

# Choosing Informative Actions for Manipulation Tasks

Shiraj Sen and Grant Sherrick and Dirk Ruiken and Rod Grupen

University of Massachusetts Amherst

Computer Science Department

140 Governors Dr.

Amherst, MA 01003

{shiraj, sherrick, ruiken, grupen}@cs.umass.edu

**Abstract**—Autonomous robots demand complex behavior to perform tasks in unstructured environments. In order to meet these expectations efficiently, it is necessary to organize knowledge of past interactions with the world in order to facilitate future tasks. With this goal in mind, we present a knowledge representation that makes explicit the invariant spatial relationships between sensorimotor features comprising a rigid body and uses them to reason about other tasks and run-time contexts.

## I. INTRODUCTION

Intelligent agents embedded in a dynamic, uncertain environment demand dexterous behavior. To meet these expectations, one needs to address a set of problems associated with real-world environments: knowledge of the environment is incomplete and approximate; sensing is limited and noisy; the dynamics of the environment are only partially predictable. Classical AI planning approaches [1] are inadequate in this setting, especially with respect to realtime decision-making and control. On the other hand, purely reactive control systems (e.g., [2], [3]), while responsive to environmental stimuli, suffer from limited access to global state information. Planning, learning and control need to be essential components of any robot system performing dexterous behavior, yet they have long been studied in isolation. This is due to the lack of a common representational framework that allows the robot to autonomously interact with the world in a manner that supports all three. In this paper, we present a solution to this representational issue by using a functional model of interactions with the world. The robot acquires control behavior, uses the acquired skills as the representational basis for control knowledge, and then builds plans in this space of control actions.

Actions in this framework are closed-loop controllers constructed from combinations of sensors, effectors, and potential functions. Programs (schemas) are written using combinations of actions. In prior work, we demonstrated how schemas are automatically acquired using intrinsically motivated reinforcement learning [4]. In this paper, we show how objects in the environment can be modeled as spatial and temporal distribution over actions. We then present an information theoretic algorithm for task-driven action selection wherein models support choosing actions that produce the greatest reduction in uncertainty.

Section II describes the mathematical framework for representing sensorimotor programs (and compositions of them) in terms of schema. Section III shows how these control programs can be used by the robot to organize knowledge about objects in the world. An algorithm that uses these models to select informative actions is then presented. Section IV presents experimental results utilizing the described approach for manipulation tasks.

### A. Related Work

Representing knowledge about the world in terms of controllable interactions provides a powerful and computationally efficient way for an agent to encode its experiences. Psychologist J. J. Gibson introduced the term affordance [5] as all *action possibilities* latent in the environment, objectively measurable in relation to the actor and therefore dependent on their capabilities. He presented an interactionist view of perception and action that focussed on the information that is available in the environment. The use of affordances within autonomous robotics is mostly confined to behavior-based control, and their use in deliberation remains a largely unexplored area. Since the formulation of the theory of affordances, a great deal of work has been done to formalize this concept in a manner that can be modeled computationally. Specifically, Stoytchev [6], [7] and Fitzpatrick [8] showed that affordance learning can be used to differentiate objects in the course of interaction with the environment. Stoytchev's and Fitzpatrick *et al.*'s work uses affordance as a higher level concept, which a developing cognitive agent learns about by interacting with objects in the environment. Montesano *et al.* [9] presented an affordance based model based on Bayesian networks that linked actions and their effects to object features.

Modeling the stable perceptual features and relationships between features of rigid body objects is a long-standing problem in computer vision, robotics, and artificial intelligence. In computer vision, there have been enhancements to both place recognition [10], [11] and scene segmentation [12] by modeling the spatial relationships between objects. Specific to object pose estimation, Maji and Malik [13] developed a probabilistically weighted Hough transform in which parts of an object cast votes for the object's pose weighted by the likelihood of each feature appearing at a specific position on

that object. In a similar fashion, [14] investigated using easy-to-find objects as contextual priors for finding items that are more difficult to locate. These studies relate to our work in that we will be using evidence and knowledge about relationships between perceptual features, object identity, and object pose to increase or decrease beliefs about object properties and action goals.

