Learning Haptic Affordances from Demonstration and Human-Guided Exploration

Vivian Chu[1] and Baris Akgun[1] and Andrea L. Thomaz[1]

***Abstract*— In this paper, we present a system for learning haptic affordance models of complex manipulation skills. The goal of a haptic affordance model is to better complete tasks by characterizing what a particular object-action pair *feels* like. We use learning from demonstration to provide the robot with an example of a successful interaction with a given object. We then use environmental scaffolding to collect grounded examples (successes and unsuccessful "near misses") of the haptic data for the object-action pair, using a force/torque (F/T) sensor mounted at the wrist. From this we build one success Hidden Markov Model (HMM) and one "near-miss" HMM for each object-action pair. We evaluate this approach with five different actions over seven different objects to learn two specific affordances (open-able and scoop-able). We show that by building a library of object-action pairs for each affordance, we can successfully monitor a trajectory of haptic data to determine if the robot finds an affordance. This is supported through cross-validation on the object-action pairs, with four of the seven object-action pairs achieving a perfect F_1 score, and with leave-one-object-out testing, where the learned object-actions models correctly identify the specific affordance with an average accuracy of 67% for scoop-able and 53% for open-able.**

I. Introduction

Robots are moving from structured environments (e.g. factories) to unstructured human environments (e.g. hospitals and homes). Our work focuses on learning affordance models for the dexterous manipulation skills required by these robots. The term *Affordance* was coined by psychologist J.J. Gibson, referring to the "action possibilities" between the environment and the agent [1]. Affordances are a representational choice to model skills as the relationship between effects and a set of actions performed by an agent on an object. For example, pulling (action) a drawer handle *and* causing the drawer to slide out (effect) would be the affordance open-able. This representation (and other versions) is commonly used in affordance learning in robotics [2, 3]. Affordances enable us to communicate object properties and tasks to the robot (e.g. to water a plant, a robot can reason that it needs an object with a contain-able and pour-able affordance).

In this paper, we investigate: (1) learning the object-action pairs for a specific affordance, and (2) using these learned models to test an object for that affordance. We define skills as a low-level trajectory that achieves a specific goal and model each skill as an object-action pair. Affordances consist of several object-action pairs, detailed in Section III.

While most prior work in affordance learning focuses on *visual* affordances, our work addresses haptic models

*This research is supported by NSF CAREER grant IIS-1032254

[1] School of Interactive Computing, Georgia Institute of Technology, Atlanta, Georgia 30332-0250 Email: {vchu, bakgun}@gatech.edu, athomaz@cc.gatech.edu

Fig. 1: Our experimental platform, Curi, with the various objects it learns affordances for in this paper.

of affordances – what successful and unsuccessful interactions of a object-action pair *feel like*. We perceive these affordances using force/torque (F/T) sensors. The haptic model of an object-action pair is complementary to visual affordances. While both require acting on the object to learn an affordance, a learned visual affordance can be used to select "action possibilities" *prior* to interacting with the object; whereas haptic models can only provide information on possibilities *during* the interaction. Furthermore, some affordances are visually difficult to detect, but are salient through force sensing (e.g. push/pull door handle). Together, they can provide a richer set of possibilities for the robot to find and utilize. This work focuses on modeling affordances with just F/T sensing as opposed to using F/T sensing as an additional channel in a multimodal feature vector. Understanding the role of haptics alone is a key step along the way to an integrated multimodal affordance model.

To model haptic affordances, the robot must successfully execute actions on objects. Our work explores an approach that leverages a human teacher to assist the robot in rapidly exploring a variety of objects to learn haptic affordances. This system uses learning from demonstration (LfD) to acquire primitives for exploration, and environmental scaffolding to guide the robot's exploration. We use this system to perform 5 different actions over 7 different objects to build object-action models for the haptic affordances of "open-able" and "scoop-able". We show that the learned object-action models achieve good cross-validation performance with 4 of the 7 object-action pairs achieving a perfect F_1 score. Also, by leveraging the set of object-action models per affordance, we perform leave-one-object-out testing to identify affordances with an average accuracy of 67% for scoop-able and 53% for open-able, with haptic sensing alone.

