Learning to Generalize Kinematic Models to Novel Objects

Ben Abbatematteo, Stefanie Tellex, George Konidaris
Department of Computer Science
Brown University
Providence, RI
{babbatem,stefie10,gdk}@cs.brown.edu

Abstract: Robots operating in human environments must be capable of interacting with a wide variety of articulated objects such as cabinets, refrigerators, and drawers. Existing approaches require human demonstration or minutes of interaction to fit kinematic models to each novel object from scratch. We present a framework for estimating the kinematic model and configuration of previously unseen articulated objects, conditioned upon object type, from as little as a single observation. We train our system in simulation with a novel dataset of synthetic articulated objects; at runtime, our model can predict the shape and kinematic model of an object from depth sensor data. We demonstrate that our approach enables a MOVO robot to view an object with its RGB-D sensor, estimate its motion model, and use that estimate to interact with the object.

Keywords: Robots, Learning, Perception, Manipulation

1 Introduction

As robots move out of controlled laboratory environments and into homes and work environments, it is critical that they are capable of readily manipulating common objects with functional parts. For example, a robot should be able to recognize an object like a microwave and infer whether the door is open or closed, as well as the information necessary to change the kinematic state of the door.

Existing approaches to learning to manipulate articulated objects are instance-based [1, 2, 3, 4]. They estimate the kinematic model of an object having observed the object's motion, either demonstrated by a human or generated by the robot through slow, deliberate interaction. This enables manipulation but requires the robot to learn about each object in its environment from scratch, regardless of how similar the object is to those it has previously experienced. We propose to achieve generalization via object types: objects may be categorized according to their shared kinematic structure, such that all objects in a particular class have the same set of kinematic constraints between parts. This allows a robot to be taught the prototypical kinematic structure for many types of objects commonly experienced in human environments, then exploit that knowledge to readily manipulate novel instances of those objects after having recognized them.

We introduce a framework capable of jointly estimating the kinematic model parameters, kinematic state, and geometry of a novel object from RGB-D observations, given its categorization and prior experience with objects of the same type. We use object recognition to select a kinematic model, and use deep neural networks to learn object-specific mappings from depth sensor observations to kinematic model parameters, kinematic state variables, and geometric parameters. Inspired by recent approaches to object recognition and pose estimation [5, 6, 7], we cast kinematic model parameter inference as a regression task given a known kinematic model type specified by the object's categorization. To train these networks, we developed a dataset of simulated articulated objects, recorded ground-truth geometry and kinematic models, then simulated object articulation and rendered depth images. We evaluated the performance of our system with withheld simulated objects, real objects observed with a Kinect sensor, and manipulation scenarios. Our results demonstrate that our networks learn to localize kinematic mechanisms, infer kinematic state, and predict object geometry sufficiently well to enable the manipulation of novel articulated objects on a real robot.

3rd Conference on Robot Learning (CoRL 2019), Osaka, Japan.

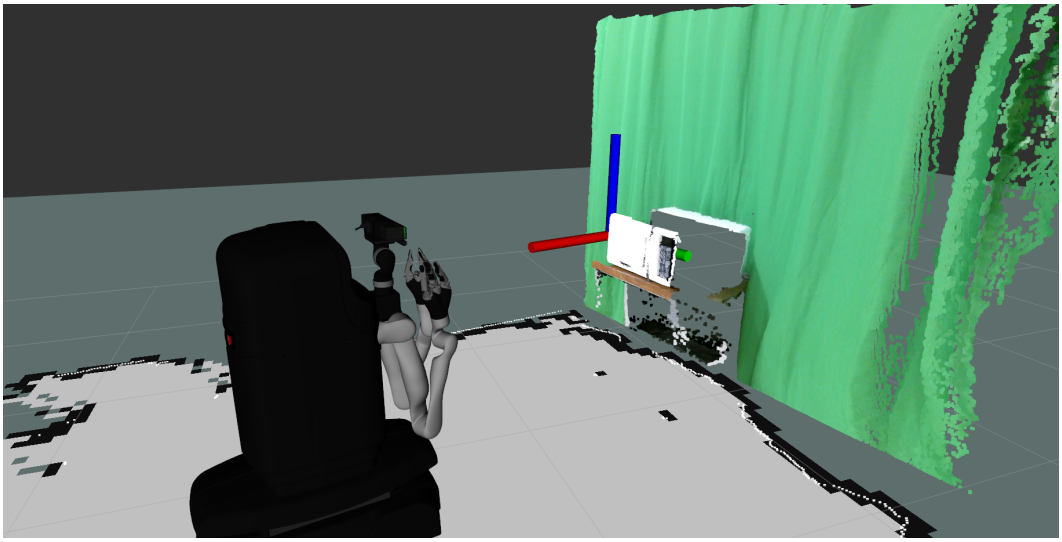


Figure 1: **Kinematic model prediction for a novel object.** Our model predicts the pose and state of a microwave's hinge from a point cloud captured with a Kinect depth sensor.