Uncertainty is a key issue when determining object and action parameters. Ek *et al.* [15] presented a system which is able to infer the commanded task and reason about action selection given information derived from partial observations. In this work, an optimal perceptual action is defined to be the action that will maximally disambiguate (reduce entropy over) the state-space. In [16], Dragiev *et al.* utilize Gaussian processes to dilate expectation for object pose in the context of reaching and grasping tasks. Recently, Petrovskaya *et al.* [17] presented the Guaranteed Recursive Adaptive Bounding (GRAB) algorithm for efficient approximate inference, which was tested in the context of a manipulation environment by accurately localizing an object’s pose from a set of relative sensor measurements. Similar to this work, [18] developed a decision theoretic framework for task-driven exploration with POMDPs in which their system iteratively minimizes uncertainty in object pose by probing an object. In contrast to these systems, we propose a system that suggests new actions based upon the expected decrease in uncertainty with respect to a task, represented by the successful completion of another action. This formulation is able to exploit past interactions from multiple objects, environments, and tasks, as well as, reason about the predicted effects of selected actions.

## II. CONTROL PROGRAMS - SEARCHTRACK

Primitive control actions,  $c \equiv \phi_\tau^\sigma$  are closed-loop feedback controllers constructed by combining potential functions,  $\phi$ , with feedback signals,  $\sigma$ , and motor resources,  $\tau$ . The sensitivity of the output of the potential function to changes in the motor variables provides a control gradient that is used to derive reference motor inputs ( $\mathbf{u}_\tau$ ). Events in the error dynamics of each controller provide a natural discrete abstraction of the underlying continuous state space [19]. In this work, we employ a four-valued control state,  $p(c) \in \{X, -, 0, 1\}$ , where ‘X’ indicates unknown control state, ‘-’ indicates that the reference signal is not available, ‘0’ indicates the transient control response and ‘1’ denotes convergence/quiescence. Given a collection of  $n$  distinct primitive control actions, a discrete state space  $\mathcal{S}$  can be formed, where  $s \in \mathcal{S}$  is defined by  $s = (p_1, \dots, p_n)$ .

There are two distinct types of actions that share potential functions and effector resources, but are distinguished by the source of their input signals : TRACK and SEARCH. TRACK actions,  $\phi_\tau^\sigma$  preserve a reference value in the feedback signal e.g., the position of a feature on the image plane or the value of a contact force on a fingertip. SEARCH actions are of the form  $\phi_\tau^{\tilde{\sigma}}$  (sharing potential functions and effector resources with their TRACK counterparts). Their input,  $\tilde{\sigma}$ , is derived from probabilistic models describing distributions over effector

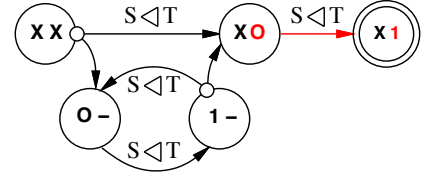


Fig. 1. SEARCHTRACK behavior in terms of state  $[p^{search} \ p^{track}]$ . A new SEARCH goal is sampled whenever SEARCH is executed from states for which  $p^{search} \in \{X, 1\}$  (designated by small circles).

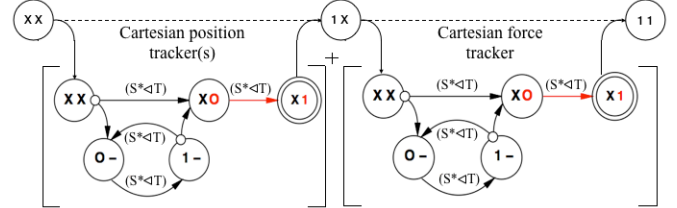


Fig. 2. Sequential programs can be learned by sequencing a set of previously learned SEARCHTRACK schemas. The robot learns how to “grasp” by sequencing two different SEARCHTRACK schemas that establishes spatial features in  $SE(3)$  followed by invariants in the force/moment domain associated with prehensile behavior. The ‘+’ sign for the first schema indicates that the robot might need to track multiple different spatial features before it can reliably track a force.

reference inputs ( $\mathbf{u}_\tau$ ), where rewarding TRACKING actions have been discovered in the past,  $p(\phi_\tau^\sigma) = 1$ . For example, such a controller can be used to direct the field of view of a robotic system to look at places on a table top where a coffee cup has previously been found. Initially the distribution  $Pr(\mathbf{u}_\tau | p(\phi_\tau^\sigma) = 1)$  is uniform; however, over the course of many learning episodes, this distribution reflects the long term statistics of the run-time environment.

