Learning to Manipulate Deformable Objects without Demonstrations

Yilin Wu*, Wilson Yan*, Thanard Kurutach, Lerrel Pinto, Pieter Abbeel
University of California, Berkeley
{yilin-wu,wilson1.yan}@berkeley.edu

***Abstract*—In this paper we tackle the problem of deformable object manipulation through model-free visual reinforcement learning (RL). In order to circumvent the sample inefficiency of RL, we propose two key ideas that accelerate learning. First, we propose an iterative pick-place action space that encodes the conditional relationship between picking and placing on deformable objects. The explicit structural encoding enables faster learning under complex object dynamics. Second, instead of jointly learning both the pick and the place locations, we only explicitly learn the placing policy conditioned on random pick points. Then, by selecting the pick point that has Maximal Value under Placing (MVP), we obtain our picking policy. This provides us with an informed picking policy during testing, while using only random pick points during training. Experimentally, this learning framework obtains an order of magnitude faster learning compared to independent action-spaces on our suite of deformable object manipulation tasks with visual RGB observations. Finally, using domain randomization, we transfer our policies to a real PR2 robot for challenging cloth and rope coverage tasks, and demonstrate significant improvements over standard RL techniques on average coverage.**

I. Introduction

Over the last few decades, we have seen tremendous progress in robotic manipulation. From grasping objects in clutter [20, 16, 13, 10, 6] to dexterous in-hand manipulation of objects [1, 25], modern robotic algorithms have transformed object manipulation. But much of this success has come at the price of making a key assumption: rigidity of objects. Most robot algorithms often require (implicitly or explicitly) strict rigidity constraints on objects. But the objects we interact with everyday, from the clothes we put on to shopping bags we pack, are deformable. Deformable object manipulation has been a long standing problem [24, 8, 21, 14, 19], with two unique challenges: state estimation and complex dynamics.

One of the recent breakthroughs in robotics has been the development of model-free visual policy learning [9, 17, 1], where robotic algorithms can reason about interactions directly from raw sensory observations. This can alleviate the challenge of state estimation for deformable objects [15], since we can directly learn on images. Moreover, since these methods do not require an explicit model of the object [11], they can overcome the challenge of having complex deformable object dynamics. However, a major issue is that model-free learning has notoriously poor sample complexity [4].

In this work, we present three contributions in this paper: (a) we propose a novel learning algorithm for picking based on the maximal value of placing; (b) we show that the conditional action space formulation significantly accelerates the learning for deformable object manipulation; and (c) we demonstrate transfer to real-robot cloth and rope manipulation using our proposed formulation.

II. Approach

We look at a more amenable action space while retaining the expressivity of the general action space: pick and place. The pick and place action space has had a rich history in planning with rigid objects [2, 12]. Here, the action space is the location to pick (or grasp) the object a^t_{pick}

We factor the pick-place policy as:

$$\pi_{factor} \equiv \pi_{pick}(a_{pick}|o) \cdot \pi_{place}(a_{place}|o, a_{pick}) \qquad (1)$$

This factorization will allow the policy to reason about the conditional dependence of placing on picking.

However, in the context of RL, we face another challenge: action credit assignment. Using RL, the reward for a specific behavior comes through the cumulative discounted reward at the end of an episode. This results in the *temporal credit assignment* problem where attributing the reward to a specific action is difficult.

To overcome the action credit assignment problem, we propose a two-stage learning scheme. Here the key insight is that training a placing policy can be done given a full-support picking policy and the picking policy can be obtained from the placing policy by accessing the Value approximator for placing. Algorithmically, this is done by first training π_{place} conditioned on picking actions from the uniform random distribution $\mathbf{U}_{pick}$. Using Soft-Actor Critic (SAC) [7], we train and obtain $\pi_{place}(a_{place}|o, a_{pick})$, s.t. $a_{pick} \sim \mathbf{U}_{pick}$ as well as the place value approximator $V^{\pi_{place}}_{place}(o, a_{pick})$. Since the value is also conditioned on pick point a_{pick}, we can use this to obtain our picking policy as:

$$\pi_{pick} \equiv \arg\max_{a_{pick}} V^{\pi_{place}}_{place}(o, a_{pick}) \qquad (2)$$

We call this picking policy: Maximum Value under Placing (MVP). MVP allows us get an informed picking policy without having to explicitly train for picking. This makes training efficient for off-policy learning with conditional action spaces especially in the context of deformable object manipulation.