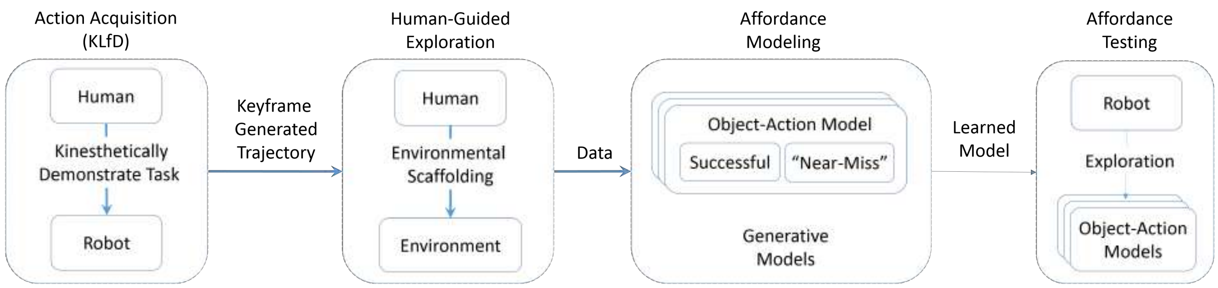


Fig. 2: Components and information flow of the system. Action Acquisition builds an action trajectory from human demonstration. In Human-Guided Exploration, environmental scaffolding yields successful and "near-miss" interactions. This data is then used during Affordance Modeling to build a set of generative object-action models where in Affordance Testing, they are used to determine if an object has an affordance.

This paper contributes a system that (1) uses a human teacher to both rapidly acquire actions *and* explore objects for learning affordances and (2) uses *only* haptic sensing to identify multiple affordances on unseen objects.

II. Related Work

To the best of our knowledge, no other work learns multiple affordances from human teachers using only F/T sensing. Related areas include affordance learning with visual information, haptic modeling, and skill acquisition/monitoring.

A. Affordances

Early work in affordance learning for robotics used primitive actions to interact and learn about object effects. These established a framework for affordance learning using exploration [3]–[8]. While some of these works included proprioceptive information, none used only haptic information. All of these systems required specific primitive actions to be learned or programmed, which resulted in works focusing on a small set of possible actions (e.g. pushing) to find affordances as opposed to a wide range of actions that can be learned by seeding exploratory behaviors with LfD.

B. Haptic Modeling

While haptic affordance learning has not been specifically addressed, haptics has been used to learn and test object properties similar to how one would test an object for an affordance. Torres-Jara et al. [9] and Sinapov et al. [10] used exploratory behaviors to classify objects. Fishel and Loeb [11] identified 117 different textures based on nine primitive motions. Gemici and Saxena [12] categorized deformable foods using force sensing. Chu et al. [13] learned haptic adjectives through exploration. Bhattacharjee et al. [14, 15] used a robotic arm with a tactile sleeve to categorize contacted objects as movable or not. While these works use haptic information to obtain object properties, [13] and [12] do not directly learn how to use these properties once found and [14, 15] only learned one affordance.

C. Skill Acquisition and Monitoring

Sukhoy et al. [16] learned the trajectory for sliding a card through a card reader with proprioceptive feedback. Sturm et al. [17] learned a kinematic model for successfully opening various doors using a F/T sensor. While both only learned one primitive, they show that haptics could be used to improve actions directly through experience. Pastor et al. [18] learned and predicted the outcome of complex skills (e.g. flipping a box with chopsticks) by using pressure sensors and reinforcement learning on dynamic motion primitives. They extended [18] and introduced Associative Skill Memories [19] where they learned haptic feedback from demonstration of actions on objects. However, they did not learn and discover haptic affordances or properties of an object. Recently, researchers have begun exploring how to model force and compliant dependent skills using LfD [20, 21]. The focus, however, is on executing the specific skill and not on using trajectories to explore the environment.

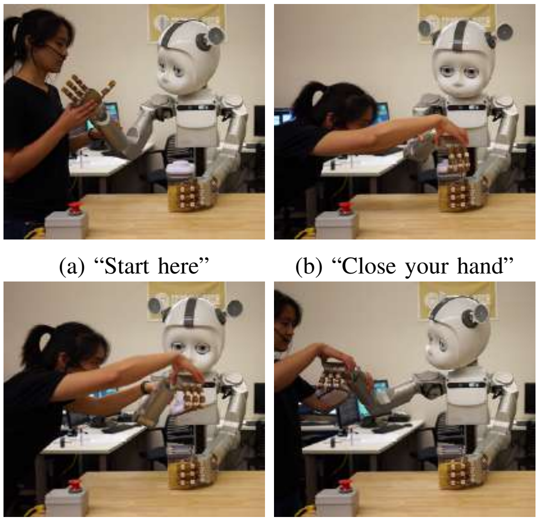
III. Approach: Learning Haptic Affordances

To learn haptic affordances, a robot must successfully interact with objects and build a model of the interaction. Our approach has four components, also shown in Fig. 2.

(1) Action Acquisition: We use LfD to show the robot an exploration action to perform (Fig. 3). LfD for affordance learning is one novelty of our work and allows us to quickly program several primitive actions. Most prior work use one or two simple primitives (e.g. push is popular), whereas here we have five primitive actions with a range of complexity.

