute of Computer Science, FORTH, Crete and Department of Computer Science, University of Crete, Crete, Greece. (`(kyriazis, argyros)@ics.forth.gr`)

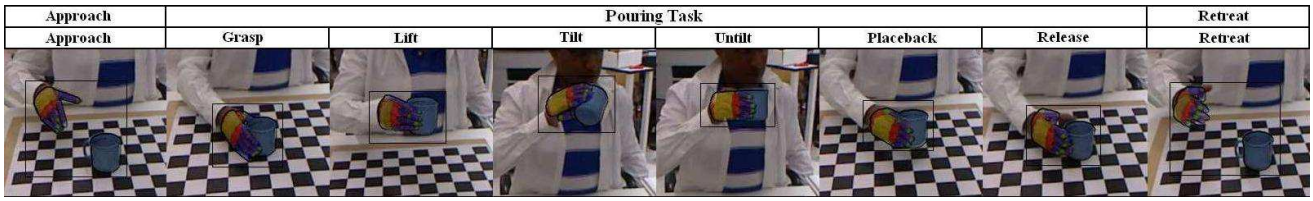

Fig. 1: Task of *Pouring* water from mug subdivided into APs. Each image depicts the output of hand-object tracking algorithm

to ensure that all motions closely follow the demonstrations while ultimately reaching and stopping at the target. The tasks learned by the SEDS were only of simple type such as moving an object from point-to-point. Dindo and Schillaci [8] proposed a *Growing Hierarchical Dynamic Bayesian Network (GHDBN)* to recognise the skills being observed and to reproduce them by exploiting the generative power of the model. The model learned and reproduced three actions i.e. *Dislocate, Approach* and *Hit*. Pastor *et. al.* [9] used a Dynamic Movement Primitive (DMP) framework in which the recorded movement were represented using non-linear differential equations. The movement library consisted of actions such as *grasping, placing* and *releasing*. Nemec and Ude [10] in their recent work also used a DMP based system to represent primitive movements. The DMP library used in their experiment consisted of tasks like *reaching, pouring, wiping, shaking, cutting, power grasps etc.*

In our previous work,we proposed a *Parametric Hidden Markov Model (PHMM)* to represent various action primitives [2]. The framework was trained in an unsupervised manner and represented and synthesized movement trajectories as a function of their desired effect on the object. The set of actions learned were *approach, grasp, push forward, push side, move side, rotate* and *remove*. Our previous work also exploited the dependencies of the hand and object features to generate the structure of a Bayesian Network (BN) [11], [12]. The evolved structure is used to predict the task of a user based on the type and object. However, the prediction of these tasks are done based on grasp instances and not on the entire trajectory of motion followed by the arm to perform a given task.

Related to the theoretical framework used in this work, HHMM has been applied to several different application areas. Nguyen *et. al.* [13] used a HHMM framework to model and recognise complex human activities. The model exploited both the natural hierarchical decomposition and shared semantics embedded in the movement trajectories. The tasks inferred were based on location semantics. In the area of ubiquitous computing, Liao [14] used an HHMM framework to infer user's mode of transportation, destination location and predict both short and long term movements. The framework was also able to infer if the user was deviating from his normal activities as an indication to provide guidance cues. With our work related to assistive robotic walker [15], we deployed a HHMM framework to infer the non-navigational and navigational intentions of the user. The hierarchical nature of the framework allowed integration of the tasks required for learning activities of daily living.

It is important to note that most of the actions learned or synthesized in [7], [8], [9], [2] are limited to basic APs such as *dislocate, hit, approach, grasp, push, rotate or move from point-to-point*. In this work we propose to use a probabilistic framework capable of representing an entire task by decomposing it into clusters of APs. Our approach is unique due to two main reasons, firstly we cluster the entire task sequence into pool of different APs and secondly, we use a unified probabilistic framework that exploits spatial relationship to learn APs and time dependent relationship between APs to predict the high level abstract task. The hierarchical nature of the model proves to be a strong tool for both learn and synthesizing tasks and APs at both levels.