In Figure 1, an abstract SEARCHTRACK control action (schema) is illustrated. Action  $S \triangleleft T$  is a concurrent combination of SEARCH and TRACK actions, where SEARCH is executed in the nullspace of TRACK [20]. The policy begins by attempting to concurrently SEARCH for and TRACK a specific cue. If this cue exists in the signal, the policy attempts to continue TRACK-ing. If no target is immediately available, the policy samples new configurations from the search distribution  $Pr(\mathbf{u}_\tau | p(\phi_\tau^\sigma) = 1)$  until the target stimulus is found; at which point, the policy **tracks** the feature.

Hart *et al.* [21] presented a detailed description of the various manipulation programs (touching, grasping, picking up, placing and inspecting objects) that can be learned in a hierarchical fashion from previously acquired control programs. All of these programs can be viewed abstractly as a sequence of SEARCHTRACK programs that are used to find and track groups of visual and tactile features. Figure 2 shows a hierarchical schema learned by the robot to reliably track a reference force using its end effector. The learned program involves concurrently tracking multiple visual stimuli (indicated by the ‘+’ symbol for the first schema) followed by a SEARCHTRACK schema that tracks forces using fingertip mounted tactile sensors. In this hierarchical schema, the

Cartesian feature position tracker becomes part of the **search** behavior that orients the robot to receive a TRACK-able force.

Our computational framework acquires programs for controlling interaction with the environment and manages redundant sensory and motor resources to discover and maintain interactions in dynamic environments. The acquired control programs and their long term statistics represent a domain general way of interacting with stimuli in the environment. The schemas capture *common sense* knowledge acquired by the robot. The environment, however, presents important kinds of structure in terms of *objects* — sets of affordances with invariant spatial relationships (co-affordances). In the next section, a Bayesian framework for acquiring these domain specific knowledge structures in terms of distributions over SEARCHTRACK programs is presented.

### III. CONTROL CO-AFFORDANCES - OBJECTS

Objects in this representation are modeled as spatial and temporal relationships between control affordances. Figure 3(a) shows a graphical model that encodes the logical dependencies between the variables of the environment affordance model. An object,  $O$ , at any instant of time affords a set of controlled interactions with the robot. This set of spatially distributed interactions define an object aspect ( $OA$ ). For each aspect, there exist  $M_j$  affordances that have a non-zero probability of occurring. There can be multiple instances of each aspect within an object. Each affordance is represented by a Bernoulli random variable  $p_j$  describing the state of each associated SEARCHTRACK action. ( $p_j = 1$ , if the action converges, and 0 otherwise.) Each possible affordance is modeled by its position and orientation ( $h_j$ ) in the object’s frame, the feature values of the signal ( $f_j$ ), and the shape ( $g_j$ ) defined by the eigenvalues of the signal. The resulting generative model describes objects in terms of affordances and the spatial relationships between them.

Utilizing past experience encoded as a prior, this model is able to aid in accomplishing a variety of tasks by telling the robot which affordances are likely to co-occur and where they occur with objects that are similar to those that have been previously encountered. However, there is no knowledge being stored that will describe how taking actions affect the existence and location of affordances. For e.g., a hammer’s handle affords grasping, however, if the handle is out of reach, the robot might have to pull the hammer closer before it can succeed in grasping it. In this case, “pulling” changes the aspect of the object in a manner that supports the goal of grasping. We encode the aspect-action dependencies of an object as a Hidden Markov Model(as shown in Figure 3(b)), where the instance of an aspect being observed is a hidden variable that the robot can infer from state of the affordances,  $s^t = \{p_1, \dots, p_k\}$  and the actions,  $a^t$ . This temporal model consists of a finite number of states (given by the aspect instances,  $OA^t$ ), a finite number of actions  $\mathcal{A} = \{a_1, a_2, \dots, a_k\}$  and a set of possible observations. For every aspect instance,  $OA^t$ , the transition probability  $T(OA^t, a^t, OA^{t+1})$  describing the probability that an aspect