(2) Human-Guided Exploration: Next, the robot repeats the demonstrated action several times. For each interaction, the human moves the object to perturb the action context slightly (Fig. 4c). This is a teaching interaction known as environmental scaffolding [22]. In similar prior work [8], human-guided exploration yielded high-quality learning examples that provided focus for exploration within a very large search space. An alternative to scaffolding is to utilize self-exploration [23] and this is an area of recent work [24].

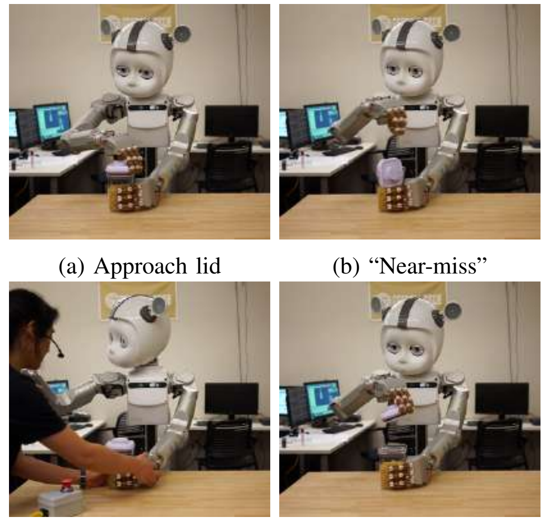
(3) Affordance Modeling: Each interaction during the exploration phase generates a continuous trajectory from a F/T sensor at the wrist. We use these trajectories to build an HMM (Section VII) of the haptic **effect** of this object-action pair. We build two HMM models for a given object-action pair, one HMM from examples of successful interactions and another HMM from examples where the object-action execution failed to find the affordance. In our human-guided exploration, we assume unsuccessful interactions are



(a) "Start here" (b) "Close your hand"

(c) "Go here" (d) "End here"

Fig. 3: Kinesthetic LfD for the open-able affordance on the pasta jar using keyframes. Keyframes are recorded using verbal commands listed below each image.



(a) Approach lid (b) "Near-miss"

(c) Change environment (d) Success

Fig. 4: Curi executing demonstrated trajectory on the pasta jar. (b) Curi misses the first time, (c) a person adjusts the object, and (d) Curi succeeds

informative "near-misses" of the action, hence this model is characterizing what it feels like when this object-action pair does not find the desired effect. Importantly, this is not a model of all failures, which would be a huge class, but of the much smaller and likely more informative class of boundary case failures that are close in action space to success [25, 26] (e.g. a lid slipping from the hand when lifting seen in Fig. 4b). Furthermore, modeling "near-misses" can provide knowledge to detect when a trajectory begins to deviate from success and adapt in real-time (future work). It is unclear how this can be done with a single model of all possible failures.

These object-action pairs, each containing a success HMM and "near-miss" HMM, provide several specific examples of a single affordance. This representation allows the robot to not only learn **what** an affordance feels like, but also provides a library of actions for **different ways** in which the affordance has been achieved. For instance, there are multiple ways for an object to be open-able (e.g. open a drawer by pulling, open a jar by twisting the top) and by modeling each of these methods, the robot now has access to a library of actions to explore a new object to find the affordance.

(4) Affordance Testing: The learned models are used to test for an affordance by monitoring the effect of the execution of each object-action pair for that affordance. A benefit of modeling both success and "near-miss" is that the decision per object-action pair can be made by relative likelihood between these two models. Given that we have multiple object-action pairs per affordance, the robot can use any/all combinations of object-action pairs previously learned (e.g. comparing the log-likelihood from all HMMs) to determine if an object has an affordance.

The remainder of the paper, following the details of our robot platform, is focused on our implementation and validation of each of the four main components mentioned above. Videos of the system and experiments can be found at `https://www.youtube.com/user/SimonTheSocialRobot`.

IV. Hardware Platform

Our robot platform, called "Curi", has two 7 degree-of-freedom (DOF) arms, each with an under-actuated 4-DOF hand, and an ATI Mini40 force/torque (F/T) sensor attached to each wrist. The experiments in this paper utilize just the left arm, which is currently the only fully-functional arm. Curi has a movable torso mounted on a mobile base. In this work, Curi is positioned in front of a table with an object on it (Fig. 1). Above the table is a ASUS Xtion Pro RGB-D sensor. We segment the objects on the table using the point cloud data. The object's pose (position and orientation) and bounding box are recorded.[1]

V. LfD Action Acquisition

In our approach, a teacher provides a demonstration of an action by physically guiding the robot to perform it (as opposed to observations of a human performing the action). This highlights a key point that affordances are "action possibilities" that occur between the *agent* and the environment. There are likely many objects that have affordances for a person that our robot would be unable to achieve (e.g. jar is closed too tightly for the robot to open). Particularly for haptic affordances, it is essential that the robot successfully explore the environment to learn what the effects of particular object-action pair *feel like to the robot*.