III. HIERARCHICAL HIDDEN MARKOV MODEL (HHMM)

Probabilistic models have been successfully used by the AI community in particular in order to represent complex systems with prominent uncertainty [16]. Models such as Hidden Markov Model (HMM), Dynamic Bayesian Network (DBN) and HHMM are popular techniques used for human motion modelling and a wide variety of other applications. The endless list includes assistive robotics [15], sign language and gesture modelling [17], robot assisted surgery [18] and many more. These models have found its applicability in the field of robotics due to its ability to handle data noise and capture both the spatial and temporal variability in the movement and the change in variance along the movement. As we are dealing with noisy data from real scenarios, the model gives us the flexibility to exploit the temporal and spatial dependencies between different APs and tasks at different levels.

The HHMM framework used in our work is capable of structuring stochastic processes at multiple levels. The HHMM is an extension of HMM that is designed to model domains with hierarchical structure including such with dependencies at multiple length/time scales [19]. In an HHMM, the states of the stochastic automaton can emit single observations or strings of observations. Those that emit single observations are called "production states", and those that emit strings are termed "abstract states" [20]. The strings emitted by abstract states are themselves governed by sub-HMMs, which can be called recursively. When the sub-HMM is finished, control is returned to wherever it was called from [20]. The hierarchical nature allows decomposition of the problem at different levels of abstraction thereby facilitating exploration (long term planning/tasks) and exploitation (short term planning/APs) within the same framework.

In the paradigm of learning long term task/activities from APs, the high-level activities call the more refined low-

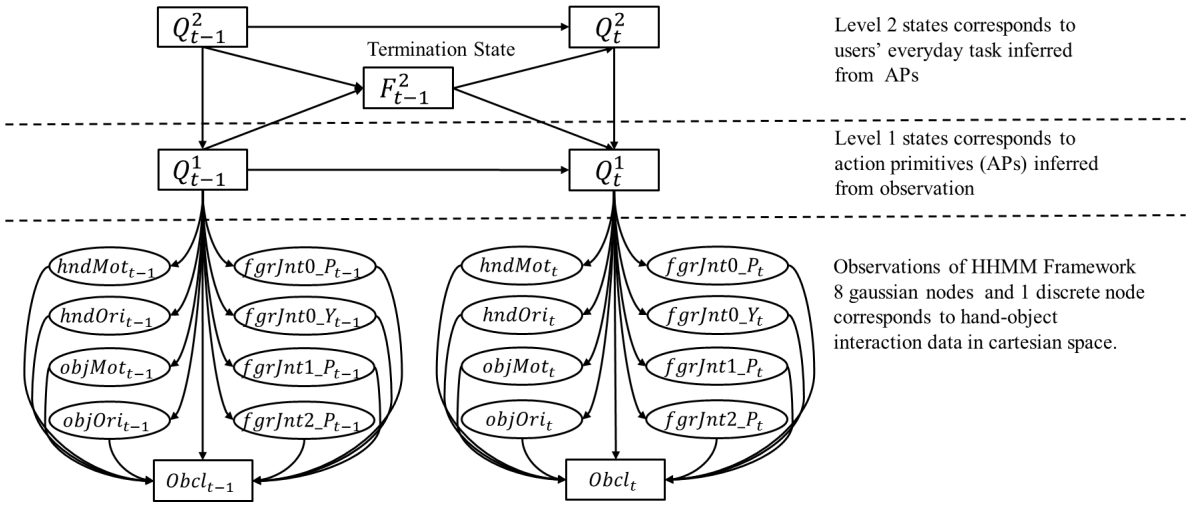

Fig. 2: HHMM Model used to infer action primitives and long term user task using different hand and object features. The latent variables of the GMM models used at the observation level are not shown here for simplicity. Refer to Table I for details of each observation node

level activities according to some distribution. A low-level activity will in turn call another lower-level activity, and this process continues until the most primitive possible activity is performed. When the lower level activity terminates - in some state - the parent behaviour may also terminate as long as the current state is in the set of destination states of the parent node.

