Noname manuscript No.
(will be inserted by the editor)

Learning Object, Grasping and Manipulations Activities using Hierarchical HMMs

Mitesh Patel · Jaime Valls Miro · Danica Kragic · Carl Henrik Ek · Gamini Dissanayake

Received: date / Accepted: date

Abstract This article presents a probabilistic algorithm for representing and learning complex manipulation activities performed by humans in everyday life. The work builds on the multi-level Hierarchical Hidden Markov Model (HHMM) framework which allows decomposition of longer-term complex manipulation activities into layers of abstraction whereby the building blocks can be represented by simpler action modules called action primitives. This way, human task knowledge can be synthesised in a compact, effective representation suitable, for instance, to be subsequently transferred to a robot for imitation. The main contribution is the use of a robust framework capable of dealing with the uncertainty or incomplete data, and the ability to represent behaviours at multiple levels of abstraction for enhanced taks generalisation. Activity data from 3D video sequencing of human manipulation of different objects handled in everyday life is used for evaluation. A comparison with a mixed generative-discriminative hybrid model HHMM/SVM (Support Vector Machine) is also presented to add rigour in highlighting the benefit of the proposed approach against comparable state of the art techniques.

Mitesh Patel, Jaime Valls Miro and Gamini Dissanayake
Faculty of Engineering and IT, University of Technology Sydney (UTS), 15 Broadway, Ultimo, NSW-2007, Australia
Tel.: +612-95143146
Fax: +612-95142655
E-mail: mitesh.k.patel@student.uts.edu.au,
jaime.vallsmiro@uts.edu.au, gamini.dissanayake@uts.edu.au

Danica Kragic, Carl Henrik Ek
Members of the Computer Vision & Active Perception Lab., Centre for Autonomous Systems, School of Computer Science and Communication, The Royal Institute of Technology (KTH), Stockholm, Sweden
E-mail: dani@kth.se, chek@kth.se

Keywords Hierarchical Hidden Markov Model (HHMM) · Action Primitives · Grasping and Manipulation · Human Daily Activities

1 Introduction & Motivation

Human behaviours are inherently complex and extracting a represention from raw sensory data is a challenging undertaking. One of the most desired objectives in the field of human-robot interaction is to endow robots with the capability of learning human activities through simple observation - imitation learning being one of the most common approaches explored (Schaal et al., 2003).

For the specific case of learning object grasping and manipulation activities there has been a growing interest in expressing these as a combination of *Action Primitives* (APs) (Krüger et al., 2010). Research 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 (Newtson et al., 1977), (Schaal et al., 2003),(Kulic et al., 2011). For example, as shown in Figure 1 the activity of *pouring water from a mug* could be decomposed into a sequence of actions that can be regarded as atomic in that given the observed data these cannot be decomposed further, e.g. *approach-grasp-lift-tilt-untilt-place_back-release-retreat.* Arguments raised in the field of neuroscience (Rizzolatti et al., 2001) reinforces the concept that human actions are composed of APs in a similar way to human speech, where utterances of words are broken down into phonemes. Hence the use of a grammar based on APs appears an attractive proposition to represent activities, in that they allow for a "symbolic" description of more complex actions. This

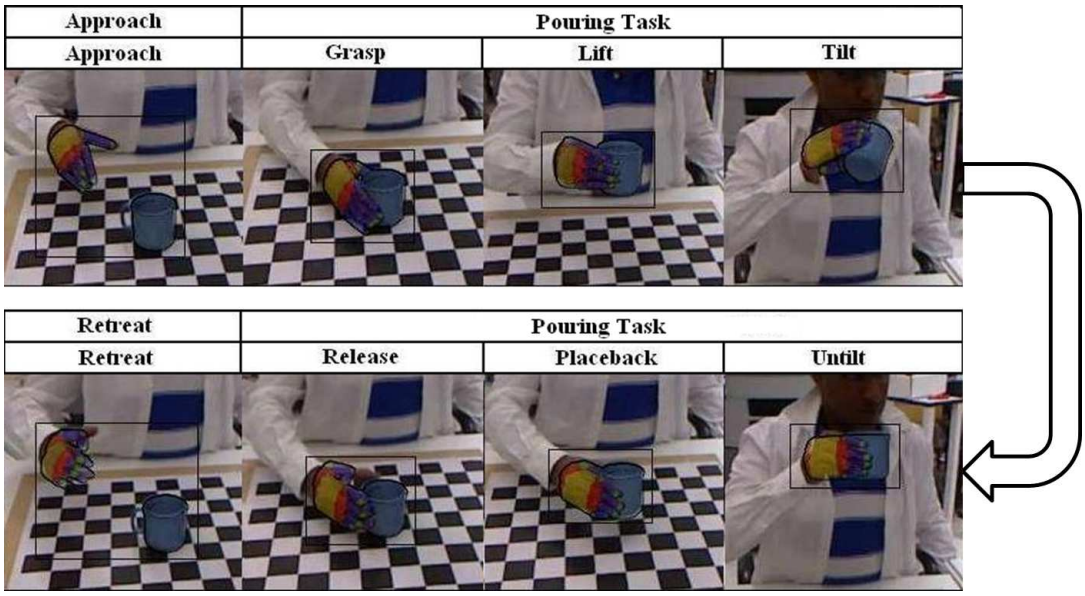


Fig. 1 Activity of *Pouring* water from mug subdivided into action primitives. Each image depicts the output of hand-object tracking algorithm.

is also in accordance with the concept, in a humanoid robotic context, that the process of recognising human tasks may be regarded as understanding sequential human behaviours which, in turn, consists of interpreting a sequence of action primitives (Jenkins and Mataric, 2004). Along with the advantage of a top-down approach (complex activities decomposed into APs), the framework also enables a bottom-up approach whereby APs can be shared to construct different activities - an attractive proposition e.g. for robotic arms to be able to generalise their learning from human teachings.

2 Proposition

In this paper we exploit a temporal probabilistic network embodied in a *Hierarchical Hidden Markov Model* (HHMM), and show how it can be used for learning and representing object grasping and manipulation activities. Given the inherent level of uncertainty, noise, and ambiguity in the sensor signals used to perceive human tasks, modelling human manipulative actions in a deterministic manner is a challenging premise. Thus, stochastic or probabilistic models are commonly employed.