2 Background

Following Sturm et al. [2], we represent an object's kinematic structure with a graph encoding kinematic relationships between parts. A *kinematic model*, $\mathcal{M}$, encodes a motion constraint between two object parts; the most prevalent model types are revolute (like a door) and prismatic (like a drawer). A kinematic model's *parameter vector*, ϕ, encodes the constraints on the location of the object parts, represented as the 6-DoF pose of the axis of rotation/translation in the camera frame, and the 6-DoF pose of the articulated sub-part's origin in the axis frame. This parameter vector varies with each object instance, depending upon the object's pose and geometry. For example, the exact trajectory of a microwave door will be different for different microwaves. The object's *configuration*, q, represents the articulated pose of the object, encoding angles in revolute models and displacements in prismatic models, and grows in dimensionality with the number of joints present in the object. An object's configuration may change with each observation as forces are applied to its parts. We denote an object's geometry by θ, and choose to represent object geometry by a set of simple primitives: length, width, height, and chirality.

A *kinematic graph* $G = (V_G, E_G)$ consists of a set of vertices $V_G = \{1, ..., p\}$ representing the object's p parts, and a set of undirected edges $E_G \subset V_G \times V_G$ specifying the motion constraints between the parts. Each joint (ij) is assigned a kinematic model $\mathcal{M}_{ij}$, parameter vector ϕ_{ij}, and configuration q_{ij}. The graph of a cabinet, for example, consists of three nodes (a body, a door, and a handle), with a revolute kinematic constraint encoded in the edge between the body and door, as illustrated in Figure 2. In this example, the parameter vector, ϕ, encodes the position and orientation of the axis of rotation between the body and the door as well as the radius of the door's rotation, and the configuration, q, encodes the angle made by the door and the body. The kinematic model $\mathcal{M}$, is revolute. Our approach learns to estimate kinematic model parameters from sensor data for novel object instances by training on a dataset of other instances of that object class.

3 Learning to Generalize Object Kinematics

Our goal is to map a segmented depth image of an object to its kinematic graph parameters, configuration, and geometry. For example, the robot will observe a depth image of a microwave and estimate the location of the microwave door's axis of rotation, the transform from that axis to the center of the door, as well as the angle between the door and the body, and a simple parameterization of the object's geometry. We begin by making the assumption that an object's kinematic structure is directly specified by its category. Thus, for an object to be a member of a class, it must possess the particular kinematic graph connectivity shared among all instances of that class. Additionally, we

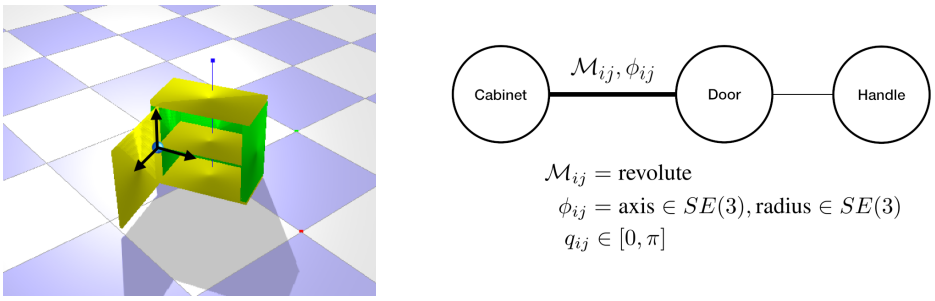


Figure 2: **Kinematic Graph.** A simulated object, annotated with the pose of its axis of rotation and corresponding kinematic graph. Our networks are trained to estimate model parameters ϕ_{ij} and configuration q_{ij} for each joint of the object. We use the object's categorization to select each model, $\mathcal{M}_{ij}$.

assume that the kinematic graph structure of each object class is known a priori, leaving the estimation of prototypical graphs for each class to future work. This assumption enables us to forgo fitting a kinematic model type to a sequence of observations and proceed to estimating model parameters immediately following object recognition. As such, our system is able to generalize within object categories without first observing a trajectory. We cast this as a regression problem, and develop a learning framework that is trained on a dataset of simulated articulated objects before being tested in reality.

3.1 Model Description

The robot takes as input an RGB-D image. We use off-the-shelf object detection algorithms to classify and segment the object from the rest of the scene. We manually specify a kinematic model for each object category. We aim to estimate $p(\phi, q_t, \theta | x_t, c)$, the probability of the object's kinematic model parameters, ϕ, its present configuration, q_t, and a simple parameterization (length, width, height, chirality) of the object's geometry, θ, conditioned upon a single depth image, x_t, and the object's class label, c. In practice, we train a unique mixture density network [8] for each object class c, which consumes a depth image and regresses to the parameters of a mixture of Gaussians:

$$p(\phi, q_t, \theta | x_t, c) = \sum_{i}^{m} \pi_i^c(x_t)\, \mathcal{N}(\phi, q_t, \theta | \mu_i^c(x_t), \sigma_i^c(x_t)), \tag{1}$$

where x_t is a depth image sampled from the object's articulation at time t, the object class c is known to the learner, and $\pi_i^c(x_t), \mu_i^c(x_t), \sigma_i^c(x_t)$ are neural networks which consume depth images and output mixing parameters, means, and diagonal covariances, respectively. We use a ResNet-18 [9] backbone with a fully connected network head for each parameter (π, σ, μ) of the mixture of Gaussians. We found $m = 20$ mixture components to be sufficient.