---

#### Algorithm 1 TASKGOAL( $a_g, E, G$ )

---

```

1: repeat
2:   Compute all possible matches of evidence,  $E_m \subseteq E$ ,
   to model,  $G$ 
3:   Compute least squares pose estimate of the object,  $x_O \in X_O$ ,
   for every match,  $e_m \in E_m$ 
4:   Update posterior over object aspects,  $\Pr(OA | E_m)$ .
5:   Compute the position,  $x_g$ , of goal affordance,  $a_g$ , for
   each of the candidate poses  $x_O \in X_O$ .
6:   if all candidate poses vote for the same set of goals
   then
7:     if goal pose,  $x_g$ , can be achieved by the robot then
8:       execute goal,  $a_g$ 
9:     else
10:      Choose action,  $a_{next}$  from the HMM which
      changes the object aspect to one which affords the
      goal.
11:     Execute action,  $a_{next}$ 
12:   end if
13: else
14:   Choose action,  $a_{next}$  from the HMM which maxi-
   mally reduces the uncertainty over aspects.
15:   Execute action,  $a_{next}$ 
16: end if
17: Gather new evidence,  $E$ 
18: until  $p(a_g) = 1$  {Goal Action Succeeds}

```

---

at time  $t$  transitions to another aspect instance at time  $t + 1$  by taking manipulation action,  $a^t$ .

#### A. Task Specific Action Selection

Reference values for each control action can be sampled from the Bayesian model. These actions can then be executed given knowledge of the object’s pose in the world frame. However, it is not necessary for a robot to completely determine the pose of an object before it can take actions towards achieving its goal, it is only required that the pose relative to the goal action is completely determined. The procedure for taking such an action ( $a_g$ ) is described in Algorithm 1.

Given a task and an object model, the robot first accumulates evidence for all of the actions with minimum cost that are afforded by the object. The inference algorithm proceeds by finding the best possible matches,  $E_m$ , of the evidence to the model (Line 2). This is achieved by computing the likelihood that the evidence (control action with a particular feature value and shape) can be generated by the model. For each match  $e_m \in E_m$ , a least squares estimate of the object pose,  $x_O$ , is computed using the point value estimate of the affordance positions [22] (Line 3). If the same evidence is afforded in multiple regions of the object, a set of candidate poses,  $X_O$ , are returned (of which only one is the correct pose). Each of the candidate poses is combined with relative spatial information from the object model to produce candidate positions and orientations of the goal (Line 5). For example, if the goal is to grasp an object, this relative information is provided