We use a keyframe-based LfD approach [27], whereby a teacher demonstrates each action by physically guiding Curi and marking salient points of the action (Fig. 3). During these points, snapshots of the joint states are stored as keyframes (KFs). To replay a demonstration, the KFs are

[1]While the visual data is not used in this work, we record it to allow for future integration with systems using visual affordance learning.

TABLE I: Affordances

Object	Action	Effect	Affordance
Cup 1	Scoop	Macaroni in container	scoop-able
Cup 2	Scoop	Macaroni in container	scoop-able
Parmesan Bottle	Scoop	Macaroni in container	scoop-able
Pasta Jar	Lift	Cap Lifts	open-able
Drawer	Pull	Drawer slides	open-able
Wooden Box	Push1	Lid opens	open-able
Bread Box	Push2	Lid opens	open-able

splined together into a single trajectory using a quintic spline at an average velocity of 0.1 radians/second. The velocity was pre-selected to execute smoothly on Curi and applies for all actions. Curi executes the trajectory autonomously on the object during playback (Fig. 4). This guarantees no external F/Ts are felt during data recording.

We taught Curi two affordances (**scoop-able** and **open-able**), which results in five separate actions to interact with seven different objects. The actions range in complexity starting with simple actions that can be easily executed by the robot (e.g. pushing) to more realistic actions on objects that can be found in homes (e.g. scooping pasta, opening drawers). The first action is on the objects shown in Fig. 5, where the same scooping motion is repeated using three different, but similar objects. We then increase the difficulty of the task by selecting an affordance that require four different actions over varying objects. The set of objects shown in Fig. 6 are all open-able, but require different low-level actions. The object-action pairs are listed in Table I. To increase the stability of some of the lighter objects during interaction, Curi's non-functional right arm was propped up and the weight of the arm prevented the objects from sliding. This was done for the objects Pasta Jar, Wooden Box, and the bowl of macaroni during all of the scoop actions.

VI. Human-Guided Exploration

Next we collect a dataset of haptic information during object-action interactions for each of the seven object-action pairs described in Table I. Each action was executed by the robot 20 times on the same object, such that 10 interactions successfully found the affordance on the object and 10 were unsuccessful "near-misses". This was done by moving the object around by the human[2] and hand labeling when the interaction did or did not find the affordance. As "near-misses" occur naturally when executing a skill, overall extra interactions were not necessary to achieve an even split of successful and "near-miss" examples. For the open-able affordance, "near-misses" often included interactions where Curi missed the handle or lid of the object. For the object-action pairs in the scoop-able affordance, "near-misses" were instances where the cup dragged along the macaroni, but did not get any macaroni in the cup. During each interaction, the robot records: F/T data from the wrist sensors (example in Fig. 7), object pose information, and all joint positions.

As mentioned previously, in some cases, it is visually very difficult to see when these actions find the affordance and could easily fall within the noise of a RGB-D sensor. For example, it was difficult for even the experimenter (one of the authors) to detect the change in amount of macaroni in the large bowl and in the cup. This suggests that using haptic feedback during action execution is important to fully understand the objects and object affordances.

[2]in this work, the teacher is one of the authors

Fig. 5: **Scoop-able objects**: Left-to-right - Bowl of macaroni, Cup 1, Cup 2, Parmesan Bottle

Fig. 6: **Open-able objects**: Left-to-right - Pasta Jar, Drawer, Wooden Box, Bread Box

VII. Learning Haptic Affordance Models

With the data collected, we build two haptic models (success and "near-miss") for each of the seven object-action pairs. This results in two different haptic affordance models: one for open-able and one for scoop-able.

A. Hidden Markov Models

We use Hidden Markov Models (HMMs) [28] to model the F/T information because of the time-varying nature of the data and because it provides us the ability in future work to generate expected F/T trajectories of an action by sampling from the HMMs. The HMMs are ergodic and the parameters of an n-state HMM, (A,B,π), are estimated using Expectation Maximization (EM) where A is the transition probability distribution (nxn), B the emission probability distributions (nx1), and π the initial state probability vector (nx1). We model the emission probability distribution using a continuous multivariate Gaussian distribution. Specifically, the observation state-space O is $[F_x, F_y, F_z, T_x, T_y, T_z]$ where F are the forces and T the torques. For our implementation, we used the Python library scikit-learn [29].

