Distilling Motion-Planner Augmented Policies into Visual Control Policies for Robot Manipulation

I-Chun Arthur Liu*, Shagun Uppal*, Gaurav S. Sukhatme, Joseph J. Lim, Peter Englert, Youngwoon Lee

USC University of Southern California

Idea

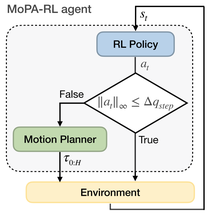
Task: To solve manipulation tasks in obstructed environments by **distilling** a state-based policy into a **visual** policy, with high sample-efficiency and sim2sim transfer capabilities.

Assumptions: The environment dynamics in our observation space are known during training.

Idea: Distill a motion-planner augmented state-based policy into a visual policy removing dependency on motion-planner and environment state using:

- *BC Trajectory Smoothing:* smoothing jittery motion planning trajectories for consistent paths
- *Weight Initialization:* Leveraging past experiences, and improving the sample-efficiency when learning from pixels
- *Entropy coefficient tuning:* Maintaining exploration v/s explitation tradeoff optimally

Background



Manipulation tasks in obstructed environment(MoPA-RL [1]):

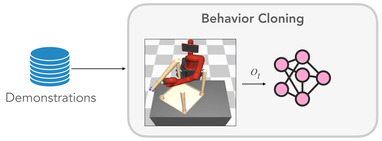
Augments the action space of an RL agent with long-horizon planning capabilities of motion planners.

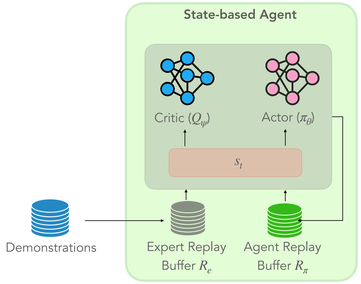
+ Learn contact-rich manipulation skills using RL
+ Plan collision-free paths using motion planner
- Depends on environment state
- Computationally expensive motion planning

Behavioral Cloning [2,3]:

- Motion Planning -> Neural Motion Planning

- Limited by the expert's performance (supervised learning)



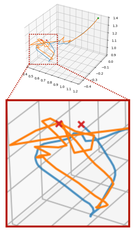


Learning from Demonstrations [4, 5]:

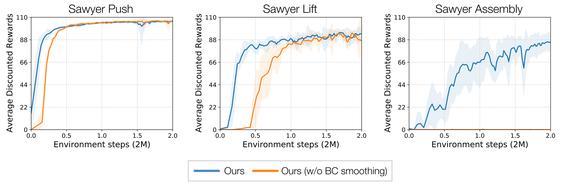
- Separate replay buffer for expert demonstrations

- Optimal trajectories -> guided exploration

BC Trajectory Smoothing

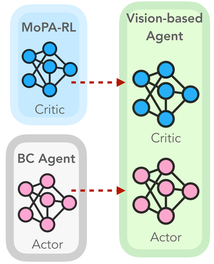


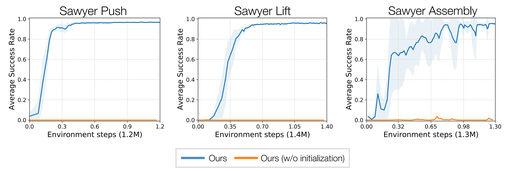
MoPA-RL trajectory vs.
BC smoothed trajectory



- BC Smoothing: circumvents jittery motion-planner paths
- Learning consistent motions, hard for RL agent without BC Smoothing

Weight Initialization





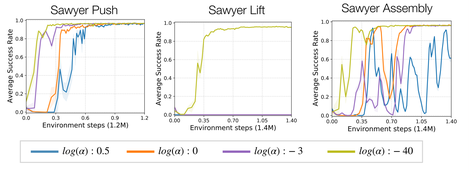
- Improves sample-efficiency for RL over pixels
- Leverages past experiences of state-based agent and BC agent for efficient distillation