III. Experimental Evaluation

A. Cloth Manipulation in Simulation

We use a MuJoCo 2.0 [23] to evaluate our method on rope and cloth spreading tasks. We compare our method against several baslines: a random policy, a joint factorization of our pick-place policy, and a autoregressive factorization of the pick-place policy.

B. Does conditional pick-place learning help?

To understand the effects of our learning technique, we compare our learned placing with uniform pick technique with the independent representation. We can see that using our proposed method shows improvement in learning speed for state-based cloth experiments, and image-based experiments in general. The state-based rope experiments do not show much of a difference due to the inherent simplicity of the tasks. Our method shows significantly higher rewards in the cloth simplified environment, and learns about 2X faster in the harder cloth environment. We also note that the independent and conditional baselines perform better on the full state-based cloth environment compared to when constraining the task to four corner pick points. This most likely occurs since the simplified cloth task structures its pick action as 4 discrete locations, which increases the likelihood of mode collapse on a single corner compared to when using a continuous pick representation for the full Cloth environment. For image-based experiments, the baseline methods do no better than random while our method gives an order of magnitude (5-10X) higher performance for reward reached. The independent and conditional factored policies for image-based cloth spreading end up performing worse than random, suggesting mode collapse that commonly occurs in difficult optimization problems [5]. Note that to strengthen the baselines, we add additional rewards to bias the pick points on the cloth; However, this still does not significantly improve performance for the challenging image based tasks. This demonstrates that conditional learning indeed speeds up learning for deformable object manipulation especially when the observation is an image.

C. Does setting the picking policy based on MVP help?

One of the key contributions of this work is to use the placing value to inform the picking policy (Eq. 2) without explicitly training the picking policy. As in both state-based and image-based case training with MVP gives consistently better performance. Even when our conditional policies with uniform pick location fall below the baselines in Cloth (State) and Rope (State), using MVP significantly improves the performance. Although MVP brings relatively smaller boosts in performance compared to the gains brought by the learned placing with uniform pick method, we observe that the learned placing with uniform pick policy already achieves a high success rate on completing the task, and even a small boost in performance is visually substantial when running evaluations in simulation and on our real robot.

D. How do we transfer our policies to a real robot?

To transfer our policies to the real-robot, we use domain randomization (DR) [22, 17, 18] in the simulator along with using images of real cloths. Randomization is performed on visual parameters (lighting and textures) as well physics (mass and joint friction) of the cloth. Additionally, in simulation evaluation, we notice no degradation in performance due to DR while training using MVP. Physics randomization most likely had little-to-no benefit (compared to visual randomization) to the learning process due to the fact that the simulator itself is already a little noisy.

In order to perform actions on our PR2 robot, we first calibrate pixel-space actions with robot actions. This is done by collecting 4-5 points mapping between robot x, y coordinates to image row, column pixel locations, and fitting a simple linear map. Next, we capture RGB images from a head-mounted camera on our PR2 robot and input the image into our policy learned in the simulator. Since a_{pick} and a_{place} are both defined as points on the image, we can easily command the robot to perform pick-place operations on the deformable object placed on the green table by planning with Moveit! [3].

E. Evaluation on the real robot

We evaluate our policy on the *rope-spread* and *cloth-spread* experiments. From our results, we see that policies trained using MVP are successfully able to complete both spreading tasks. For our cloth spreading experiment, we also note that due to domain randomization, a single policy can spread cloths of different colors. For quantitative evaluations, we select 4 start configurations for the cloth and the rope and compare with various baselines (Table I) on the spread coverage metric. For the rope task, we run the policies for 20 steps, while for the much harder cloth task we run policies for 150 steps. The large gap between MVP trained policies and independent policies supports our hypothesis that the conditional structure is crucial for learning deformable object manipulation. Robot execution videos can be accessed from the video submission. We observe that, compared to our simulation policy which solves the manipulation tasks in 20-30 actions, the robot sometimes makes unnecessary manipulation actions. This may be attributed to a combination of a sim-to-real gap, and deficiencies of the robot (e.g. the robot would miss its pick, or its thick gripper would pick up both layers of a folded cloth).