The proposed model builds upon alphabets of APs which are combined to describe complex human activities. The hierarchical nature of the framework allows decomposition of a typical activity into different levels of action representation. Moreover, the algorithm is robust to uncertain or incomplete data to infer user's long-term intent. In the manipulative space hereby presented APs are learned and inferred by observing hand-object interactions and their motion in the Cartesian space, whereas the higher level activities are inferred by learning the time-sequence of APs. The framework proves to be a strong tool for learning and synthesizing complex activities as it enables the robot to not only learn activities through imitation, but also to reproduce the learned activities by combining APs in different sequences to perform higher level activities. To this end, for the robot to efficiently imitate or perform tasks similar to those performed by their human counterpart, the string of APs generated by decomposing activities are such that they can map directly across to actions (i.e movements of the arm), which a robot can then perform sequentially to complete a "human-like" activity. For instance a humanoid robot would learn to pour water with the right arm, as taught by a right-handed human teacher, but would be able to generalise these movements to perform a similar action with the left arm, or as part of a similar activity such as adding ingredients during cooking.

For completeness, the proposed HHMM framework is also compared with a HHMM/SVM hybrid model, motivated by the exceptional performance of discriminative models in general in relevant state-of-the-art literature. Generative-discriminative hybrid frameworks have been successfully explored by the research community in areas such as automatic speech recognition, facial/gesture expression and more (Abou-Moustafa et al., 2004). The HHMM/SVM hybrid framework uses the strong kernel projection characteristics of the SVM

classifier, which are then combined with the HHMM model to exploit temporal relationships. Results highlight not only the inherent superior generalization capabilities of the proposed technique, but also their practicality given their unsupervised nature, and better suitability for novelty detection so as to be able to incorporate new relevant data into the models.

3 Related Work

Probabilistic models have been used extensively by the AI community in particular to represent complex systems with prominent uncertainty (Jensen, 1996). These models have found its applicability in the field of robotics given their inherent ability to handle sensor noise and data ambiguity, thus capturing both spatial and temporal variability in their movements and perception of their surroundings. 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 list includes aviation monitoring (Heinze, 2003), sign language and gesture modelling (Iba et al., 2005), assistive robotics (Patel et al., 2012), skills transfer (Dillmann et al., 1999), robot assisted surgery (Kragic et al., 2005) and many more.

Learning by imitation is an approach that has been used by roboticists for bootstrapping learning of robot activities based on human observation, a relevant context for this work. Preliminary work done by Ijspreet and his colleagues used a Control Policy (CPs) based approach to represent complex dynamical systems based on human movements (Ijspeert et al., 2002). These CPs, which represent various human like movement plans, are derived based on ease of representation, compactness, robustness against changes in the dynamic environment, re-usability and overall simplicity in learning different human movement trajectories. This Dynamic Motion Primitive (DMP) based framework was later on illustrated in a number of application related to humanoid robotics which involved planning, movement recognition, perception-action coupling, imitation and general reinforcement learning (Schaal et al., 2004). Khansari-Zadeh and Billard (Khansari-Zadeh and Billard, 2010) proposed the *Stable Estimator of Dynamical Systems (SEDS)*, a method for learning the parameters of a time invariant dynamical system to ensure that all motions closely follow the demonstrations while ultimately reaching and stopping at the target. The activities learned by the SEDS were simple tasks such as moving an object from point-to-point. Dindo and Schillaci (Dindo and Schillaci, 2010) proposed a *Growing Hierarchical Dynamic Bayesian Network (GHDBN)* to recognise the skills being observed and to reproduce them by exploiting the generative characteristics of the model. The model learned and reproduced three actions i.e. *dislocate, approach* and *hit*. Pastor *et. al.* (Pastor et al., 2009) 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*. Aksoy *et. al.* used a Semantic Event Chain (SEC) based approach to represent the relations between objects and hand at decisive time points during a manipulation activity Aksoy et al. (2011). The time points defined using SEC were descriptive for distinguishing different manipulation activity. In their recent work, Nemec and Ude (Nemec and Ude, 2012) also used a DMP based system to represent primitive movements. The DMP library used in their experiment consisted of activities like *reaching, pouring, wiping, shaking, cutting and power grasps*.

Kruger *et. al.* proposed a *Parametric Hidden Markov Model (PHMM)* to represent various action primitives (Krüger et al., 2010). The framework was trained in an unsupervised manner and synthesized movement trajectories as a function of their desired effect on the object (e.g. *approach, grasp, push forward, push side, move side, rotate, remove*). Song *et. al.* used structure learning to exploit the dependencies between hand and object to generate the structure of a Bayesian Network (BN) (Song et al., 2011a), (Song et al., 2011b). The evolved structure was used to predict the activity performed by the user based on the type of action, and the object being manipulated. However, the prediction of these activities was done based on grasp instances, and did not exploit features from the entire trajectory as followed by the arm to perform a given activity.

Our work suggests the use of a HHMM to better exploit temporal constraints for grasp and manipulation activities. The HHMM theoretical framework hereby proposed has been applied in several application areas. Nguyen *et. al.* (Nguyen et al., 2005) 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 activities inferred were based on location semantics. Kawanaka *et. al.* (Kawanaka et al., 2005) used a HHMM model for recognising human activities as a series of actions from image sequences. Each target activity had its own individual model which were clubbed as sub-model within the HHMM framework. In the area of ubiquitous computing, Liao (Liao, 2006) used a HHMM framework to infer user's mode of transportation, destination location

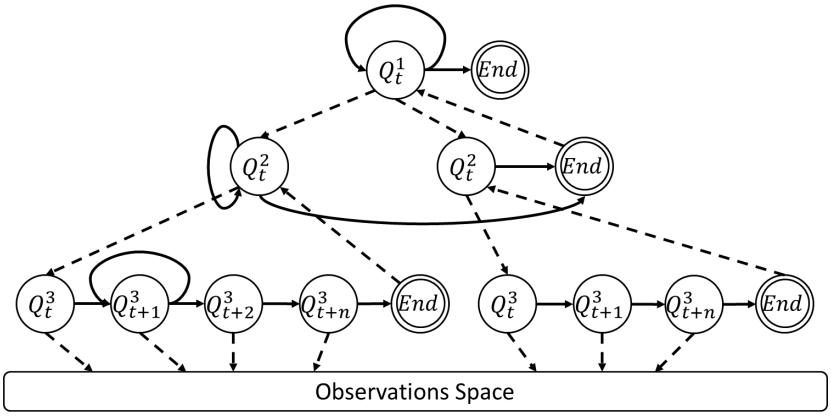


Fig. 2 Example of a three level HHMM model where solid arcs represent horizontal transitions between states, and dashed arcs represent vertical transitions, i.e., connections between sub-HMMs. Double-ringed states represents end states (at least one per sub-HMM), where control flow is returned to the parent (calling) state. Each node at level 3 emits a single state based on the distribution over the observation space.

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. In work related to assistive robotic walkers (Patel et al., 2012), a HHMM framework was deployed to infer navigational and non-navigational intentions of a walker user. The hierarchical nature of the framework allowed learning of typical activities of daily living such as *stand up* or *going to kitchen*.