3.2 Training

Training the model requires point clouds annotated with the underlying pose, geometry, and kinematics of the articulated object. To generate this data, we created a dataset of synthetic articulated objects (see Section 3.3). Ground truth kinematic models allowed us to train inference networks by maximizing the probability of the labels (ϕ, q_t, θ) under the model $p(\phi, q_t, \theta | x_t, c)$:

$$\mathcal{L} = -\mathbb{E}\left[\log p(\phi, q_t, \theta | x_t, c)\right]. \tag{2}$$

We sampled batches of individual image-label pairs rather than training on full object articulation demonstrations to enable one-shot model inference with new objects.

3.3 Synthetic Dataset

Our synthetic dataset consists 6 object categories (cabinet, drawer, microwave, toaster oven, two-door cabinet, refrigerator). The kinematic graph for each of these classes, known a priori to the learner, was specified in Universal Robot Description Format (URDF) and rendered via Mujoco

[10]. Example objects are shown in Figure 3. Two-door cabinets and refrigerators are modeled as having two degrees of freedom, while the rest of the objects are modeled with one degree of freedom.

A generator specific to each object class sampled object pose and geometry, and computed the location(s) of kinematic mechanism(s). The simulator rendered observations from each object's articulation. We sampled object position uniformly from the view frustum of the camera up to a maximum depth dependent upon object size. Object orientations were sampled uniformly from the range $[-\frac{\pi}{4}, \frac{\pi}{4}]$ about the z-axis. A unique neural network was trained for each object class, using $160,000$ depth-image/label pairs ($10,000$ objects) for training and $16,000$ pairs ($1,000$ withheld objects) for testing per class. We make the dataset publicly available here for broader use by the community. More details are provided in Appendix A.

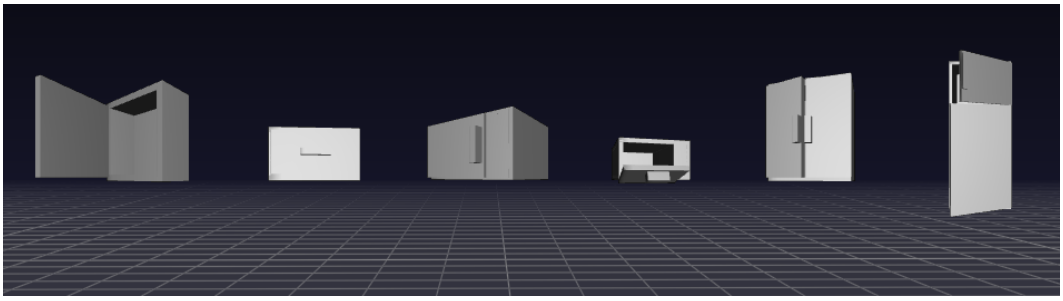


Figure 3: **Example synthetic objects:** cabinet, drawer, microwave, toaster oven, cabinet2, refrigerator, respectively.

4 Experimental Evaluation

We evaluated our approach's ability to jointly estimate kinematic model parameters and kinematic configuration of articulated objects from depth images using withheld simulated objects and real objects observed with a Kinect depth sensor. Additionally, to demonstrate that our approach indeed enables the manipulation of real articulated objects, we use our model to open a microwave door and a drawer with a real mobile manipulator.

4.1 Predicting Kinematics From Depth Images

Parameter estimation accuracy was evaluated by measuring the Euclidean distance between the estimated and the true pose of the axis of rotation/translation, and the orientation error of the axis of rotation/translation about the z-axis. Kinematic state estimation accuracy was evaluated by measuring the error in kinematic state, reported in degrees for revolute mechanisms and centimeters for prismatic mechanisms. These estimates are produced from a single depth image, and are not accumulated over time. The error is computed using the maximum likelihood sample from the mixture density network for each class. As a baseline, we compared against the mean of each quantity for each object class (Mean), nearest neighbor in depth-image space (NN), and a variant of ICP in which we registered each test observation to the mean model for each object class in 16 different poses and found the best match (ICP). The ICP baseline finds a 6-DoF transformation, but we report error only about the z-axis for comparison. Table 1 displays these results for $1,000$ withheld objects for each of the six object classes studied. Our framework learns to infer the position, orientation, and state of simulated kinematic mechanisms to within a few centimeters/degrees.