Domains	Random policy	Conditional Pick-Place	Independent / Joint policy	MVP (ours)
Rope	0.34	0.16	0.21	**0.48**
Cloth	0.59	0.34	0.32	**0.84**

TABLE I: Average goal area intersection coverage for rope and cloth spreading tasks on the PR2 robot.

References

[1] Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, et al.

Learning dexterous in-hand manipulation. *arXiv preprint arXiv:1808.00177*, 2018.

[2] Rodney A Brooks. Planning collision-free motions for pick-and-place operations. *The International Journal of Robotics Research*, 2(4):19–44, 1983.

[3] Sachin Chitta, Ioan Sucan, and Steve Cousins. Moveit![ros topics]. *IEEE Robotics & Automation Magazine*, 19(1):18–19, 2012.

[4] Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. In *International Conference on Machine Learning*, pages 1329–1338, 2016.

[5] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[6] Abhinav Gupta, Adithyavairavan Murali, Dhiraj Prakashchand Gandhi, and Lerrel Pinto. Robot learning in homes: Improving generalization and reducing dataset bias. In *Advances in Neural Information Processing Systems*, pages 9094–9104, 2018.

[7] Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018.

[8] Dominik Henrich and Heinz Wörn. *Robot manipulation of deformable objects*. Springer Science & Business Media, 2012.

[9] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *JMLR*, 2016.

[10] Sergey Levine, Peter Pastor, Alex Krizhevsky, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *ISER*, 2016.

[11] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[12] Tomás Lozano-Pérez, Joseph L. Jones, Emmanuel Mazer, and Patrick A. O'Donnell. Task-level planning of pick-and-place robot motions. *Computer*, 22(3):21–29, 1989.

[13] J. Mahler, F. T. Pokorny, B. Hou, M. Roderick, M. Laskey, M. Aubry, K. Kohlhoff, T. Kröger, J. Kuffner, and K. Goldberg. Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards. In *ICRA*, 2016.

[14] Jeremy Maitin-Shepard, Marco Cusumano-Towner, Jinna Lei, and Pieter Abbeel. Cloth grasp point detection based on multiple-view geometric cues with application to robotic towel folding. In *2010 IEEE International Conference on Robotics and Automation*, pages 2308–2315. IEEE, 2010.

[15] Jan Matas, Stephen James, and Andrew J Davison. Sim-to-real reinforcement learning for deformable object manipulation. *arXiv preprint arXiv:1806.07851*, 2018.

[16] Lerrel Pinto and Abhinav Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. *ICRA*, 2016.

[17] Lerrel Pinto, Marcin Andrychowicz, Peter Welinder, Wojciech Zaremba, and Pieter Abbeel. Asymmetric actor critic for image-based robot learning. *arXiv preprint arXiv:1710.06542*, 2017.

[18] Fereshteh Sadeghi and Sergey Levine. Cad2rl: Real single-image flight without a single real image. *arXiv preprint arXiv:1611.04201*, 2016.

[19] Daniel Seita, Nawid Jamali, Michael Laskey, Ajay Kumar Tanwani, Ron Berenstein, Prakash Baskaran, Soshi Iba, John Canny, and Ken Goldberg. Deep transfer learning of pick points on fabric for robot bed-making. *arXiv preprint arXiv:1809.09810*, 2018.

[20] Karun B Shimoga. Robot grasp synthesis algorithms: A survey. *The International Journal of Robotics Research*, 15(3):230–266, 1996.

[21] Jan Stria, Daniel Prusa, Vaclav Hlavac, Libor Wagner, Vladimir Petrik, Pavel Krsek, and Vladimir Smutny. Garment perception and its folding using a dual-arm robot. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 61–67. IEEE, 2014.

[22] Josh Tobin, Lukas Biewald, Rocky Duan, Marcin Andrychowicz, Ankur Handa, Vikash Kumar, Bob McGrew, Alex Ray, Jonas Schneider, Peter Welinder, et al. Domain randomization and generative models for robotic grasping. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3482–3489. IEEE, 2018.

[23] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012.

[24] Takahiro Wada, Shinichi Hirai, Sadao Kawamura, and Norimasa Kamiji. Robust manipulation of deformable objects by a simple pid feedback. In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, volume 1, pages 85–90. IEEE, 2001.

[25] Hanna Yousef, Mehdi Boukallel, and Kaspar Althoefer. Tactile sensing for dexterous in-hand manipulation in robotics—a review. *Sensors and Actuators A: physical*, 167(2):171–187, 2011.