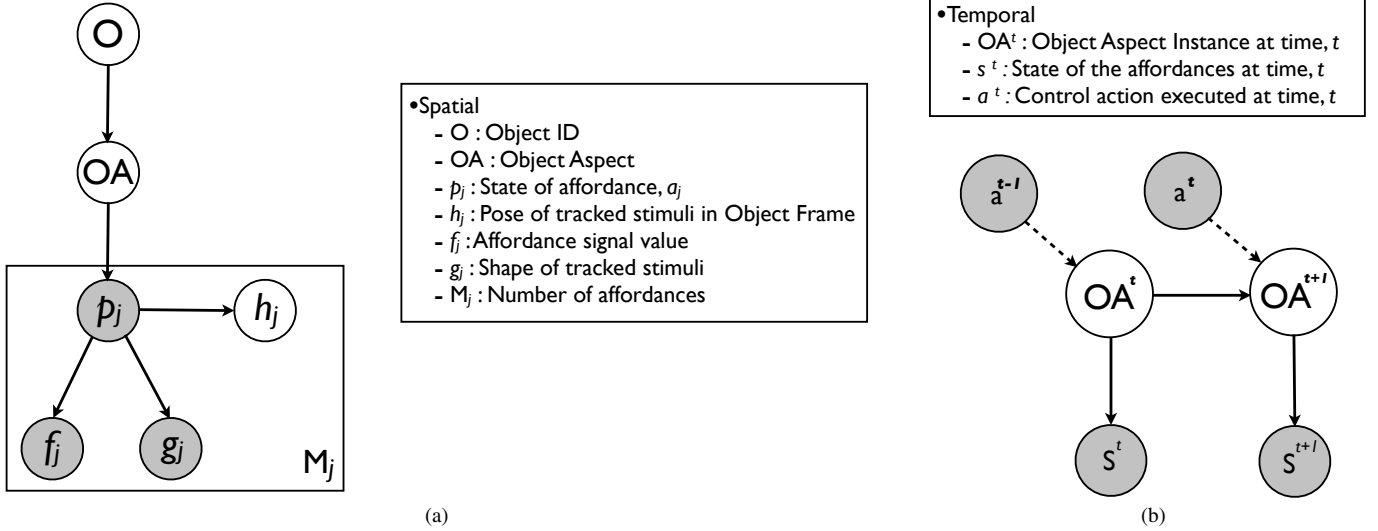


Fig. 3. Panel (a) shows a Bayesian network model  $G$  representing object  $O$  as a spatial distribution over  $M_j$  control affordances. The multinomial random variable  $OA$  represents the aspects of the object. An aspect induces a distribution over the state of controllable interactions ( $p_j$ ) afforded by the object. The random variables  $f_j$  and  $g_j$  model the signal value (e.g., color, force) and shape of the signal (eigen values) as gaussian distributions. The gaussian random variable  $h_j$  models the position and orientation of the affordance instance in the object frame. Panel (b) shows a dynamic Bayes net diagram representing graphically the relations between the different parts of the temporal model of the aspect instances,  $OA^t$  to state of the affordances,  $s^t$  and manipulation action,  $a^t$ . (Shaded nodes are observed, clear nodes are hidden).

by SEARCH distributions for the arm(s) and hand(s) relative to the object frame that orient the system appropriately for TRACK-able forces comprising a grasp. Each candidate pose is modeled as a mixture of gaussians. If all the candidate poses vote probabilistically for the same positions and orientations of the goal in the world frame (Line 6), we say that the candidate goals are equivalent and unambiguous. Hence, even though there is uncertainty in the pose of the object, there is none in the goal affordance. Such cases arise in the case of symmetric objects, where the pose of the object may remain ambiguous even when grasping goals are not. If goals are ambiguous, then actions are selected that reduce uncertainty with respect to the target action to the greatest degree. This is done by using the temporal model as a forward model for predicting the next action to execute that maximally reduces the uncertainty over aspects. This is achieved by computing the posterior over the hidden aspect instances,  $OA$  (Line 4), and then using the transition probabilities,  $T$  for selecting the action  $a \in \mathcal{A}$  that will maximally reduce the entropy over the aspect posterior (Line 14). Furthermore, the above model is also used for planning sequences of actions that can change the aspect of the object to one that affords the goal (e.g., for a grasping task, if the goal is out of reach, the model can predict the sequence of actions that change the aspect of the object to one where a grasp can succeed). This process is repeated until the robot believes that there is no information to be gained by taking more actions in the context of the given task ( $a_g$ ) or, until the recommended course of action is sufficiently certain.

#### IV. EXPERIMENTS

We demonstrate the applications of the above approach on our experimental platform, Dexter. Dexter is a bimanual robot with two 7-DOF Whole-Arm Manipulators (WAMs) from Barrett Technologies, two 3-finger 4-DOF Barrett Hands equipped with one 6-axis force/torque load cell sensor on each fingertip, a stereo camera pair and a Kinect mounted on a pan/tilt head.

In this set of experiments, we show the efficacy of our representation for utility-driven action selection. Models of objects were hand-built spatial distributions of blobs (represented in terms of first and second moments) describing homogeneous hues, range image blobs, and search distributions of hand goals in Cartesian space where “grasp” and “touch” affordances can be found. Grasp affordances are defined by TRACK-able force closure conditions and touch affordances are defined by small magnitude TRACK-able force events. For purposes of illustration, the middle of the mallet’s handle and the middle of the emergency light were set to afford grasping while their entire body affords touching. The temporal part of the object model captures the transitions between aspects when a manipulation action is executed in the context of each object. 10 trials were conducted for each object, in which the object was placed in the workspace in a variety of poses. In certain regions of the workspace, the object does not afford haptic aspects, and additional manipulation actions have to be taken before grasp goals can be achieved. Figure 4 and 6 shows the case when the object is presented in a region where the robot can grasp successfully. In such a case, the Algorithm 1 computes the pose of the object by matching observations to the model and returns the estimated goal pose. However,

when the goal affordance is out of reach (and hence the object aspect doesn't afford the goal - grasping in this case), the action selection algorithm chooses a manipulation action that can change the aspect to one that affords grasping. Figure 5 and 7 shows the two scenarios where the robot chooses to pull the object towards itself before executing the grasp action.