HMM-SVM hybrid models have also been widely used in areas such as automatic speech recognition (Stadermann and Rigoll, 2004), tele-operation (Castellani et al., 2004) or modelling of facial action temporal dynamics (Valstar and Pantic, 2007). Stadermann used a SVM/HMM hybrid model for speech recognition which combines the strong classification capabilities of SVM with the time varying modelling capability of HMM model (Stadermann and Rigoll, 2004). Valster and Pantic also exploited the capabilities of SVM/HMM hybrid model for facial action recognition. In this application the SVM classified the distinction between the temporal (facial expression) phases at a single point in time which were then combined over a time period by the HMM model to predict the temporal dynamics (Valstar and Pantic, 2007). A similar technique was used by Castellani and colleagues for analysing and segmenting various tele-operation activities (Castellani et al., 2004). In all these approaches the strong characteristics of SVM to handle non-linear data through kernel induced feature maps was exploited to discriminate segments, which were in turn utilised by the HMM to model the temporal relationship between data points.

Our work proposes the use of a probabilistic framework capable of representing an entire grasping and manipulation task by decomposing it into clusters of APs. The approach is novel in that firstly, the entire activity sequence is clustered into a pool of different APs and secondly, the unified probabilistic framework exploits spatial relationships to learn both, APs and time dependent relationship between them, to accurately predict the complex manipulation activities at the highest level of abstraction. Clustering activities into different APs becomes an important criteria as the time taken by any user to perform a given activity will vary (even for the same user), which implies a high variability in users remaining within a given (action primitive) state. For that, the use of hierarchical models with specific conditions to model the end of sub-processes is an important proposition. Considering a unique user state at each time instance makes it computationally intractable as the state space would grow unbounded.

4 Hierarchical Hidden Markov Model (HHMM)

The proposed HHMM framework is capable of structuring stochastic processes at multiple levels. The HHMM is an extension of the traditional HMM model, designed to model domains with hierarchical structure including such with dependencies at multiple length/time scales (Fine et al., 1998). In a 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" (Murphy, 2002).

The example shown in Figure 2 provides an intuitive description of the process. The states at the highest level correspond to the abstract states, are themselves

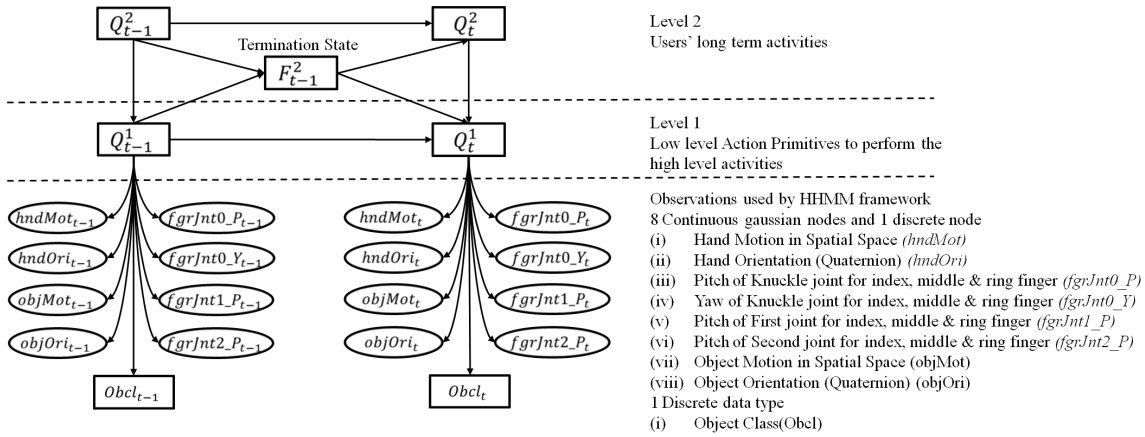


Fig. 3 HDBN representation of the HHMM Model used to infer action primitives and long term user activities using hand and object features (described in Table 2)

governed by sub-HHMMs, entering into states Q^2. Since a state at level 2 is abstract, it enters its child HMM via its subsequence states Q^3. The horizontal transition in each child HMM (at level 3) emits unique state w.r.t the observations perceived by the model and is hence referred to as production state. Once the sub-HMM reaches the end state, the control is returned to the higher level, from wherever the sub-HMM sequence was called from. This is done recursively till the time the control is returned to the highest abstract state (level 1). The abstract state can transit to the next possible state only after all the sub-HMM at lower level are terminated (Murphy, 2002).

The hierarchical nature allows decomposition of the problem at different levels of abstraction thereby facilitating exploration (long term planning/activities) and exploitation (short term planning/action primitives) within the same framework. In the paradigm of learning long term task/activities from APs, the high-level activities call the more refined low-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.

4.1 Representation

A HHMM framework can be formally represented as a Hierarchical Dynamic Bayesian Network (H-DBN) as shown in Figure 3. Its structure comprises of three types of nodes, Q_t^d, O_t, F_t^d 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_t^d 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.
- As the true state of the user is hidden, observation nodes O_t are required that provide user/environment information. These are modelled either as a mixture of Gaussian (μ, Σ) or as discrete $P(O_t|Q_t^d)$ node.
- F_t^d is the terminating state which specifies the natural completion of a sub-HMM and returns 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 probabilities. The prior and the transition probabilities are defined at every level (d). Once defined these probabilities are further optimised from data using the Expectation-Maximisation (EM) algorithm.

4.2 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

$$P(Q_1^2) = \pi^2(j)$$
$$P(Q_1^1) = \pi_k^1(j) \quad (1)$$

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

4.3 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 Figure 3) 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:

$$P(Q^2_t = j|Q^2_{t-1} = i, F^2_{t-1} = f) = \begin{cases} A^2(i,j) \text{ if } F^2_{t-1} = 0 \\ \pi^2(j) \quad \text{if } F^2_{t-1} = 1 \end{cases} \quad (2)$$

Similarly, the states at the intermediate level (level 1 in Figure 3) 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 as,

$$P(Q^1_t = j|Q^1_{t-1} = i, F^2_{t-1} = f, Q^2_t = k) = \begin{cases} A^1_k(i,j) \text{ if } F^2_{t-1} = 0 \\ \pi^1_k(j) \quad \text{if } F^2_{t-1} = 1 \end{cases} \quad (3)$$

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

4.4 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 as:

$$P(F^2_t = 1|Q^2_t = k, Q^1_t = i) = A^2_k(i, end) \quad (4)$$

4.5 Observation Model

The observation model signifies the probability of 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 are given by:

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

4.6 Learning and Inference

Different techniques can be used for learning the HHMM model, both supervised and unsupervised. Expectation Maximisation (EM) (Blimes, 1998) and its variants are some of the most popular statistical techniques used for unsupervised learning. In realistic circumstances it is difficult to obtain labelled data, hence an unsupervised mode of learning is preferable. We used EM for learning the model and maximum likelihood estimator to predict user activities. The EM algorithm iterates between an Expectation step (E-step) and Maximization step (M-step). In each E-Step it estimates the expectations (distributions) over the latent variables using the observations along with the conditional probability density (CPD) of the model. Then in the M-step the model parameters (i.e. the CPDs) are updated using the expectations of the hidden variables obtained in the E-step. Each iteration would continue to improve the estimates of the hidden variables and will eventually converge to a local optimum.