A. PROBLEM SPECIFIC HHMM FRAMEWORK

The HHMM framework used to test our proposition is shown in Fig. 2. User state/tasks are inferred at the top level whereas the intermediate level represents the APs (shown in Fig. 2). In everyday life a single object can be used to perform many tasks (e.g. Mug can be used for drinking, pouring or handing it over to another person), hence it is difficult to predict the user task when he/she is approaching to grasp the object, but becomes more apparent after the object has been grasped. Similarly, after accomplishing the desired task, the action of retreating the hand after releasing the object cannot be described as part of the task sequence. Hence such action primitives (e.g. approaching to grasp an object (**APPRH**), and retreating after the object is released (**RETRT**) are not defined as a part of any long term task listed in Table II, but are described as APs independent of any task. In our framework, such independent APs are inferred at both levels of hierarchy. To better illustrate this concept, consider the example in Fig. 1. The user first approaches to grasp the mug, which has the same AP defined at both the levels as the specific task cannot be inferred without the object being grasped. Once the object is grasped, the user task can be inferred based on the type of grasp and the object, hence the HHMM model will infer the task at the higher level (2) and the action primitives at the lower level (1). After releasing the object the AP of retreating being independent from any task sequence will hence be inferred at both levels.

At the observation level, features extracted using a hand-object tracking (details given in Section IV) algorithm is used

TABLE I: Hand & object features used by the HHMM framework

Feature	Dim.	Description
hndMot	3	Hand motion in Cartesian space
hndOri	4	Hand orientation (quaternion)
fgrJnt0_P	1	Pitch of knuckle joint for index, ring & middle finger
fgrJnt0_Y	1	Yaw of knuckle joint for index, ring & middle finger
fgrJnt1_P	1	Pitch of first finger joint for index, ring & middle finger
fgrJnt2_P	1	Pitch of second finger joint for index, ring & middle finger
objMot	3	Object motion in Cartesian space
objOri	4	Object orientation (quaternion)
Obcl	6	Object class

which represents the interaction between the hand and object and its movement in cartesian space. Data features used in this experiment (listed in Table I) consists of 3D motion of hand and object, rotation of hand and object. The data feature also included the rotation movement of each finger joint. The trajectory of hand and object provided information regarding the motion of hand and object whereas the rotational motion (yaw, pan, tilt) provided the corresponding orientation information in the space. The movement of each finger joint provided details regarding the grasping of objects. All these data features were utilised to predict the APs at the lower level of the HHMM model.

B. Representation

A HHMM framework can be represented as a Hierarchical Dynamic Bayesian Network (H-DBN) as shown in Fig. 2. Its structure comprises of three types of nodes, Q^d_t, O_t, F^d_t where d is the depth of the hierarchy ($d = 2$ in our case). Edges between nodes represent their dependencies on each other. The detail of each node is specified as follows:

- Q^d_t represents the state of the system at time t and level d. Note that at any given time the system will be probabilistically represented by the state belief at all levels, and so will be the user goal state at the top level.
- Observations nodes O_t provide a probability of evidence as a function of a hiden state. In this work these are modelled as Gaussian Mixture Models (GMMs)

(represented by (μ,Σ)) or discrete nodes, $P(Q_t^d|O_t)$ node. As in [11], the nodes with GMM distribution are modelled by a discrete latent parent to store the mixture coefficients.

- F_t^d is the terminating state which specifies the natural completion of a sub-HMM and return the control back to the higher level/parent states.

Given the parameters (Q_t^d, O_t, F_t^d), the H-DBN defines the joint distribution over the set of variables that represents the evolution of the stochastic process over time. These distributions are in the form of prior distributions (initial probabilities), the transition probabilities and the observation probability distribution. The prior distribution and the transition probabilities are defined at every level (d).