## V. CONCLUSIONS

In this paper, we introduced a knowledge representation framework that organizes knowledge about objects in terms of long term statistics of controllable interactions. We presented how objects in the environment can be modeled as temporal distributions over spatially related co-affordances. Lastly, we provided experimental results of using this approach for modeling objects and using these models for utility-driven action selection. The above representation of objects as spatially structured control affordances provides a powerful mechanism for action planning. In future work, we plan to extend this technique to multi-object relationships.

## ACKNOWLEDGMENT

This work was supported by the DARPA grant iRobot Corp. Prime Army W91CRB-10-C-0127.

## REFERENCES

- [1] R. E. Fikes and N. Nilsson, "Strips: A new approach to the application of theorem proving to problem solving," *Artificial Intelligence*, vol. 5, no. 2, pp. 189–208, 1971.
- [2] R. Brooks, "A robust layered control system for mobile robot," *IEEE Journal of Robotic Automation*, vol. 2, no. 1, pp. 14–23, 1986.
- [3] R. J. Firby, "An investigation into reactive planning in complex domains," in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 1987.
- [4] S. Hart, "The development of hierarchical knowledge in robot systems," Ph.D. dissertation, Department of Computer Science, University of Massachusetts Amherst, 2009.
- [5] J. Gibson, "The theory of affordances," in *Perceiving, acting and knowing: toward an ecological psychology*. Hillsdale, NJ: Lawrence Erlbaum Associates Publishers, 1977, pp. 67–82.
- [6] A. Stoytchev, "Toward learning the binding affordances of objects: A behavior-grounded approach," in *Proceedings of the AAAI Spring Symposium on Developmental Robotics*, Stanford University, 2005.
- [7] —, "Behavior-grounded representation of tool affordances," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain, 2005.
- [8] P. Fitzpatrick, G. Metta, L. Natale, S. Rao, and G. Sandini, "Learning about objects through action: Initial steps towards artificial cognition," in *IEEE International Conference on Robotics and Automation*, Taipei, May 2003.
- [9] L. Montesano, M. Lopes, A. Bernardino, and J. Santos-Victor, "Modeling affordances using bayesian networks," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, San Diego, CA, 2007.
- [10] E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky, "Depth from familiar objects: A hierarchical model for 3d scenes," in *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2*, ser. CVPR '06. Washington, DC, USA: IEEE Computer Society, 2006, pp. 2410–2417.
- [11] A. Ranganathan and F. Dellaert, "Semantic modeling of places using objects," in *Proceedings of Robotics: Science and Systems*, Atlanta, GA, USA, June 2007.
- [12] S. Gould, J. Rodgers, D. Cohen, G. Elidan, and D. Koller, "Multi-class segmentation with relative location prior," *Int. J. Comput. Vision*, vol. 80, pp. 300–316, December 2008.
- [13] S. Maji and J. Malik, "Object detection using a max-margin hough transform," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, june 2009, pp. 1038 –1045.
- [14] J. Vogel and K. Murphy, "A non-myopic approach to visual search," in *Proceedings of the Fourth Canadian Conference on Computer and Robot Vision*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 227–234.
- [15] C. Ek, D. Song, K. Huebner, and D. Kragic, "Task modeling in imitation learning using latent variable models," in *Proceedings of 10th IEEE-RAS International Conference on Humanoid Robots*, 2010.
- [16] S. Dragiev, M. Toussaint, and M. Gienger, "Gaussian process implicit surfaces for shape estimation and fluent grasping," in *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE, 2011.
- [17] A. Petrovskaya, S. Thrun, D. Koller, and O. Khatib, "Towards dependable perception: Guaranteed inference for global localization," Dependable Robots in Human Environments Workshop at Robotics: Science and Systems, June 2010.
- [18] K. Hsiao, L. Kaelbling, and T. Lozano-Perez, "Task-driven tactile exploration," in *Proceedings of Robotics: Science and Systems*, Zaragoza, Spain, June 2010.
- [19] J. Coelho and R. Grupen, "A control basis for learning multifingered grasps," *Journal of Robotic Systems*, vol. 14, no. 7, pp. 545–557, 1997.
- [20] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task-priority based redundancy control of robot manipulators," *Int. J. Rob. Res.*, vol. 6, no. 2, pp. 3–15, 1987.
- [21] S. Hart, S. Sen, and R. Grupen, "Intrinsically motivated hierarchical manipulation," in *Proceedings of 2008 IEEE Conference on Robotics and Automation*, Pasadena, CA, 2008.
- [22] R. M. Haralick, H. Joo, C. Lee, X. Zhuang, V. G. Vaidya, and M. B. Kim, "Pose estimation from corresponding point data," *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics*, vol. 19, no. 6, 1989.