5 Problem Specific HHMM Framework

The HHMM framework used to test our proposition is shown in Figure 3. User state/activities are inferred at the top level whereas the intermediate level represents the APs (shown in Figure 3) while the lowest level corresponds to the features of object-hand interaction in the Cartesian space. In everyday life a single object can be used to perform many activities (e.g. a mug can be used for drinking, pouring or handing it over to another person), hence it is difficult to predict the user activity when he/she is approaching to grasp the object, but it becomes more apparent after the object has been grasped. Similarly, after accomplishing the desired activity, the action of retreating the hand after releasing the object cannot be described as part of the activity 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 activity listed in Table 1, but are described as APs independent of any activity. In our framework, such independent APs are inferred at both levels of hierarchy. To better illustrate this concept, consider the example in Figure 1. The user first approaches to grasp the mug, which has the same AP defined at both levels. This means that the specific activity cannot be inferred without the object being grasped. Once the object is grasped, the activity can be inferred based on the type of grasp and the object. Hence, the HHMM model will infer activities at the higher level (2) and the action primitives at the lower level (1). After releasing

Activities	Abbrev.	Description
Pour	POUR	Activity of pouring from a mug or bottle
Handover	HNDOVR	Activity of handing over an object to another person
Tool Use (Hammer)	TLUSE	Hammering a nail
Spray	SPRAY	Spraying from a spray bottle
Dish Wash	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 1 Users' everyday activities

the object the AP of retreating being independent from any activity sequence will be thus inferred at both levels.

At the observation level, features are extracted using a hand-object tracking algorithm (details are given in Section 6). It represents the interaction between the hand and object and its movement in Cartesian space.

Fig. 4 Objects used to perform manipulation activities

6 Data Acquisition

In order to validate our proposed approach, we collected data using an RGB-D kinect sensor while the human subject demonstrated the grasping and manipulation activities. The parameters that describe the configuration of the users' hand and the configuration of the object while performing the activities need to be extracted from the 3D video stream data. The extracted features which involves the interaction between the hand and object should be such that they can be mapped to the motion of a robotic arm for activity synthesis/imitation. In order to extract such information we combined the methods presented in (Oikonomidis et al., 2011b) and (Oikonomidis et al., 2011a) towards a system that can track both the hand and object while they are interacting (in Cartesian space). Tracking of the hand is performed using the technique described in (Oikonomidis et al., 2011a), which optimizes the objective function that quantifies the discrepancy between a hypothesis over the scene state and the actual observations. The tracking algorithm also accommodated the tracking of the object and its motion in Cartesian space. At each new frame a new tracking optimization is 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 seen in Figure 1), which becomes a chess-board calibration pattern. All objects used in the manipulative activities were painted blue, as per Figure 4, so as to rely upon a single, uniform appearance model for tracking, thus facilitating the overall set-up.

To initialise the hand and object position we employed a similar technique to the one specified in (Oikonomidis et al., 2011b), (Oikonomidis et al., 2011a) and (Papazov and Burschka, 2011). To successfully track the hand, the tracking algorithm expects the hand to be at a given initial position in the space. To initialise the pose of the object, we integrated the tracking algorithm with the RGB-D based registration method used by Papazov (Papazov and Burschka, 2011).

The features extracted in the experimental results to validate the proposed work are listed in Table 2. They consist of the 3D motion (translation and rotation) of the hand and the object being manipulated. The features in the data also include a selection of the rotational joint movements of three of the fingers, index, middle and ring. The derived trajectory provided information about the motion of the hand and object, whereas the rotational motion (yaw, pan, tilt) added information about their corresponding orientation in space. Furthermore, the movement of the finger joints provided details about the grasping of the objects. All these data features were utilised to predict the APs at the lower level.

It is worth noting that the primary goal in this work is the representation of human grasping and manipulation so that these behaviours can effectively be learned from a human teacher and ultimately transferred to a robot arm. Kinematic models and DOFs between a human arm and a robotic manipulator differ, thus the paths followed by both in exercising a manipulation activity will diverge. However, for a capable anthropomorphic arm the interactions between a robotic arm

Table 2 Hand and 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

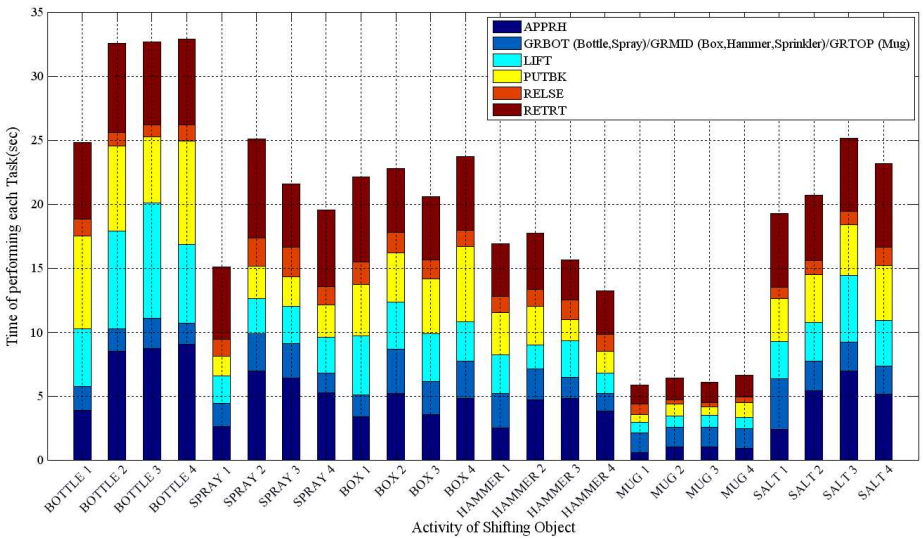


Fig. 5 Time taken by each action primitive (APs) to perform the activity of shifting objects. Note that the time taken for shifting the same object and the time spend within each AP varies between same and different objects

and the objects in their surroundings (e.g. grasping the object with a particular pose in order to accomplish the desired activity) will be of similar nature - subject of course to their differing kinematic arrangements. As such, the APs learned by the robot (*GRTOP, TILT* etc.) and the sequences needed to accomplish a given task are directly transferable to any grasping manipulator of sufficient dexterity.