Entropy Temperature

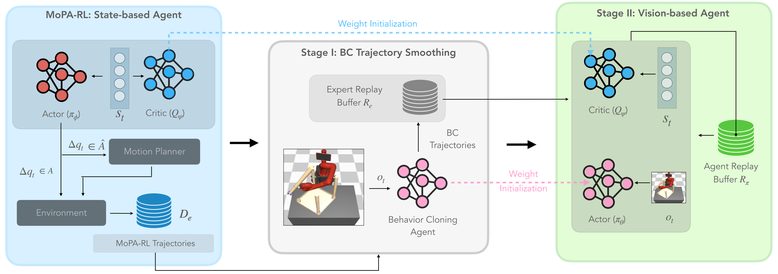
$$J(\pi) := \sum_{t=0}^{T} \gamma^t \mathbb{E}_{\mathbf{s}_t, \mathbf{a}_t \sim \rho_\pi} \left[r(\mathbf{s}_t, \mathbf{a}_t) + \alpha \mathcal{H}(\pi(\cdot|\mathbf{s}_t)) \right]$$

entropy temperature tuning

- α : Initialized to values lower than the final α of our state-based agent.
- Smaller α leverages prior experiences collected using MoPA-RL, improving smaple-efficiency.



Approach

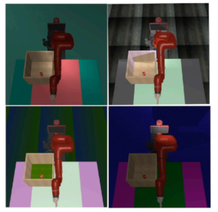


Stage 1

- Collect transitions from our state-based agent MoPA-RL policy wth actions in the direct action space in a dataset.
- Train a BC policy using the MoPA-RL dataset and collect BC-smoothed trajectories a separate replay buffer with optimal trajectories.

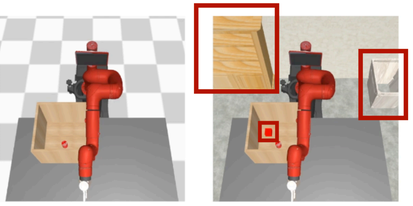
Stage 2

- Train an asymmetric actor-critic agent, where the actor is learnt with image observations, and the critic using environment states.
- Leverage prior experience collected by state-based agent and initialize critic weights with the MoPA-RL critic and actor with BC agent.



Training Environments

Domain Randomization for sim-to-sim transfer



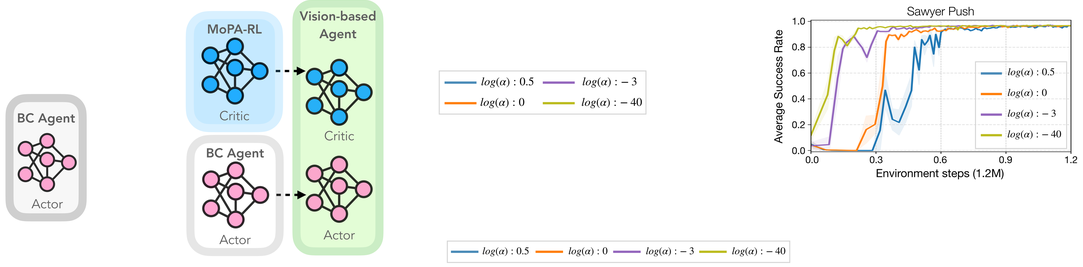
Original Environment

Environment with **Distractors**

Testing Environments (unseen during training)

References

[1] J. Yamada, Y. Lee, G. Salhotra, K. Pertsch, M. Pflueger, G. S. Sukhatme, J. J. Lim, and P. Englert. Motion planner augmented reinforcement learning for robot manipulation in obstructed environments. In Conference on Robot Learning, 2020.
[2] T. Jurgenson and A. Tamar. Harnessing reinforcement learning for neural motion planning. In Proceedings of Robotics: Science and Systems, June 2019.
[3] A. H. Qureshi, M. J. Bency, and M. C. Yip. Motion planning networks. In IEEE International Conference on Robotics and Automation, pages 2118–2124, 2019.
[4] V. G. Goecks, G. Gremillion, V. Lawhern, J. Valasek, and N. R. Waytowich. Integrating behavior cloning and reinforcement learning for improved performance in sparse reward environments. In Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS' 20), page 465–473, 2020.
[5] A. Nair, B. McGrew, M. Andrychowicz, W. Zaremba, and P. Abbeel. Overcoming exploration in reinforcement learning with demonstrations. IEEE International Conference on Robotics and Automation, pages 6292–6299, 2018.


BC Agent
Actor
MoPA-RL
Critic
BC Agent
Actor
Vision-based Agent
Critic
Actor
$log(\alpha): 0.5$
$log(\alpha): 0$
$log(\alpha): -3$
$log(\alpha): -40$
Sawyer Push
Average Success Rate
Environment steps (1.2M)
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.3
0.6
0.9
1.2
$log(\alpha): 0.5$
$log(\alpha): 0$
$log(\alpha): -3$
$log(\alpha): -40$
$log(\alpha): 0.5$ $log(\alpha): 0$ $log(\alpha): -3$ $log(\alpha): -40$