B. Training

We split the data randomly into a train (80%) and test (20%) set for each object-action pair and each type of model (i.e. 8 training and 2 testing interactions for both success and "near-miss"). We select the optimal number of states (between 2-6 states inclusive) for the HMMs by performing leave two-out cross-validation (CV). With 8 interactions in the training set, this results in 28 CV sets where each set has a different variation of 2 trajectories removed for testing.

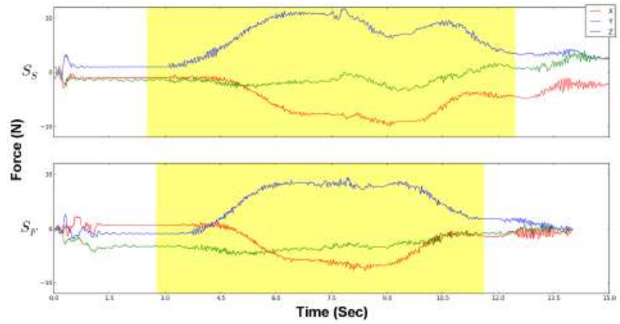


Fig. 7: An example of force data we collect from the sensor during a scooping skill. The yellow shaded portion indicates the time period when the hand is in contact with the object. The top graph is a success and the bottom a "near-miss".

C. Modeling Results

For each object-action pair, we look at whether the models can determine success versus "near-miss" for each test interaction. Per Section III, correctly monitoring the success and "near-miss" of an object-action pair allows us to test for affordances in objects. Therefore, to evaluate our models, we look at how the models perform at monitoring test interactions. We use the standard binary classification metrics of precision, recall, and F_1 such that precision $= \frac{\text{tp}}{\text{tp+fp}}$; recall $= \frac{\text{tp}}{\text{tp+fn}}$; and $F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$ where tp is the number of true positives, fp false positives, tn true negatives, and fn false negatives. We also include overall accuracy $= \frac{\text{tp+tn}}{\text{tp+fp+tn+fn}}$. The interaction is classified as successful if the log-likelihood of the successful model is greater than the log-likelihood of the unsuccessful "near-miss" model.

The resulting scores for correctly determining each object-action pair can be found in Table II. Overall each of the models perform well at determining if the test interaction within each pair was successful versus "near-miss" with four object-action pairs achieving a perfect F_1 score and another with a score of 0.75. This could be attributed to the fact that "near-misses" and successes have very unique F/T readings. For example, when lifting the lid off of the jar, the robot ended up with the weight of the lid firmly in its hand vs. having no weight at all. The two exceptions to this are scooping with the small blue cup and opening the bread box. For Cup 1, the models were overly optimistic, with all of the trajectories being classified as succeeding. This could be due to the interactions having more noise than Cup 2 and Parmesan Bottle because of the rigidness of the object. While scooping, Cup 2 and Parmesan do not deform as greatly as Cup 1. For detecting if Curi opened the Bread Box, the models were overly pessimistic with none of the test trajectories being classified as successfully opening the box. We believe that this is because "near-misses" often still pushed on the object and the sensory readings look similar to pushing on the handle successfully.

VIII. Affordance Testing

Our final experiment is a case study of how well existing object-action pairs can classify other object-action pairs within the same affordance. Note, this is different than testing for an affordance on a new object as we do not execute any of the existing action trajectories on different objects. This is a topic of future work due to the difficulty of adapting an existing trajectory to a new object (a task that on its own is a current active research area [30]–[32]). We present this section to show the limits of using only the previously built object-action pairs to generalize to other interactions from existing object-action pairs. More specifically, whether two different object-action pairs can be correctly identified to be the same affordance. It is interesting to note that for the scoop-able affordance, by using the same action across similar objects, we are in fact simulating how well testing an unseen object could possibly perform.

TABLE II: Affordance Skill Monitoring Results

Object-Action Pair	Precision	Recall	F_1	Accuracy
Cup 1-Scoop	0.50	1.00	0.67	0.50
Cup 2-Scoop	1.00	1.00	1.00	1.00
Parmesan-Scoop	0.67	1.00	0.80	0.75
Pasta Jar-Lift	1.00	1.00	1.00	1.00
Drawer-Pull	1.00	1.00	1.00	1.00
Wooden Box-Push1	1.00	1.00	1.00	1.00
Bread Box-Push2	0.00	0.00	0.00	0.50

A. Experiment Setup

To test if the affordance model can classify an existing object-action pair, we use leave-one-object-out cross validation within an affordance to demonstrate how a robot might test an object for that affordance. This results in three tests for **scoop-able** and four tests for **open-able**. For example, to test if the **Cup1-Scoop** pair would be classified as having the scoop-able affordance, we remove the model learned from the Cup 1 interactions completely and all interactions with Cup 1 then become the test set (10 successful and 10 "near-miss" trajectories). For each test interaction, we use each object-action model within the affordance to evaluate the log-likelihood of that interaction. For scoop-able, this results in 4 different log-likelihood values (from each of the successful and "near-miss" HMMs) and for open-able, 6 log-likehood values. We label an object as having the affordance if the log-likelihood value is greatest with a successful HMM and not if the value is greatest with a "near-miss" HMM.