Fig. 4. The robot performing a top grasp on the object and placing it on the goal.

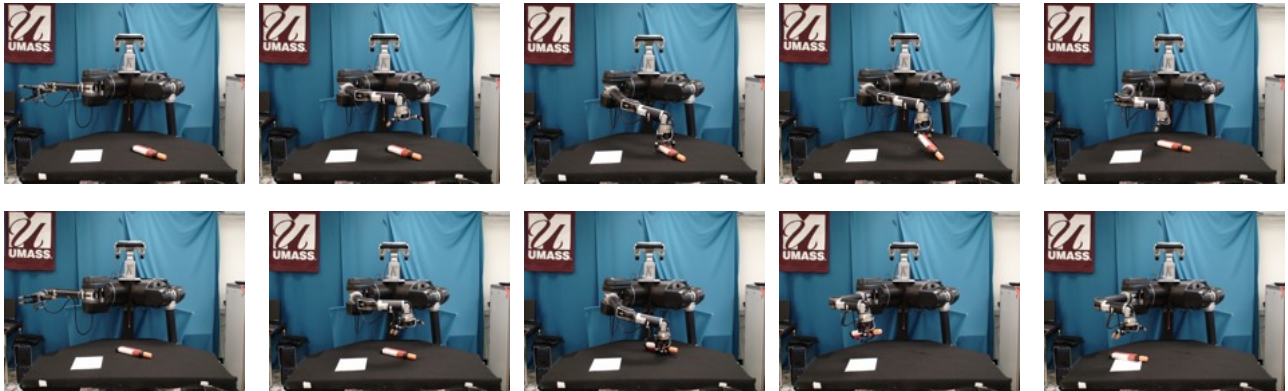


Fig. 5. The robot pulling the object towards itself before performing a re-grasp on the object and placing it on the goal.



Fig. 6. The robot performing a top grasp on the mallet and placing it on the goal.

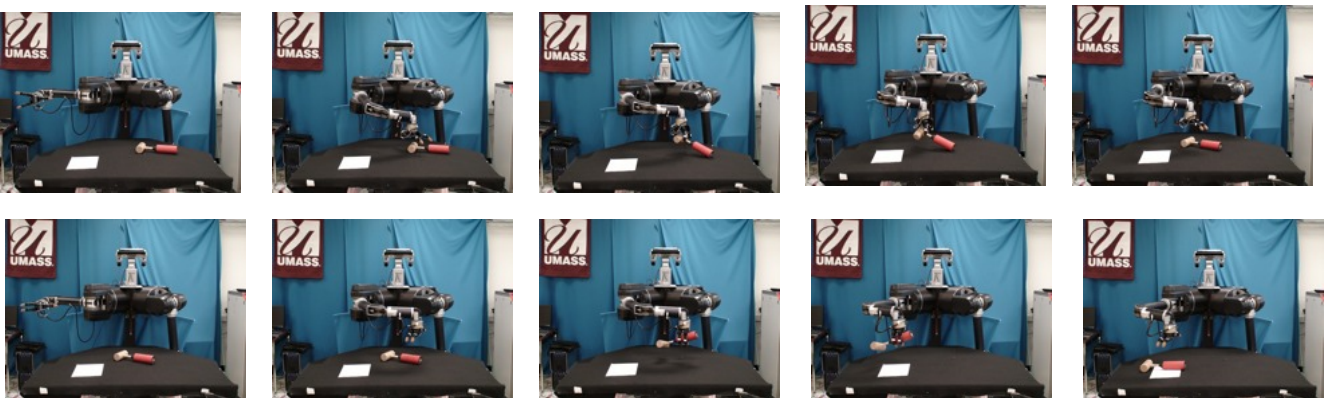


Fig. 7. The robot pulling the mallet towards itself before performing a top grasp on the object and placing it on the goal.