7 Results

To test the proposed methodology, we used a selection of everyday objects from different classes. We intentionally selected objects that can be used in the context of more than one activity, e.g. a mug and a bottle which can be used both for drinking and pouring. We selected the six objects depicted in Figure 4 to perform the activities listed in Table 1. Data was collected with a single user, who repeated the same activity 4 times

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

Table 3 Action Primitives to perform various activities

to capture variations which might occur in performing the same activity. The user was asked to perform each activity such that it resembles natural execution. The video and depth data was collected at a rate of 30 frames per second. The motion of hand and object was extracted off-line using the hand-object tracking algorithm described in Section 6. The output of the tracking algorithm provided data of hand and object motion in the Cartesian space and its orientation. The tracker also extracted the features for each finger joint. Activities were decomposed into a total of 19 interpretable APs based on visual inspection, and are collected in Table 3. It is important to emphasize that each AP represents a feature set that consists of a cluster of continuous, time-varying trajectories and not a single instance.

Due to the time variation in performing different activities, the time spent in executing each AP will vary. That would be the case even if its the same activity that is being repeated over and over again. To illustrate this, Figure 5 shows an example of the time taken to perform the activity of SHIFT which involves shifting different objects from one location to other. It can be noted how the time taken for each AP in a given activity varies even if it is repeated on the same object. For example, when comparing the activity of shifting a bottle (as shown in Figure 5), *BOTTLE 1* took significantly less time than the other three times (*BOTTLE 2, BOTTLE 3, BOTTLE 4*). This variation in the activity directly effects the time taken to undertake each AP.

The HHMM model (shown in Figure 3) was trained and tested using the hand and object motion data captured described in Section 6. The data set was manually labelled for both APs and long term activities for

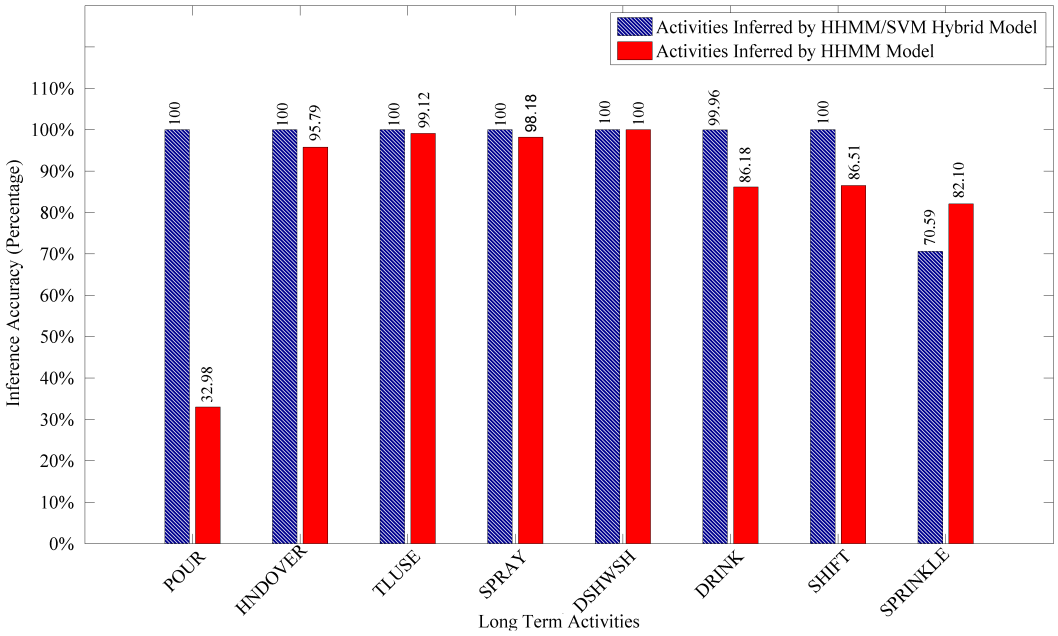


Fig. 6 Activities inference accuracy by HHMM and HHMM/SVM Hybrid Models

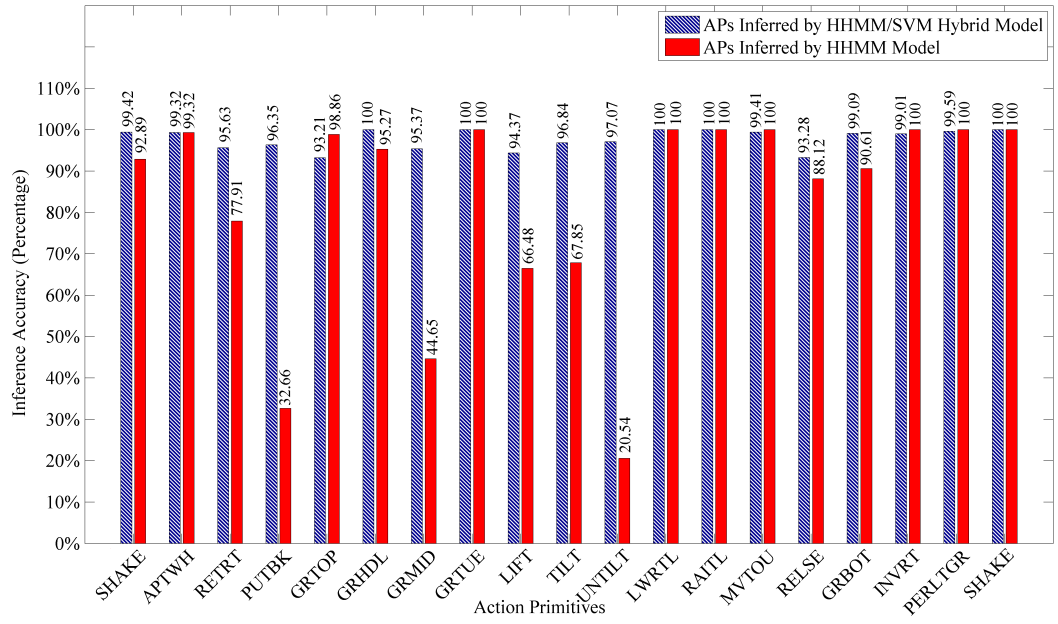
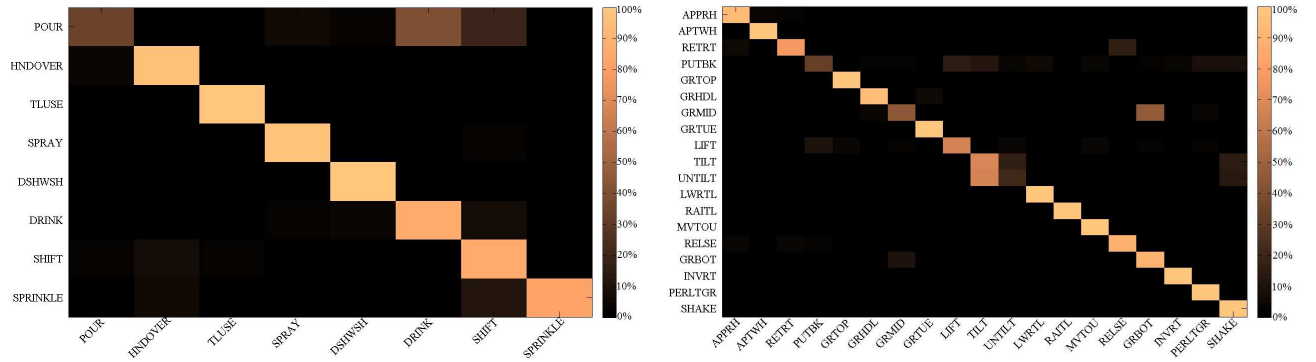