Next, we evaluated the system's accuracy on objects in the real world. We evaluated our trained models with one real object per class, show in Figure 4. Recognition and segmentation were performed using Mask R-CNN [11, 12] pretrained on the MS COCO dataset [13] for the microwave and refrigerator object classes, and manually for the others. Kinematic mechanism poses were acquired using AR tags, and configurations were held constant. Between 47 and 170 samples were acquired for each class. For real objects, we found the model's accuracy improved when only considering points in the point cloud as candidates for the mechanism position. To compute the estimates for Table 2, points in the point cloud were scored by computing their likelihood under the model, and the

	Axis Position Error (cm)	**Axis Rotation Error** (degrees)	**Configuration Error**
Cabinet - Mean	53.83 ± 18.36	10.81 ± 14.29	40.62 ± 26.06 °
Cabinet - NN	18.08 ± 11.31	10.39 ± 12.02	19.66 ± 17.30 °
Cabinet - ICP	35.68 ± 17.36	34.75 ± 26.76	40.58 ± 26.13 °
Cabinet - Ours	$\mathbf{1.84 \pm 4.43}$	$\mathbf{0.36 \pm 1.03}$	$\mathbf{1.61 \pm 1.70}$ °
Drawer - Mean	55.59 ± 17.98	11.22 ± 14.42	9.75 ± 5.43 cm
Drawer - NN	16.51 ± 9.99	8.72 ± 11.84	9.89 ± 8.03 cm
Drawer - ICP	22.8 ± 12.36	19.33 ± 25.02	16.47 ± 11.16 cm
Drawer - Ours	$\mathbf{5.23 \pm 3.91}$	$\mathbf{1.14 \pm 3.38}$	$\mathbf{4.64 \pm 4.05}$ cm
Microwave - Mean	50.78 ± 16.73	11.33 ± 14.28	22.74 ± 12.88 °
Microwave - NN	18.08 ± 11.31	10.39 ± 12.02	19.66 ± 17.30 °
Microwave - ICP	21.29 ± 10.76	25.3 ± 20.0	40.64 ± 26.14 °
Microwave - Ours	$\mathbf{1.28 \pm 1.31}$	$\mathbf{0.34 \pm 1.58}$	$\mathbf{1.73 \pm 2.11}$ °
Toaster - Mean	61.65 ± 18.58	11.48 ± 14.08	22.75 ± 12.89 °
Toaster - NN	14.59 ± 8.37	10.58 ± 12.40	18.79 ± 20.70 °
Toaster - ICP	16.22 ± 5.9	21.26 ± 20.22	40.73 ± 26.15 °
Toaster - Ours	$\mathbf{2.33 \pm 1.46}$	$\mathbf{0.80 \pm 1.60}$	$\mathbf{3.22 \pm 4.06}$ °
Cabinet2 - Mean	62.43 ± 20.04	11.16 ± 14.41	42.91 ± 21.52 °
Cabinet2 - NN	17.16 ± 10.20	10.81 ± 11.95	39.95 ± 35.99 °
Cabinet2 - ICP	19.56 ± 9.9	26.22 ± 22.92	66.08 ± 33.77 °
Cabinet2 - Ours	$\mathbf{3.81 \pm 2.71}$	$\mathbf{2.14 \pm 4.59}$	$\mathbf{11.97 \pm 9.56}$ °
Refrigerator - Mean	108.96 ± 56.02	11.17 ± 14.23	35.23 ± 23.09 °
Refrigerator - NN	24.69 ± 16.00	8.92 ± 11.39	24.07 ± 32.20 °
Refrigerator - ICP	58.25 ± 35.16	27.15 ± 19.89	42.3 ± 29.6 °
Refrigerator - Ours	$\mathbf{4.67 \pm 9.17}$	$\mathbf{0.75 \pm 3.93}$	$\mathbf{6.59 \pm 11.17}$ °

Table 1: **Simulation Results.** Mechanism localization and configuration results are presented as mean and standard deviation of the error, evaluated over 1000 withheld objects in simulation, measured using the model's maximum likelihood sample.

maximally likely point was selected as the system's estimate. Our framework learns to accurately generalize kinematic models to real, novel objects after being trained exclusively in simulation.

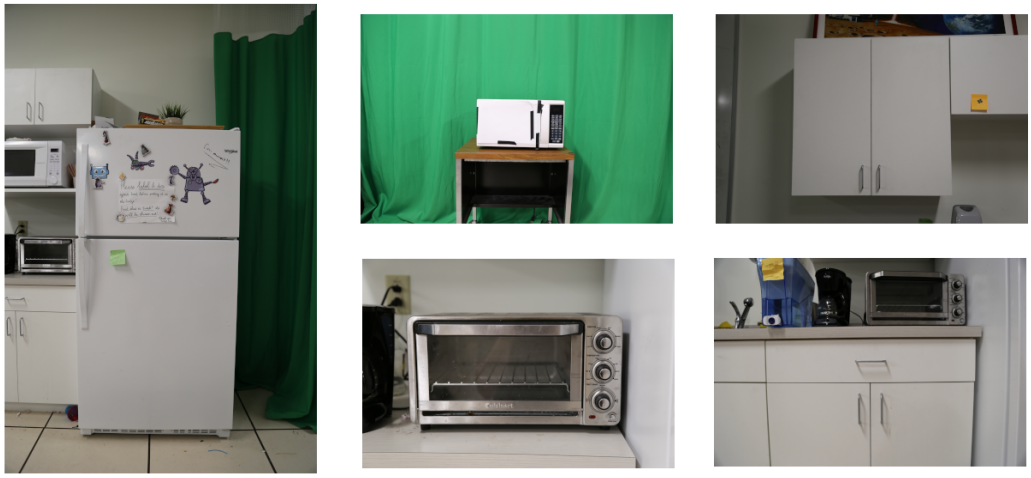


Figure 4: **Real objects studied**. Clockwise: refrigerator, microwave, 2-DoF cabinet, drawer, toaster oven. To obtain 1-DoF cabinet data, we segmented the 2-DoF cabinet into its respective halves.