C. Prior Model

The prior provides the initial probabilities of the most likely initial state of the user. The initial probabilities at both the levels are defined by

$$\begin{aligned} P(Q_1^2) &= \pi^2(j) \\ P(Q_1^1) &= \pi_k^1(j) \end{aligned} \tag{1}$$

where π^2 represent the initial probabilities at level 2 and π_k^1 represents the same at level 1, given the state at level 2 is k.

D. Transition Model

Each node in the HHMM represents a conditional probability distribution (CPD) or table (CPT). The state of the highest level (level 2 in Fig 2) at time t, depends upon the previous state at the same level and the termination flag at time $t-1$. Probabilities at the highest level are defined by

$$P(Q_t^2 = j|Q_{t-1}^2 = i, F_{t-1}^2 = f) = \begin{cases} A^2(i,j) & \text{if } F_{t-1}^2 = 0 \\ \pi^2(j) & \text{if } F_{t-1}^2 = 1 \end{cases} \tag{2}$$

Similarly, the states at the intermediate level (level 1 in Fig. 2) at time t, depends upon the previous state at the same level and the termination flag at time step $t-1$ and the state at the higher level in the same time step t, the probabilities of which are defined in (3).

$$P(Q_t^1 = j|Q_{t-1}^1 = i, F_{t-1}^2 = f, Q_t^2 = k) = \begin{cases} A_k^1(i,j) & \text{if } F_{t-1}^2 = 0 \\ \pi_k^1(j) & \text{if } F_{t-1}^2 = 1 \end{cases} \tag{3}$$

In (2), A^2 represents the transition probabilities from state i to j at level 2 whereas in (3), A_k^1 corresponds to transition probabilities at level 1 given the state at level 2 is k.

E. Termination Model

The termination state F at time t depends upon the level 2 state and level 1 state in the same time step t. The distribution of the termination state is defined by (4).

$$P(F_t^2 = 1|Q_t^2 = k, Q_t^1 = i) = A_k^2(i, end) \tag{4}$$

F. Observation Model

The observation model signifies the probability of seeing a specific observation conditioned on a discrete hidden state. For our application, observations are modelled as both Gaussian and discrete. The CPDs for Gaussian and discrete nodes is given by

Fig. 3: Objects used to perform manipulation tasks

$$\begin{aligned} P(O_t|Q_t^1 = i) &= N(\mu_i, \Sigma_i) \\ P(O_t|Q_t^1 = i) &= C(i) \end{aligned} \tag{5}$$

G. Learning and Inference

Expectation Maximisation (EM) and its variants are popular statistical technique used for learning. We use a semi-supervised mode of learning where the observation model to infer APs is learned in a supervised manner whereas the high level abstract states are learned without supervision. We use EM to learn the model and maximum likelihood estimator for predicting the users' activities. The algorithm iterates between an Expectation step (E-step) which estimates the expectations over the hidden variables using the observations along with the conditional probability density (CPD) of the model, and a Maximization step (M-step) in which the model parameters (i.e. the CPDs) are updated using the expectations of the hidden variables obtained in the E-step.

IV. DATA ACQUISITION

Common tasks, like the ones described so far, demonstrated by human subjects have been acquired by means of a RGB-D sensor. From these image sequences the parameters that regard the configuration of the subject's hand and the configuration of the object need to be extracted, so that they are provided for learning or inference. In order to extract such information we combine the methods in [21], [22] towards a system that can track an object and a hand, while in close interaction, in 3D, from RGB-D input. Tracking is performed as in [22], i.e. through the optimization of an objective function that quantifies the discrepancy between a hypothesis over the scene state and the actual observations. Whereas in [22] the scene amounted to a single hand, in this work, the scene comprises a hand and a rigid object, thus increasing the problem dimensionality to 32 DoFs, as in [21]. At each new tracking frame a new optimization is