B. Results

The results of affordance testing can be seen in Table III for scoop-able and Table IV for open-able. However, unlike the results in Section VII, it is difficult to fully understand what a "near-miss" example should be classified as given that the interaction was on an object that did indeed *have that affordance*. Instead, it makes more sense to look only at the interactions that successfully found the affordance. This is shown in Fig. 8 and Fig. 9 with accuracy values for successful interactions reported separately from "near-misses". We only include the full precision, recall, and F_1 scores in Table III and Table IV and the accuracy scores of the "near-miss" interactions to show that the models are not merely classifying all interactions as having the affordance.

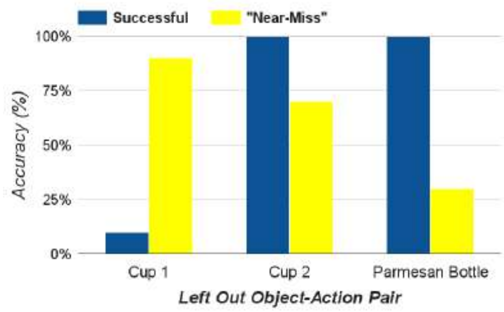


Fig. 8: Scoopable: Accuracy values for Leave One Object Out. The figure shows the accuracy breakdown between successful and "near-miss" interactions

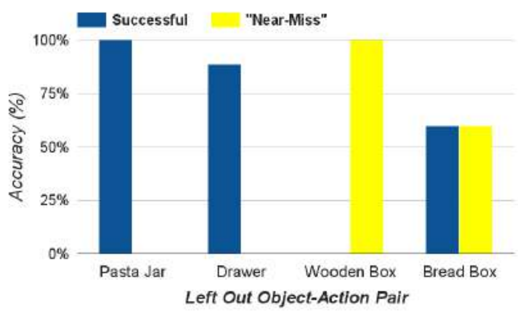


Fig. 9: Openable: Accuracy values for Leave One Object Out. The figure shows the accuracy breakdown between successful and "near-miss" interactions

For scoop-able, the object-action pairs correctly identify an unseen object for both Cup 2 and Parmesan Bottle with accuracy scores of 65% and 85% respectively. Interestingly, Cup1-Scoop does not perform as well. This suggests that the interactions from Cup1-Scoop were not as easily distinguishable, which is supported by our results in Section VII. The results of scoop-able show that for an affordance with relatively similar objects, it is possible to identify an unseen object using the learned object-action models. We look next at how well object-action models perform on an affordance that requires very different actions on dissimilar objects. As expected, scoop-able outperforms open-able for identifying an affordance on unseen objects with an average accuracy score of 67% where open-able has an average accuracy of 53%. Within open-able, the only object that the models could reasonably classify were those of BreadBox-Push2. While the accuracy of PastaJar-Lift and Drawer-Pull are high for successful interactions, it is unclear if it is due to the models truly finding the affordance because all of the interactions (including "near-miss" interactions) were labeled as finding the affordance. For WoodenBox-Push1, the object-action pairs were conservative and did not label any interactions successfully finding the affordance. This could be due to the small F/T values overall felt during the push compared to the other actions that opened objects.

We believe this difference in performance between the two affordances can be attributed to different actions required to

TABLE III: Scoop-able Leave One Object Out

Object-Action Pair	Precision	Recall	F_1	Accuracy
Cup 1-Scoop	0.50	0.10	0.17	0.50
Cup 2-Scoop	0.77	1.00	0.87	0.85
Parmesan Bottle-Scoop	0.59	1.00	0.74	0.65

TABLE IV: Open-able Leave One Object Out

Object-Action Pair	Precision	Recall	F_1	Accuracy
Pastajar-Lift	0.50	1.00	0.67	0.50
Drawer-Pull	0.44	0.89	0.59	0.42
Wooden Box-Push1	0.00	0.00	0.00	0.50
Bread Box-Push2	0.60	0.60	0.60	0.60

find each of these affordances, with open-able requiring more varying actions and scoop-able using similar actions. This suggests that for affordances that require different actions, additional work must be done to adapt and recognize each action (e.g. increasing the number of object-action pairs, including self-exploration, integrating visual information).