Fig. 7 APs inference accuracy by HHMM and HHMM/SVM Hybrid Models

cross validating the inference accuracy. We divided the data set into two equal halves for training and testing purposes. We used the BNT toolbox (Murphy, 2002) to learn and infer APs and long term activities using the proposed HHMM model. Expectation Maximisation (EM) was used to learning APs and high level activities where as Maximum Likelihood Estimator was used for inference. The features used by the HHMM framework and its corresponding dimension size are listed in Table 2.

The APs were inferred with an overall accuracy of 72% at the intermediate level (level 1) of the HHMM model whereas the long term activities was inferred with 86% accuracy (at the higher level). The inference accuracy to predict each AP and the high level activities are graphically depicted in Figure 7 and 6 respectively.



(a) Activities inferred by HHMM Model

(b) APs inferred by HHMM Model

Fig. 8 Confusion matrix of inferring activities at the high level and APs at the intermediate level by HHMM model

Most of the APs were inferred with an accuracy higher than 72%. APs such as putback (**PUTBK**), tilt (**TILT**), un-tilt (**UNTLT**), grasp object from middle (**GRMID**) and lift (**LIFT**) are inferred with an accuracy lower than 70%. **PUTBK** is often confused with **LIFT** (can be seen in Figure 8(b)), this is due to the high level of confusion in the data, since both actions follow almost the same trajectory in the Cartesian space. A very high level of confusion is observed between action states **TILT** and **UNTLT**. This is not surprising as in the continuous space both these actions are performed one after another, and hence the framework is unable to clearly discriminate between them. Lastly, high level of confusion exists between the state of grasping the object of middle and bottom due to unavailability of relevant information such as distance offset between the center of object and grasping points.

At a higher level, apart from the activity of **POUR** and **DRINK**, all other activities were inferred with fairly high accuracy (refer to confusion matrix in Figure 8(a)). Confusion occurs between these two activities as there is minimal difference in the sequence of APs followed to perform both drinking and pouring.

8 Comparison with HHMM/SVM Hybrid Model

We also compared the accuracy of the HHMM model with that of a hybrid HHMM/SVM model. HMM/SVM hybrid model has been successfully used in a number of application (Bishop and Lasserre, 2007) (Castellani et al., 2004) (Valstar and Pantic, 2007) (Stadermann and Rigoll, 2004), where the excellent discrimination performance of SVM complements the temporal modelling properties of HMM to provide a higher inference accuracy. In this work, a SVM was used to predict the APs at a single time instance which are then combined in a temporal space within the HHMM model to predict high level activities. The HHMM/SVM hybrid model used for comparison is shown in Figure 9. To make the comparison fair, we used a Hierarchical HMM framework instead of a flat HMM model so that the self transition and inter state transition characteristics at level 1 remains the same for both the models. The high level activities were inferred at level 2 with an overall inference accuracy of 95% (see Figure 6). The APs were inferred with an overall accuracy of 97% at level 1 (see Figure 7), which corresponds to a direct mapping of the APs classified by the SVM model. The confusion matrix of high level activities and APs inferred by the HHMM/SVM hybrid model are depicted in Figure 10(a) and 10(b) respectively.

Most of the APs are inferred with around the same accuracies with both HHMM and HHMM/SVM hybrid model except for **PUTBK, GRMID, LIFT, TILT, ULTILT** . The HHMM model is less able to discriminate between these classes as described in Section 7. However, SVM is able to predict these APs with high accuracy which is not surprising as SVM possess strong capability to discriminate between these classes with minimal difference in observation. The HHMM/SVM hybrid model outperforms HHMM model in inferring the high level activities given the strong classification of APs by the SVM classifier as compared to the HHMM model.

9 Discussion

The HHMM/SVM hybrid model appears an overall stronger inference engine, yet that is somewhat misleading when put into the correct context, and the authors advocate for the benefits that a HHMM model exhibit over a HHMM/SVM hybrid model when the appropriate criteria to model real-life complex manipulation tasks are taken into consideration, as described next.

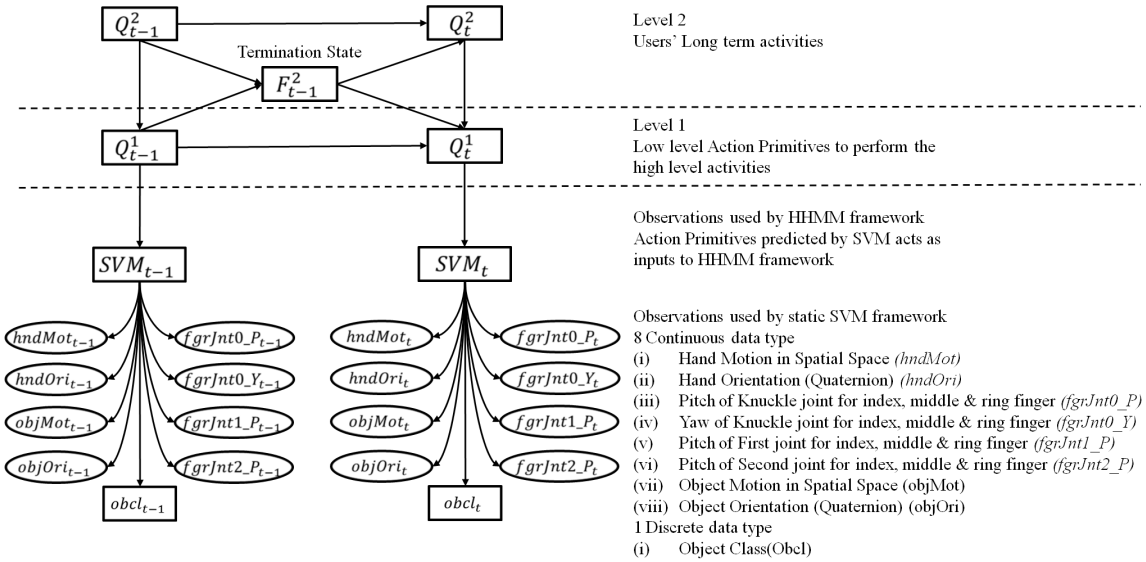
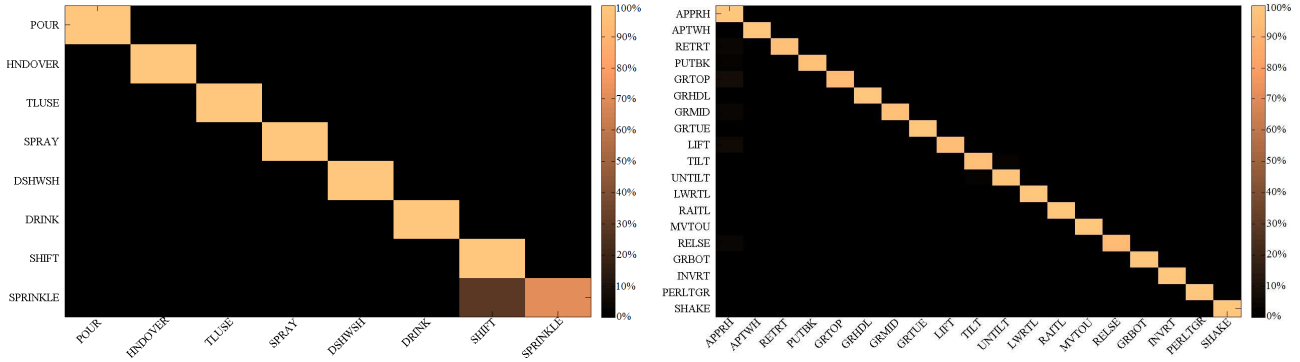