	Axis Position Error (cm)	**Axis Rotation Error** (degrees)	**Configuration Error**
Cabinet	14.96 ± 13.37	7.82 ± 8.68	15.27 ± 10.75 °
Drawer	9.12 ± 5.97	1.32 ± 1.83	11.45 ± 3.77 cm
Microwave - Closed	3.96 ± 3.48	2.65 ± 3.60	12.96 ± 14.34 °
Microwave - Open	11.87 ± 6.97	5.45 ± 4.81	19.39 ± 9.72 °
Toaster	6.30 ± 3.56	2.79 ± 3.31	10.32 ± 10.47 °
Cabinet2	6.55 ± 3.10	3.02 ± 3.39	16.49 ± 8.87 °
Refrigerator	3.56 ± 2.32	2.59 ± 3.62	1.30 ± 2.85 °

Table 2: **Real Object Results.** Mechanism localization and configuration results are presented for real sensor data with real objects. Objects were imaged in a closed configuration unless denoted otherwise. Errors were measured using the maximally likely point in the point cloud under the model. Ground truth measurements were obtained using AR tags.

Performance on the simulated refrigerator object class is relatively low due to more pose variation in the dataset in order to accommodate for the refrigerator's size, and more samples with only partial views of the object. Interestingly, networks trained on this object class generalized relatively well to the real object. Additionally, the drawer object class is challenging in both simulation and reality due to occlusion of the mechanism's pose during actuation; this affects the accuracy of the model on the real data of the open microwave, as well.

The error rate of the model on the 1-DoF cabinet class is higher due to ambiguity in which side of the object the mechanism resides on. In simulation, there are sufficient artifacts to predict this accurately, but in reality, the model makes some erroneous predictions. Resulting samples from the model are visualized in Figure 5, wherein the agent has assigned probability to both sides of a 1-DoF cabinet after several estimates of the cabinet's hinge pose.

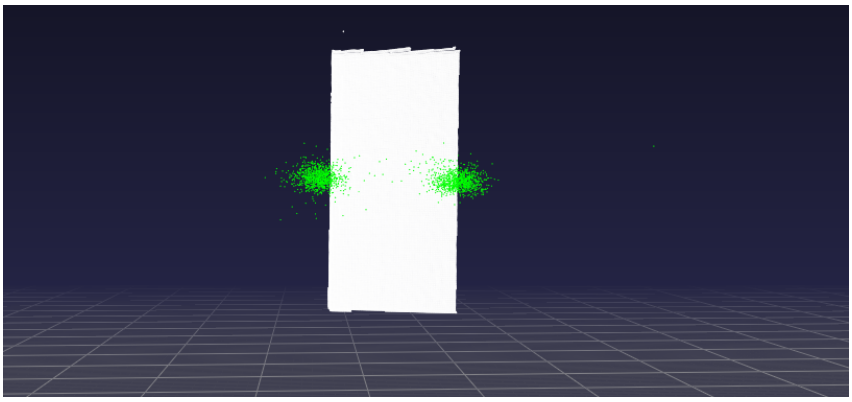


Figure 5: **Model estimates under uncertainty.** Samples drawn from the model of a cabinet's hinge pose, accumulated over time and overlaid on a point cloud. With the door closed, it is challenging to discern whether the cabinet opens left or right. Accordingly, our model has assigned probability to both sides of the object.

4.2 Using Estimated Kinematic Models for Manipulation

Finally, we evaluated our system's accuracy by demonstrating successful manipulation of novel articulated objects in the real world. The robot, a Kinova MOVO mobile manipulator, was driven around the area in front of each object, and recorded observations of the object and its own pose in a pre-computed map of the room. Using these observations, the agent estimated the position and orientation of the object's axis of rotation/translation, articulated pose, and geometry. We provided the agent with a fixed grasp on the object's handle, and computed a series of waypoints as a function of the estimated object kinematics, end-effector pose, and desired object configuration. Using our system's estimates, we were able to generate kinematic motion plans that achieved the desired manipulations, as shown in Figures 6 and 7, and in the supplemental video.