TABLE II: Users' everyday tasks

Tasks	Abbrev.	Description
Pour	POUR	Task of pouring from a mug or bottle
Handover	HNDOVR	Task of handing over an object to another person
Tooluse (Hammer)	TLUSE	Hammering a nail
Spray	SPRAY	Spraying from a spray bottle
Dishwash	DSHWSH	Loading an object like a mug in a dishwasher
Drink	DRINK	Drink from a mug or bottle
Shift	SHIFT	Shift object for a one location to another
Sprinkle Salt	SPRINKLE	Sprinkle salt using a salt sprinkler

TABLE III: Action Primitives to perform various tasks

Action Primitive	Abbrev.	Description
Approach	APPRH	Approach to grasp objects in a given space
Approach with twisted hand	APTWH	Approach to grasp objects with inverted hand
Retreat	RETRT	Retreat hand into original position
Putback	PUTBK	Place back the grasped object
Grasp from top	GRTOP	Grasp object from top
Grasp from handle	GRHDL	Grasp object from handle (if any)
Grasp from middle	GRMID	Grasp object from middle
Grasp from tool use end	GRTUE	Grasp object from tool use end
Lift object	LIFT	Lift grasped object
Tilt object	TILT	Tilt grasped object
Un-tilt object	UNTLT	Un-tilt grasped object
Lower object (tool)	LWRTL	Lower object for usage
Raise object (tool)	RAITL	Raise object for usage
Move object towards You	MVTOU	Move object towards you
Release	RELSE	Release the grasped object
Grasp from bottom	GRBOT	Grasp object from bottom
Invert object	INVRT	Invert the grasped object by 180 degrees
Press and release trigger	PERLTGR	Press and release trigger of spray bottle
Shake salt sprinkler	SHAKE	Shake salt sprinkler to sprinkle salt

performed that is initialized in the vicinity of the solution for the previous frame. The reference 3D coordinate system is conveniently defined to reside on the demonstration table (Fig. 1). This is achieved through a chessboard calibration pattern. All objects used were painted blue so as to rely upon a single, uniform appearance model and thus facilitate set-up.

Additionally and with respect to [21], [22], in this work, we deal with the hand-object initialization problem. With the hand, we always expect it at a given position before tracking starts. In order to tackle the more unconstrained problem of initializing the pose of the object, we integrate the registration method of [23] that works over RGB-D input.

V. RESULTS & DISCUSSION

For testing our hypothesis we selected objects from different classes used for a collection of everyday activities. We intentionally selected objects that can be used in the context of more than one activity. As an example, a mug and a bottle can be used both for drinking and pouring. We selected six object (see Fig. 3) to perform tasks as listed in Table II. Data was collected with a single user, who repeated the same task 4 times (to capture variations in performing the same task). The user was asked to perform each task such that its a natural resemblance if the task was performed in a natural environment. The videos and depth data was collected at a rate of 30 frames per second (*fps*). The motion of hand and object was extracted offline using the hand-object tracking algorithm as described in section IV.The output of the tracking algorithm provided data of hand and object motion in the cartesian space and its orientation. The tracker also extracted data feature of each finger joint. Based on visual inspection the tasks were decomposed into a total of 18 meaningful APs listed in Table III. It should be noted that each APs represents a cluster of continuous motion/feature trajectories and not a single instance.

The HHMM model (shown in Fig. 2) was trained and tested using the hand and object motion data captured as described in section IV. The data set was manually labelled for both APs and long term tasks for cross validating the inference accuracy. The labels of APs were used at the lower level to perform supervised learning from the raw stream of observation data listed in Table I. At the higher level (Level 2) of the HHMM model the long term tasks were learned from APs in an unsupervised manner. The features used by the HHMM framework and its corresponding dimension size are listed in Table I. The dataset was divided in two equal halves for training and testing purposes. Expectation Maximization was used to learn user task, and the Maximum Likelihood Estimator was used for inference.