Fig. 9 HHMM/SVM Hybrid Model used to infer action primitives and long term user activity using different hand and object features. The SVM classifier at the lower level classifies action primitives using hand and object features which are then used by the HHMM framework to predict the long term activities.



(a) Activities inferred by HHMM/SVM Model

(b) APs classified by HHMM/SVM Model

Fig. 10 Confusion matrix of inferring activities at the high level and APs at the intermediate level by HHMM/SVM hybrid model

9.1 *Missing Data*

One of the challenges in dealing with real-time application such as ours, is dealing with missing data. Data can be missing or inexact due to various factors such as erroneous/faulty instrument/sensor measuring, missing attributes from one or more sensor. The discriminative nature of the SVM classifier, makes it less capable of handling *missing data*. On the contrary, HHMM being a generative model is more able of learning in the presence of missing values, and often performs better when training set sizes are small (Raina et al., 2004). This is mainly due to the EM learning methodology which optimizes the model over the whole dimensionality, and thus models all the relationships between the variables in a more equal manner (Le and Bengio, 2002).

In order to emulate a case of missing data and smaller training data set, we conducted experiment by randomly removing data samples from the training data. We divided the entire data set into two equal half for training and testing as we did for the HHMM experiments specified in Section 7. The training data set was down sized further by randomly sampling data at a frequency of *1/2 Hz, 1 Hz, 3 Hz, 5 Hz & 7 Hz*. By generating random data sets using this method, the information related to a given activity or AP lost by down sampling can be regarded as representing missing/lost data. Note that the random sampling of data is done such that there will be at least one sample which represents an AP in any given activity sequence, so the down sample rates are approximate. This is done so as to maintain the representation of sequence of APs in

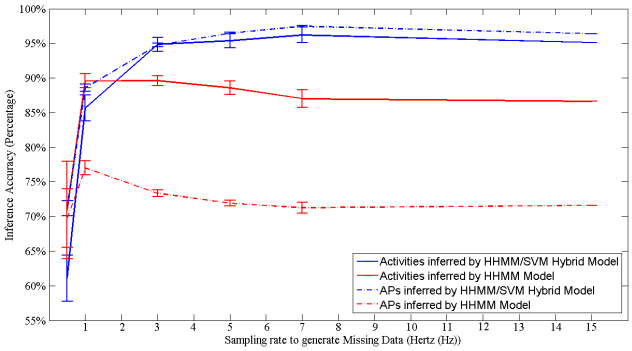


Fig. 11 Comparison of inference accuracy of HHMM and HHMM/SVM Hybrid Model when training the model with varying amount of missing data

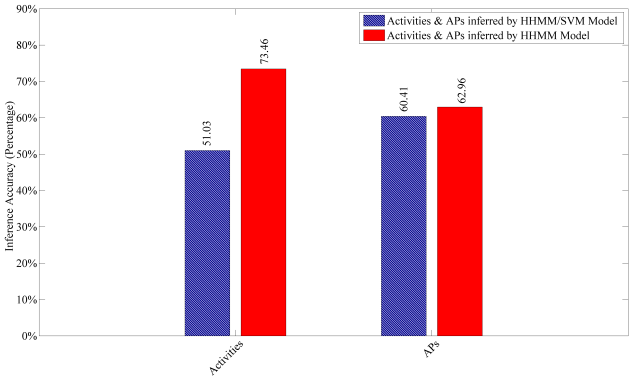


Fig. 12 Activities and APs inferred by the HHMM and HHMM/SVM hybrid model when tested with unseen data

any given activity. Further, to quantitatively analyse the impact of smaller and missing data on the performance of HHMM and HHMM/SVM hybrid model, we generated 10 random training data sets for each case, i.e. 10 different data sets for *1/2 Hz, 1 Hz* etc. Each of the trained models was then tested with a single testing data set which was sampled at *7 Hz*. Note that samples used for testing are separate, and do not overlap with any of the training data sets.

Figure 11 plots the mean and variance of the inference accuracy of the two models. It can be seen how the performance of both models decreases substantially when the amount of missing data is around 97% of the full training data at a sample rate of *1/2 Hz*. The inference accuracy of the HHMM/SVM hybrid model gradually increases as more training data becomes available. Conversely, the inference accuracy of the HHMM model remains almost constant despite the model being trained with varying amounts of training data. Hence the HHMM model seems better suited to generalise in the presence of missing data, as compared to the HHMM/SVM hybrid model.

9.2 *Testing with Unseen Activity Sequences*

To further strengthen our advocacy of HHMM model over HHMM/SVM hybrid models, we performed an experiment where we trained both models with 3 of the 4 sequences for each activities, and tested it with the unseen 4th sequence. For this experiment we used data down sampled at *7 Hz*, as the experiment in Section 9.1 showed no measurable improvement at the higher rate. As can be seen in Figure 12, the HHMM model infers the long term activities with and accuracy of 74% whereas the HHMM/SVM hybrid model inference accuracy floats around 51%. Similarly APs were inferred with an accuracy of 63% by the HHMM model and 60% by HHMM/SVM hybrid model. The HHMM model outperforms the HHMM/SVM hybrid model in inferring both the long term activities and APs, which further validates the better generalisation characteristics of the HHMM model.