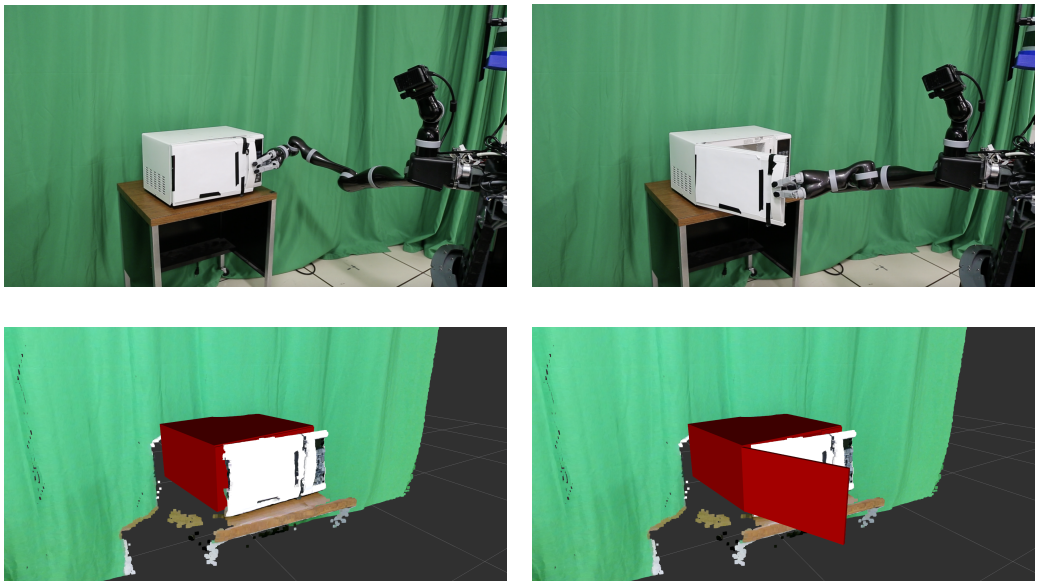


Figure 6: **Novel object manipulation.** *Top:* A MOVO robot manipulating a novel microwave using our system's estimates of the object's kinematics and geometry. *Bottom:* Rendered estimates of the object's pose, geometry, and kinematic state before and after the manipulation.

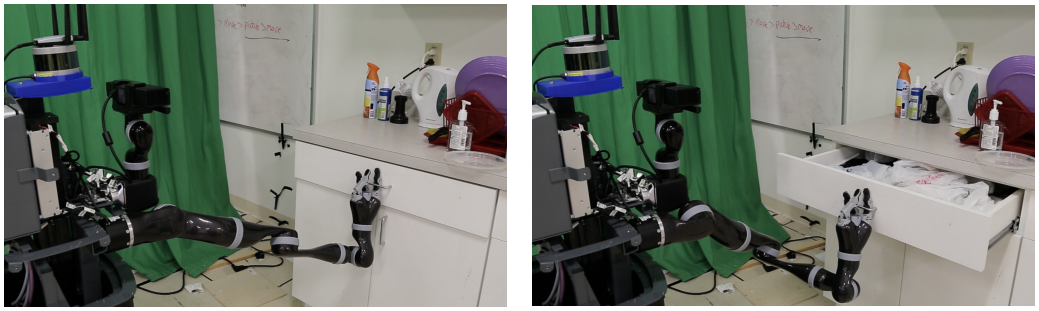


Figure 7: **Novel object manipulation.** A MOVO robot manipulating a novel drawer using a kinematic model estimated from a single depth image.

5 Related Work

The problem of estimating the kinematic model and articulated pose of an individual object has been studied extensively in the robotics literature.

Several approaches estimate kinematic models from visual demonstrations, tracking sub-parts either with fiducial markers or by clustering feature trajectories [2, 3, 14, 15, 16]. Other approaches explore object mechanisms and joint dependencies interactively, tracking sub-parts via fiducial markers or by clustering feature trajectories [1, 4, 17, 18, 19, 20, 21]. These systems accurately estimate the kinematic structure of an object having first directly observed or estimated a sequence of sub-part poses. However, they require observing the very manipulation the robot aims to achieve. Moreover, these methods lack a notion of object type and typically assume uniform priors over model parameters, whereas our approach leverages object recognition and priors implicit in the data for each object category. Most critically, existing approaches fail to generalize across similar objects, learning about each new object from scratch. Our method complements these approaches: our system provides an initial estimate of the kinematic model and its parameters, which could be subsequently refined through interaction and observation. Crucially, the aforementioned approaches require observation of an object's articulation, whereas our system can provide model and parameter estimates from observations of static objects.

Brookshire and Teller [22] and Desingh et al. [23] estimate the configuration of a system with a known kinematic model. Our work considers a simplified setting in which objects have few degrees of freedom, enabling an end-to-end, differentiable regression approach in lieu of a more complicated configuration estimate. An interesting avenue of work is integrating a more robust configuration estimate when generalizing kinematic models to novel objects.

Sung et al. [24] learn multi-modal embeddings of visual data, language commands, and trajectories for manipulating common object parts. In contrast, our work focuses on generalizing using object classes rather than object parts, does not rely on natural language, and explicitly models latent kinematic states, enabling optimization-based manipulation at runtime.