TABLE IV: Inference accuracy of HHMM model to infer long term intentions using APs (Percentage)

Conf. Matrix	POUR	HNDOVER	TLUSE	SPRAY	DSHWSH	DRINK	SHIFT	SPRINKLE
POUR	**52.59**	0.00	0.00	1.00	0.00	37.23	9.02	0.00
HNDOVER	0.00	**98.83**	0.00	0.00	0.00	0.00	0.27	0.00
TLSUE	0.00	0.00	**98.12**	0.00	0.00	0.00	0.00	1.59
SPRAY	0.00	0.00	0.00	**99.82**	0.00	0.00	0.00	0.00
DSHWSH	0.00	0.00	0.00	0.00	**99.58**	0.00	0.00	0.00
DRINK	7.10	0.00	0.00	0.00	0.00	**91.54**	0.40	0.88
SHIFT	0.00	5.33	0.42	0.00	0.00	0.00	**92.16**	1.62
SPRINKLE	0.00	0.00	0.00	0.00	0.00	0.00	24.90	**75.10**

The APs were inferred with an overall accuracy of 88% at the lower level of the HHMM model whereas the long term task was inferred with 91% accuracy (at the higher level). The inference accuracy to predict each APs and the high level tasks are listed in Table V and Table IV respectively.

Out of 18 meaningful APs most of them were inferred with an accuracy higher than 90%. APs such as putback (**PUTBK**), grasp from handle (**GRHDL**), tilt (**TILT**), un-tilt (**UNTLT**) and grasp from bottom (**GRBOT**) are inferred with an accuracy lower than 80%. **PUTBK** is often confused with **LIFT**, this is due to the high level of confusion in the data, where both the actions follow almost the same trajectory in the cartesian space. A very high level of confusion is observed between action states **TILT** & **UNTLT**. This is not surprising as in a continuous space both these actions are performed one after another and hence the framework is unable to clearly discriminate between the two action space. Confusion existed between action class **GRBOT** & **GRMID** due to minimum resolution between the grasp position between the middle and bottom of the object.

At a higher level, apart from task of **POUR** and **DRINK**, all other tasks were inferred with fairly high accuracy. Confusion occurs between these two tasks due to two reasons: firstly there is only a minimal difference in the sequence of APs followed to perform both drinking and pouring and secondly both these tasks are performed with the object belonging to the same class *(mug and bottle)*.

In the experiment, the observation space was restricted to use only finger joints of index, middle and ring fingers. This was mainly because they provided sufficient information to infer most of the APs and inclusion of pinky and thumb joint features did not add any more information and hence were deemed redundant. Also at various occasions the data of the thumb added a lot of noise as it was severely occluded by the object and was not observed by the tracking algorithm.

VI. CONCLUSION & FUTURE WORK

In this paper we evaluated our approach of inferring users' long term task from different APs using a HHMM based probabilistic model. The HHMM framework allows to flexibly divide a task into a hierarchy. The long term tasks were considered sequential combination of APs. The

TABLE V: Inference accuracy of APs (Percentage)