9.3 Unsupervised Learning

Beyond the significant advantage of using HHMM models given their inherent generalization capabilities from smaller data sets, their unsupervised learning nature can not be under estimated. It significantly overcomes the rather difficult and costly process of obtaining labelled data for training. Moreover, unsupervised learning also opens the door to incorporate online learning algorithms whereby novelty in the patterns of performing an activity can be accomplished within the HHMM framework, e.g. using online-EM (Cappé and Moulines, 2009), a work currently under way. The modular nature of the HHMM framework thereby is better equipped for real-time addition/deletion/modification in the state space (Dindo and Schillaci, 2010), a less attractive proposition using generative models such as SVM where full re-training might be required.

10 Conclusions and Future Work

In this paper we have proposed a novel approach to infer users' manipulative activities using a HHMM probabilistic model. The HHMM framework allows to flexibly divide an activity into a hierarchy, where longer-term activities are regarded as sequential combinations of more primitive building actions, or APs. The framework was tested on a set of manipulative sequences collected for different objects used in everyday life. The hierarchical framework proved to be a powerful tool to divide activities both vertically for natural language description of different activities from APs, and horizon-

tally where the continuous observations are clustered into different APs.

We also compared the inference accuracies of the HHMM model with a HHMM/SVM hybrid model, which performs learning in a semi-supervised manner and was in general able to infer more accurately at both AP and higher activity level. The model takes full advantage of the temporal characteristics of HHMM model and strong discriminating capability of the SVM classifier to infer APs and the related long term activities. However, it was shown to be less able to generalise in the absence of rich datasets, a well-known trade-off between generative and discriminative models. Current work is investigating development of online adaptable systems within the HHMM framework. Also, while in the existing work we used data features extracted from the raw observation data to be tracked, work is in progress to apply discretisation and feature extraction techniques such as the Gaussian Process Latent Variable Model proposed in (Song et al., 2011b) to enhance the inference accuracy of the APs. Finally, we also plan to release the dataset to the research community.

11 Acknowledgement

The authors would like to acknowledge Nikolaos Kyriazis and Antonis Argyros from Institute of Computer Science, FORTH and Department of Computer Science, University of Crete, Crete, Greece for their contribution towards acquiring the data sets used in this paper.

References

Abou-Moustafa, K. T., Cheriet, M., and Suen, C. Y. (2004). Classification of time-series data using a generative/discriminative hybrid. In *Proceedings of the Ninth International Workshop on Frontiers in Handwriting Recognition*, pages 51–56.

Aksoy, E. E., Abramov, A., Dörr, J., Ning, K., Dellen, B., and Wörgötter, F. (2011). Learning the semantics of object-action relations by observation. *International Journal Robotics Research*, 30(10):1229–1249.

Bishop, C. M. and Lasserre, J. (2007). Generative or discriminative? getting the best of both worlds. In Bernardo, J. M., Bayarri, M. J., Berger, J. O., Dawid, A. P., Heckerman, D., Smith, A. F. M., and West, M., editors, *Bayesian Statistics 8*, pages 3–24. Oxford University Press.

Blimes, J. (1998). A gentle tutorial on the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. Technical Report ICSI-TR-97-021, University of Berkeley.

Cappé, O. and Moulines, E. (2009). On-line expectationmaximization algorithm for latent data models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(3):593–613.

Castellani, A., Botturi, D., Bicego, M., and Fiorini, P. (2004). Hybrid hmm/svm model for the analysis and segmentation of teleoperation tasks. In *IEEE International Conference on Robotics and Automation, 2004*, volume 3, pages 2918 – 2923.

Dillmann, R., Rogalla, O., Ehrenmann, M., Zllner, R., and Bordegoni, M. (1999). Learning robot behaviour and skills based on human demonstration and advice: The machine learning paradigm. *International Symposium on Robotics Research*, pages 229–238.

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

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

Heinze, C. (2003). *Modeling Intention Recognition for Intelligent Agent Systems*. PhD thesis, The University of Melbourne.

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

Ijspeert, A., Nakanishi, J., and Schaal, S. (2002). Movement imitation with nonlinear dynamical systems in humanoid robots. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1398–1403.

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

Jensen, F. V. (1996). *An Introduciton to Bayesian Networks*. UCL Press.

Kawanaka, D., Okatani, T., and Deguchi, K. (2005). Hierarchical-hmm based recognition of human activity. *Proc of Machine Vision Applications*.

Khansari-Zadeh, S. and Billard, A. (2010). 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.

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

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

Kulic, D., Kragic, D., and Krüger, V. (2011). Learning action primitives. In Moeslund, T. B., Hilton, A., Krger, V., and Sigal, L., editors, *Visual Analysis of Humans*, pages 333–353. Springer London.

Le, Q. and Bengio, S. (2002). Hybrid generative-discriminative models for speech and speaker recognition. Idiap-RR Idiap-RR-06-2002, IDIAP.

Liao, L. (2006). *Location-Based Activity Recognition*. PhD thesis, University of Washington.

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

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

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

Nguyen, N., Phung, D., Venkatesh, S., and Bui, H. (2005). 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.

Oikonomidis, I., Kyriazis, N., and Argyros, A. (2011a). 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.

Oikonomidis, I., Kyriazis, N., and Argyros, A. (2011b). 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.

Papazov, C. and Burschka, D. (2011). An efficient ransac for 3d object recognition in noisy and occluded scenes. *Computer Vision*, pages 135–148.

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

Patel, M., Miró, J. V., and Dissanayake, G. (2012). 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.

Raina, R., Shen, Y., Ng, A. Y., and McCallum, A. (2004). Classification with hybrid generative/discriminative models. In Thrun, S., Saul, L., and Schölkopf, B., editors, *Advances in Neural Information Processing Systems*. MIT Press, Cambridge, MA.

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

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

Schaal, S., Peters, J., Nakanishi, J., and Ijspeert, A. (2004). Learning movement primitives. In *International Symposium on Robotics Research*. Springer.

Song, D., Ek, C. H., Huebner, K., and Kragic, D. (2011a). Embodiment-specific representation of robot grasping using graphical models and latent-space discretization. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 980 –986.

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

Stadermann, J. and Rigoll, G. (2004). A hybrid svm/hmm acoustic modeling approach to automatic speech recognition. In *INTERSPEECH*. ISCA.

Valstar, M. F. and Pantic, M. (2007). Combined support vector machines and hidden markov models for modeling facial action temporal dynamics. In *IEEE international conference on Human-computer interaction*, HCI'07, pages 118–127, Berlin, Heidelberg. Springer-Verlag.