IX. Discussion

The results of both monitoring and affordance testing show that the system can successfully learn multiple haptic object-action pairs by using LfD and human-guidance to initiate the exploration and ground affordances with F/T sensing. For five of the seven object-action pairs, we achieve high F_1 scores at identifying successful execution of the action on the object. We then show that affordance monitoring using multiple object-action pairs can correctly identify the scoop-able affordance on an unseen object with high accuracy. More importantly, our approach allowed us to quickly generate vastly different actions for exploration, which allowed us to analyze and gain insight into affordances not typically explored in the robotics community.

Scoop-able outperforms open-able and we believe this is due to the inherent difference between the affordances open-able and scoop-able. While there are several different methods to open an object, there are far fewer ways to scoop. Furthermore, actions vary significantly between the different methods of opening vs. scooping. For example, testing if any object can scoop macaroni, would result in similar sweeping motions with slightly different rotations in the wrist and end-effector offsets (e.g. scooping with a spoon or ladle). However, as seen in this work, open-able can break down into several different affordances. One can imagine open-able as comprising of various affordances (e.g. lift-able, pull-able, push-able) while scoop-able is the "lowest" level of the affordance. However, modeling and understanding "high-level" affordances such as open-able is crucial for robots to truly plan and execute tasks. This interesting distinction between the generality and specificity of different affordances is a main driver of our future work. Our end goal is for robots to reason about high-level affordances at the task planning stage, but then dive down into the low-level representations of how to achieve that affordance on different types of objects when it comes time to decide how to control the manipulator.

While our current system demonstrates capabilities to test for specific object-action pairs, it cannot adapt to new objects. Our ongoing work tackles this directly by allowing the robot to sample from the HMMs and build hybrid control models that use position and haptic feedback to adapt to new objects. This work on action generation fits within the framework of this paper as it will augment the human provided trajectory with robot-generated ones.

X. Conclusion

We showed that learned object-action models achieve good cross-validation performance with 4 of the 7 object-action pairs achieving a perfect F_1 score. Furthermore, by leveraging the set of object-action models per affordance, we can identify 2 haptic affordances with an average accuracy of 67% for the simpler scoop-able affordance and 53% for more complex open-able affordance using only haptic sensing.

In studying haptic affordances, we can begin to understand the role haptics plays in discovering object functions and come closer to building a representation of skills that will allow a robot to achieve tasks in a variety of environments. Furthermore, we have demonstrated that using LfD allows us to quickly provide concrete examples to the robot and allow the robot to discover the "action possibilities" that exist *for the robot* as opposed to any agent. This generated trajectory provides a means to act to sense the environment and human-guided exploration provides a means to obtain high-quality grounded examples of affordances.

References

[1] J. Gibson, "The concept of affordances," *Perceiving, acting, and knowing*, pp. 67–82, 1977.

[2] E. Şahin, M. Çakmak, M. R. Doğar, E. Uğur, and G. Üçoluk, "To afford or not to afford: A new formalization of affordances toward affordance-based robot control," *Adaptive Behavior*, vol. 15, no. 4, pp. 447–472, 2007.

[3] L. Montesano, M. Lopes, A. Bernardino, and J. Santos-Victor, "Learning object affordances: From sensory–motor coordination to imitation," *Transactions on Robotics*, vol. 24, no. 1, pp. 15–26, 2008.

[4] P. Fitzpatrick, G. Metta, L. Natale, S. Rao, and G. Sandini, "Learning about objects through action - initial steps towards artificial cognition," in *International Conference on Robotics and Automation (ICRA)*, Sept 2003, pp. 3140–3145.

[5] A. Stoytchev, "Behavior-grounded representation of tool affordances," in *International Conference on Robotics and Automation (ICRA)*, April 2005, pp. 3060–3065.

[6] M. R. Dogar, E. Ugur, E. Sahin, and M. Cakmak, "Using learned affordances for robotic behavior development," in *International Conference on Robotics and Automation (ICRA)*, 2008, pp. 3802–3807.

[7] T. Hermans, J. M. Rehg, and A. F. Bobick, "Decoupling behavior, perception, and control for autonomous learning of affordances," in *International Conference on Robotics and Automation (ICRA)*, 2013, pp. 4989–4996.

[8] A. L. Thomaz and M. Cakmak, "Learning about objects with human teachers," in *International Conference on Human Robot Interaction (HRI)*, 2009, pp. 15–22.

[9] E. Torres-Jara, L. Natale, and P. Fitzpatrick, "Tapping into touch," 2005.

[10] J. Sinapov, T. Bergquist, C. Schenck, U. Ohiri, S. Griffith, and A. Stoytchev, "Interactive object recognition using proprioceptive and auditory feedback," *The International Journal of Robotics Research*, vol. 30, no. 10, pp. 1250–1262, 2011.