Conf. Matrix	APPRH	APTWH	RETRT	PUTBK	GRTOP	GRHDL	GRMID	GRTUE	LIFT	TILT	UNTLT	LWRTL	RAITL	MVTOU	RELSE	GRBOT	INVRT	PERLTGR	SHAKE
APPRH	**99.37**	0.00	0.10	0.00	0.00	0.00	0.53	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
APPWH	0.00	**100.00**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
RETRT	0.00	0.00	**95.20**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	4.80	0.00	0.00	0.00	0.00
PUTBK	0.00	0.00	0.00	**72.38**	0.41	2.31	1.84	0.00	8.57	0.20	4.56	3.27	0.41	0.00	0.27	0.00	0.34	5.44	0.00
GRTOP	0.00	0.00	0.00	0.00	**95.25**	0.00	0.00	0.00	4.75	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
GRHDL	1.57	0.00	0.00	0.79	0.00	**77.17**	8.66	0.00	0.00	0.00	0.00	9.45	1.57	0.00	0.79	0.00	0.00	0.00	0.00
GRMID	0.32	0.00	0.00	0.00	0.00	0.80	**95.38**	0.00	2.71	0.00	0.00	0.00	0.00	0.00	0.00	0.80	0.00	0.00	0.00
GRTUE	0.00	0.00	0.00	0.00	0.00	0.00	7.94	**92.06**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
LIFT	0.04	0.00	0.00	5.00	1.59	0.08	5.65	0.61	**83.82**	0.16	0.20	0.00	0.00	0.04	0.08	0.45	0.00	2.24	0.04
TILT	0.00	0.00	0.00	0.20	0.00	0.00	0.98	0.00	4.13	**74.61**	18.11	0.00	0.00	0.98	0.20	0.00	0.00	0.00	0.79
UNTLT	0.00	0.00	0.00	4.23	0.40	0.00	0.00	0.00	0.40	33.27	**57.26**	0.00	0.00	0.00	0.20	0.00	0.00	0.00	4.23
LWRTL	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**98.51**	1.49	0.00	0.00	0.00	0.00	0.00	0.00
RAITL	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**100.00**	0.00	0.00	0.00	0.00	0.00	0.00
MVTOU	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.16	0.00	0.00	0.00	**98.84**	0.00	0.00	0.00	0.00	0.00
RELSE	0.00	0.00	4.23	2.77	0.00	0.00	0.15	0.00	1.02	0.00	0.00	0.00	0.00	0.00	**91.84**	0.00	0.00	0.00	0.00
GRBOT	0.00	0.00	0.00	0.00	0.00	0.00	20.00	0.00	0.91	0.00	0.00	0.00	0.00	0.00	0.00	**79.09**	0.00	0.00	0.00
INVRT	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**100.00**	0.00	0.00
PERTGR	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**100.00**	0.00
SHAKE	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**100.00**

framework was tested on a set of task sequences collected for different objects used in everyday life. The hierarchical framework proved to be a powerful tool to divide tasks both vertically for natural language description of different tasks as APs and horizontally where the continuous observations are clustered into different APs.

The HHMM framework has been tested with 6 objects belonging to different classes to perform tasks listed in Table II. Our future goals are mainly aimed in three directions *(data set, observation model & learning)*. Firstly, we plan to add new object classes such as *bowl, remote control, plate, ball* and also add more objects belonging to same object class (for e.g. adding mugs of different sizes to the object class mug, adding tools such as *screwdriver, plier, knife*). With the observation data, in the existing work we used the raw data features extracted by the tracking algorithm. Work is in progress to apply discretisation and feature extraction techniques such as the Gaussian Process Latent Variable Model proposed in [12] with the raw data to enhance the inference accuracy of APs. Finally, we we hope to be able to learn the entire HHMM model in an unsupervised manner. We also plan to release the dataset to the research community.

VII. ACKNOWLEDGEMENT

This work was carried out during a visit by Mr Mitesh Patel of the Royal Institute of Technology, Sweden and was partly supported by European Union FP7 project RoboHow.Cog (FP7-ICT-288533). Mr Patel gratefully acknowledges the financial support to undertake the visit received from the Department of Industry, Innovation, Science, Research and Tertiary Education (DIISRTE), Australia, through its Endeavours Award research scholarship scheme.

REFERENCES

[1] S. Schaal, A. J. Ijspeert, and A. Billard. Computational approaches to motor learning by imitation. *Philosophical transaction of the Royal Society of London, series B*, 358(1431):537–547, 2003.

[2] V. Krüger, D. Herzog, S. Baby, A. Ude, and D. Kragic. Learning actions from observations. *IEEE Robotics & Automatation Magazine*, 17(2):30–43, 2010.