Englert and Toussaint [25] propose kinematic morphing networks in order to transfer kinematic models and manipulation policies to novel doors. This work addresses generalization of 1-DoF revolute kinematic models by iteratively warping an observed point cloud to a class prototype. While this approach enables generalization, it is unlikely to scale to objects with more degrees of freedom or complicated geometry. Additionally, this work does not estimate articulated pose. In contrast, our approach is probabilistic, estimates articulated pose, studies several object classes with real data, and is shown to enable the manipulation of real articulated objects.

Object representations without kinematics are common for tasks such as grasping [26, 27, 28], pose estimation [6, 29, 30, 7] and object detection [31, 11, 32]. Human articulated pose estimation has also been studied extensively, i.e., Pavlakos et al. [33], Wei et al. [5].

6 Discussion and Conclusions

We presented a framework for the perception and manipulation of articulated objects, without the need for first observing the object's articulation at runtime, by leveraging object recognition and a novel dataset of synthetic articulated objects.

Our work assumes that models are directly specified by object class, and that each object class has exactly one appropriate kinematic model. This assumption could be relaxed to accommodate the variation present in many real object classes. Conversely, while previous approaches to learning object classes have relied exclusively on appearance, our work suggests that functionality should also play a key role.

Our work introduces and relies upon a simulated dataset which has several shortcomings. The simulated objects present artifacts and are not photo-realistic, forcing us to use only depth images and to regularize the models heavily during training. Additionally, the length, width, and height of objects was sampled independently, which does not accurately reflect the true joint distribution of object shapes in many categories. Furthermore, our dataset only includes objects in isolation; objects are often occluded in real scenes, and this presents a challenge for our models. This avenue of research could benefit immensely from a large dataset of real articulated objects. Nonetheless, our real-world experiments provide evidence that our system does generalize well to real objects.

Robots will have to be able to generalize their manipulation skills in order to be useful in real environments. Our system accurately produces probabilistic estimates of kinematic model parameters, articulated pose, and object geometry from individual sensor observations; our experiments show that our approach enables the perception of household articulated objects immediately following object detection—thus enabling a robot to manipulate novel instances of familiar object classes.

Acknowledgments

This research was supported by NSF CAREER Award 1844960 to Konidaris, and by the ONR under the PERISCOPE MURI Contract N00014-17-1-2699. Disclosure: George Konidaris is the Chief Roboticist of Realtime Robotics, a robotics company that produces a specialized motion planning processor.

References

[1] D. Katz and O. Brock. Manipulating articulated objects with interactive perception. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 272–277. IEEE, 2008.

[2] J. Sturm, C. Stachniss, and W. Burgard. A probabilistic framework for learning kinematic models of articulated objects. *Journal of Artificial Intelligence Research*, 41:477–526, 2011.

[3] S. Pillai, M. R. Walter, and S. Teller. Learning articulated motions from visual demonstration. *arXiv preprint arXiv:1502.01659*, 2015.

[4] K. Hausman, S. Niekum, S. Osentoski, and G. S. Sukhatme. Active articulation model estimation through interactive perception. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 3305–3312. IEEE, 2015.

[5] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016.

[6] B. Burchfiel and G. Konidaris. Hybrid bayesian eigenobjects: Combining linear subspace and deep network methods for 3d robot vision. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6843–6850. IEEE, 2018.

[7] H. Wang, S. Sridhar, J. Huang, J. Valentin, S. Song, and L. J. Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2642–2651, 2019.

[8] C. M. Bishop. Mixture density networks. 1994.

[9] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[10] E. Todorov, T. Erez, and Y. Tassa. Mujoco: A physics engine for model-based control. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 5026–5033. IEEE, 2012.

[11] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2980–2988. IEEE, 2017.

[12] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He. Detectron. `https://github.com/facebookresearch/detectron`, 2018.

[13] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[14] S. Niekum, S. Osentoski, C. G. Atkeson, and A. G. Barto. Online bayesian changepoint detection for articulated motion models. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 1468–1475. IEEE, 2015.

[15] A. F. Daniele, T. M. Howard, and M. R. Walter. Learning articulated object models from language and vision. 2017.

[16] J. Yan and M. Pollefeys. Automatic kinematic chain building from feature trajectories of articulated objects. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 1, pages 712–719. IEEE, 2006.

[17] P. R. Barragän, L. P. Kaelbling, and T. Lozano-Pérez. Interactive bayesian identification of kinematic mechanisms. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 2013–2020. IEEE, 2014.

[18] R. M. Martin and O. Brock. Online interactive perception of articulated objects with multi-level recursive estimation based on task-specific priors. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2494–2501, Sept 2014. doi:10.1109/IROS.2014.6942902.

[19] J. Kulick, S. Otte, and M. Toussaint. Active exploration of joint dependency structures. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 2598–2604. IEEE, 2015.

[20] M. Baum, M. Bernstein, R. Martin-Martin, S. Höfer, J. Kulick, M. Toussaint, A. Kacelnik, and O. Brock. Opening a lockbox through physical exploration. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 461–467, Nov 2017. doi:10.1109/HUMANOIDS.2017.8246913.