[11] J. A. Fishel and G. E. Loeb, "Bayesian exploration for intelligent identification of textures," *Frontiers in neurorobotics*, vol. 6, 2012.

[12] M. C. Gemici and A. Saxena, "Learning haptic representations for manipulating deformable food objects." in *International Conference on Intelligent Robots and Systems (IROS)*, 2014.

[13] V. Chu, I. McMahon, L. Riano, C. G. McDonald, Q. He, J. M. Perez-Tejada, M. Arrigo, N. Fitter, J. C. Nappo, T. Darrell, and K. J. Kuchenbecker, "Using robotic exploratory procedures to learn the meaning of haptic adjectives," in *International Conference on Robotics and Automation (ICRA)*, 2013, pp. 3048–3055.

[14] T. Bhattacharjee, J. M. Rehg, and C. C. Kemp, "Inferring object properties from incidental contact with a tactile sensing forearm," *arXiv preprint arXiv:1409.4972*, 2014.

[15] T. Bhattacharjee, A. Kapusta, J. M. Rehg, and C. C. Kemp, "Rapid categorization of object properties from incidental contact with a tactile sensing robot arm," in *IEEE-RAS International Conference on Humanoid Robots*, 2013.

[16] V. Sukhoy, V. Georgiev, T. Wegter, R. Sweidan, and A. Stoytchev, "Learning to slide a magnetic card through a card reader," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2012, pp. 2398–2404.

[17] J. Sturm, A. Jain, C. Stachniss, C. C. Kemp, and W. Burgard, "Operating articulated objects based on experience," in *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2010, pp. 2739–2744.

[18] P. Pastor, M. Kalakrishnan, S. Chitta, E. Theodorou, and S. Schaal, "Skill learning and task outcome prediction for manipulation," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, May 2011, pp. 3828–3834.

[19] P. Pastor, M. Kalakrishnan, L. Righetti, and S. Schaal, "Towards associative skill memories," in *IEEE-RAS International Conference on Humanoid Robots*, 2012. [Online]. Available: http://www-clmc.usc.edu/publications//P/PeterPastorHumanoids12.pdf

[20] L. Rozo, P. Jimenez, and C. Torras, "A robot learning from demonstration framework to perform force-based manipulation tasks," *Intelligent Service Robotics*, vol. 6, no. 1, pp. 33–51, 2013.

[21] P. Kormushev, S. Calinon, and D. G. Caldwell, "Imitation learning of positional and force skills demonstrated via kinesthetic teaching and haptic input," *Advanced Robotics*, vol. 25, no. 5, pp. 581–603, 2011.

[22] M. F. Mascolo, "Change processes in development: The concept of coactive scaffolding," *New Ideas in Psychology*, vol. 23, no. 3, pp. 185–196, 2005.

[23] P.-Y. Oudeyer, F. Kaplan, and V. Hafner, "Intrinsic motivation systems for autonomous mental development," *Evolutionary Computation, IEEE Transactions on*, vol. 11, no. 2, pp. 265–286, April 2007.

[24] V. Chu, T. Fitzgerald, and A. L. Thomaz, "Learning object affordances by leveraging the combination of human-guidance and self-exploration," in *Human-Robot Interaction (HRI), 2016 11th ACM/IEEE International Conference on*, In Press.

[25] D. Grollman and A. Billard, "Robot learning from failed demonstrations," *International Journal of Social Robotics*, vol. 4, no. 4, pp. 331–342, 2012.

[26] ——, "Donut as i do: Learning from failed demonstrations," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, May 2011, pp. 3804–3809.

[27] B. Akgun, M. Cakmak, K. Jiang, and A. L. Thomaz, "Keyframe-based learning from demonstration," *Int'l Journal of Social Robotics*, vol. 4, pp. 343–355, 2012.

[28] L. Rabiner and B.-H. Juang, "An introduction to hidden markov models," *ASSP Magazine, IEEE*, vol. 3, no. 1, pp. 4–16, 1986.

[29] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python ," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[30] M. E. Taylor and P. Stone, "Transfer learning for reinforcement learning domains: A survey," *J. Mach. Learn. Res.*, vol. 10, pp. 1633–1685, Dec. 2009. [Online]. Available: http://dl.acm.org/citation.cfm?id=1577069.1755839

[31] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, "Learning and generalization of motor skills by learning from demonstration," in *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, May 2009, pp. 763–768.

[32] P. Pastor, L. Righetti, M. Kalakrishnan, and S. Schaal, "Online movement adaptation based on previous sensor experiences," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, Sept 2011, pp. 365–371.