[3] D. Kulic, D. Kragic, and V. Krüger. Learning action primitives. In *Visual Analysis of Humans*, pages 333–353. 2011.

[4] D. Newtson, G. A. Engquist, and J. Bois. The objective basis of behaviour units. *Journal of Personality and Social Psychology*, 35(12):847 – 862, 1977.

[5] G. Rizzolatti, L. Foggassi, and V. Gallese. Neurophysiological mechanisms underlying the understanding and imitation of action. *Nature Reviews Neuroscience*, 2:661–670, 2001.

[6] O. C. Jenkins and M. J. Mataric. Performance-derived behavior vocabularies: Data driven acqusition of skills from motion. *International Journal of Humanoid Robotics*, 1(2):237–288, 2004.

[7] S.M. Khansari-Zadeh and A. Billard. Imitation learning of globally stable non-linear point-to-point robot motions using nonlinear programming. In *IEEE/RSJ International conference on Intelligent Robots and Systems*, pages 2676 –2683, oct. 2010.

[8] H. Dindo and G. Schillaci. An adaptive probabilistic approach to goal-level imitation learning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4452–4457, Oct. 2010.

[9] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal. Learning and generalization of motor skills by learning from demonstration. In *IEEE/RSJ International Conference on Robotics and Automation*, pages 1293–1298, 2009.

[10] B. Nemec and A. Ude. Action sequencing using dynamic movement primitives. *Robotica*, 30(5):837–846, 2012.

[11] D. Song, K. Huebner, V. Kyrki, and D. Kragic. Learning task constraints for robot grasping using graphical models. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1579 –1585, oct. 2010.

[12] D. Song, C. H. Ek, K. Huebner, and D. Kragic. Multivariate discretization for bayesian network structure learning in robot grasping. In *IEEE/RSJ International Conference on Robotics and Automation*, pages 1944 –1950, may 2011.

[13] N.T. Nguyen, D.Q. Phung, S. Venkatesh, and H. Bui. Learning and detecting activities from movement trajectories using the hierarchical hidden markov model. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 955 – 960, 2005.

[14] L. Liao. *Location-Based Activity Recognition*. PhD thesis, University of Washington, 2006.

[15] M. Patel, J. V. Miro, and G. Dissanayake. A hierarchical hidden markov model to support activities of daily living with an assistive robotic walker. In *4th IEEE RAS EMBS International Conference on Biomedical Robotics and Biomechatronics*, pages 1071 –1076, 2012.

[16] F. V. Jensen. *An Introduciton to Bayesian Networks*. UCL Press, 1996.

[17] S Iba, C. J. J. Predis, and P. K. Khosla. Interactive multi-model robot programming. *International Journal of Robotics Research*, 24(1):83–104, 2005.

[18] D. Kragic, P. Marayong, M. Li, A. M. Okamura, and G. D. Hager. Human-machine collaborative systems for microsurgical applications. *International Journal Robotics Research*, 24(9):731–741, 2005.

[19] S. Fine, Y. Singer, and N. Tishby. The hierarchical hidden markov model: Analysis and applications. *Machine Learning*, 32:41–62, 1998.

[20] K. P. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of Califronia, Berkeley, 2002.

[21] I. Oikonomidis, N. Kyriazis, and A. Argyros. Full dof tracking of a hand interacting with an object by modeling occlusions and physical constraints. In *IEEE International Conference on Computer Vision*, pages 2088 – 2095, 2011.

[22] I. Oikonomidis, N. Kyriazis, and A. Argyros. Efficient model-based 3d tracking of hand articulations using kinect. In *Proceedings of the British Machine Vision Conference*, pages 101.1–101.11. BMVA Press, 2011.

[23] C. Papazov and D. Burschka. An efficient ransac for 3d object recognition in noisy and occluded scenes. *Computer Vision–ACCV 2010*, pages 135–148, 2011.