[21] J. Sturm, A. Jain, C. Stachniss, C. C. Kemp, and W. Burgard. Operating articulated objects based on experience. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2739–2744. IEEE, 2010.

[22] J. Brookshire and S. Teller. Articulated pose estimation using tangent space approximations. *The International Journal of Robotics Research*, 35(1-3):5–29, 2016.

[23] K. Desingh, S. Lu, A. Opipari, and O. C. Jenkins. Efficient nonparametric belief propagation for pose estimation and manipulation of articulated objects. *Science Robotics*, 4(30), 2019. doi:10.1126/scirobotics.aaw4523. URL `https://robotics.sciencemag.org/content/4/30/eaaw4523`.

[24] J. Sung, S. H. Jin, and A. Saxena. Robobarista: Object part based transfer of manipulation trajectories from crowd-sourcing in 3d pointclouds. In *Robotics Research*, pages 701–720. Springer, 2018.

[25] P. Englert and M. Toussaint. Kinematic morphing networks for manipulation skill transfer. In *Proceedings of the IEEE International Conference on Intelligent Robotics Systems*, 2018.

[26] J. Bohg, A. Morales, T. Asfour, and D. Kragic. Data-driven grasp synthesisa survey. *IEEE Transactions on Robotics*, 30(2):289–309, April 2014. ISSN 1552-3098. doi:10.1109/TRO.2013.2289018.

[27] E. Jang, C. Devin, V. Vanhoucke, and S. Levine. Grasp2vec: Learning object representations from self-supervised grasping. *Proceedings of Machine Learning Research 87:99-112*, 2018.

[28] P. R. Florence, L. Manuelli, and R. Tedrake. Dense object nets: Learning dense visual object descriptors by and for robotic manipulation. *Proceedings of Machine Learning Research 87:373-385*, 2018.

[29] M. Sundermeyer, Z.-C. Marton, M. Durner, M. Brucker, and R. Triebel. Implicit 3d orientation learning for 6d object detection from rgb images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 699–715, 2018.

[30] J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birchfield. Deep object pose estimation for semantic robotic grasping of household objects. *arXiv preprint arXiv:1809.10790*, 2018.

[31] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[32] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[33] G. Pavlakos, L. Zhu, X. Zhou, and K. Daniilidis. Learning to estimate 3d human pose and shape from a single color image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 459–468, 2018.

A Synthetic Dataset

Our simulated dataset of procedurally generated articulated objects (available here) consists of 6 categories (cabinet, drawer, microwave, toaster oven, two-door cabinet, refrigerator). The quantities which were randomized during data generation and their ranges are displayed in the tables below.

Geometric parameters were sampled from independent, uniform distributions over depth, width, and height. Cabinets that open clockwise/counterclockwise were generated with equal probability.

Table 3: Randomized Geometric Parameters

	Depth (cm)	Width (cm)	Height (cm)
Cabinet	56 - 64	60 - 140	60 - 140
Drawer	45 - 60	30 - 76	10 - 30
Microwave	25 - 56	40 - 76	22 - 45
Toaster	20 - 50	40 - 60	20 - 50
Cabinet2	56 - 64	60 - 140	60 - 140
Refrigerator	60 - 80	60 - 90	82 - 88

Pose parameters were sampled similarly for each object. The camera was positioned looking down the positive x-axis with the positive z-axis upward. The x,y,z values refer to the center of the base of the object. Rotation denotes angle about the z-axis. Configuration denotes articulated pose, reported in radians for revolute mechanisms and centimeters for prismatic joints. For 2-DoF objects, configuration was randomized to cover the full 2-dimensional configuration space in the range provided.

Table 4: Randomized Pose Parameters

	x (m)	y (m)	z (m)	Rotation (rad)	Configuration
Cabinet	[1.0, 2.0]	[-0.5, 0.5]	[-0.7, 0.3]	$[-\frac{\pi}{4}, \frac{\pi}{4}]$	$[0, \frac{\pi}{2}]$ rad
Drawer	[1.0, 2.0]	[-0.5, 0.5]	[-0.8, 0.0]	$[-\frac{\pi}{4}, \frac{\pi}{4}]$	[0, 40] cm
Microwave	[1.0, 2.0]	[-0.5, 0.5]	[-0.7, 0.3]	$[-\frac{\pi}{4}, \frac{\pi}{4}]$	$[0, \frac{\pi}{2}]$ rad
Toaster	[1.0, 2.0]	[-0.5, 0.5]	[-0.7, 0.3]	$[-\frac{\pi}{4}, \frac{\pi}{4}]$	$[0, \frac{\pi}{2}]$ rad
Cabinet2	[1.0, 2.0]	[-0.5, 0.5]	[-0.7, 0.3]	$[-\frac{\pi}{4}, \frac{\pi}{4}]$	$[0, \frac{\pi}{2}]$ rad
Refrigerator	[1.5, 3.5]	[-1.0, 1.0]	[-1.5, -0.7]	$[-\frac{\pi}{4}, \frac{\pi}{4}]$	$[0, \frac{\pi}{2}]$ rad