Reinforcement and Imitation Learning to Control Contact-Rich Interactions

Roberto Martín-Martín, RobIn Lab, UT Austin

RobIn
ROBOT INTERACTIVE INTELLIGENCE LAB

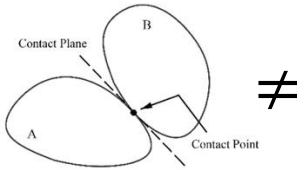
TEXAS Robotics

THE UNIVERSITY OF TEXAS AT AUSTIN

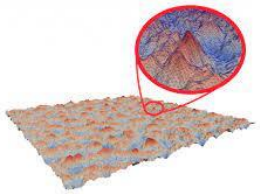

Challenging Contact-Rich Interactions!

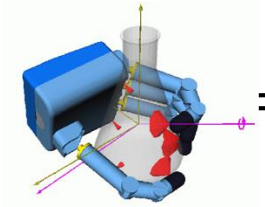
Why is hard to control Contact-Rich Manipulation?

Predicting the effect of contact interactions is difficult

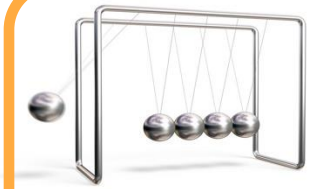


≠

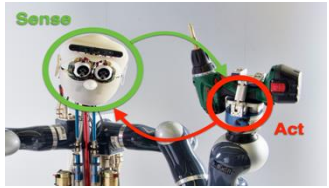




≠



Requires learning how to deal with reactive forces



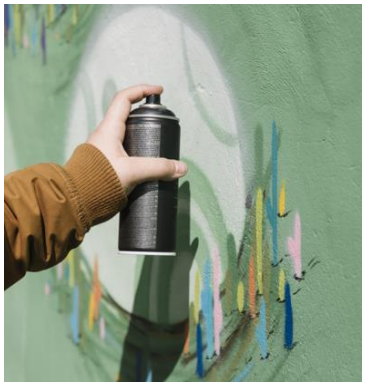
Requires learning feedback loops

is dangerous for robot and environment


Action
Perception

Learning Tasks in Different Action Spaces

Configuration/Joint Space



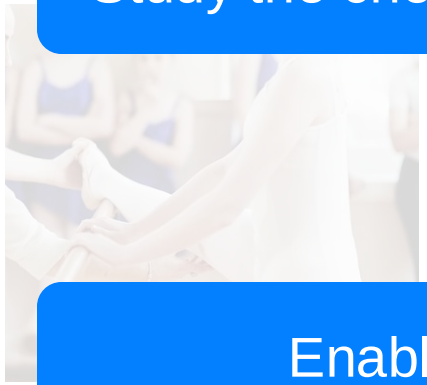
Task/Cartesian Space

Motion

Forces

Learning Tasks in Different Action Spaces

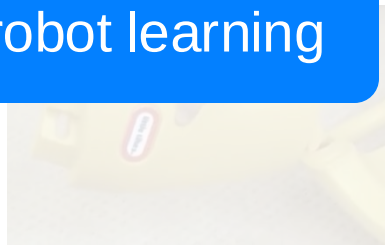
Study the choice of action space affects robot learning

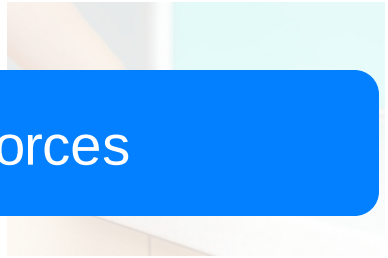


Configuration/Joint Space

Task/Cartesian Space

Enable RL algorithms to control forces

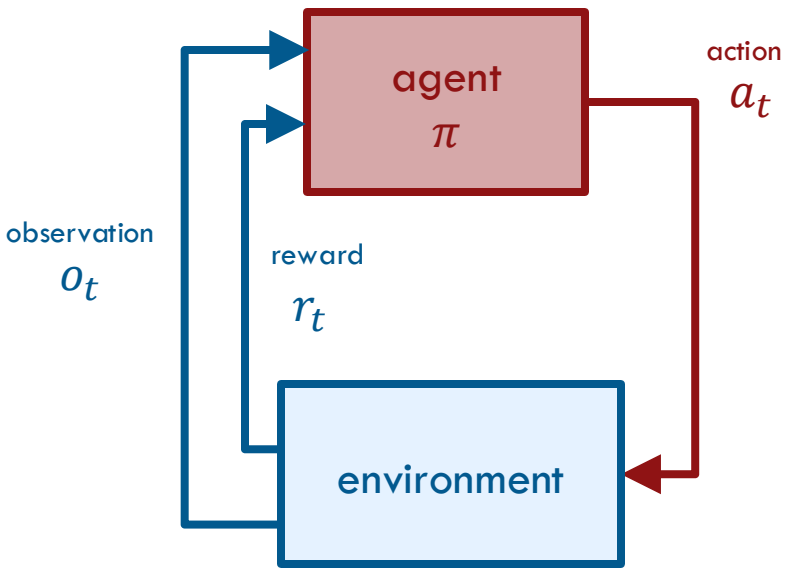




Motion

Forces

Reinforcement learning to control physical interactions



$$\mathcal{M} = \langle S, A, O, P(\cdot,\cdot), R(\cdot,\cdot), \gamma \rangle$$

state space | action space | observation space | transition function | reward function | discount factor

$P: S \times A \to S$ $R: S \times A \to \mathbb{R}$

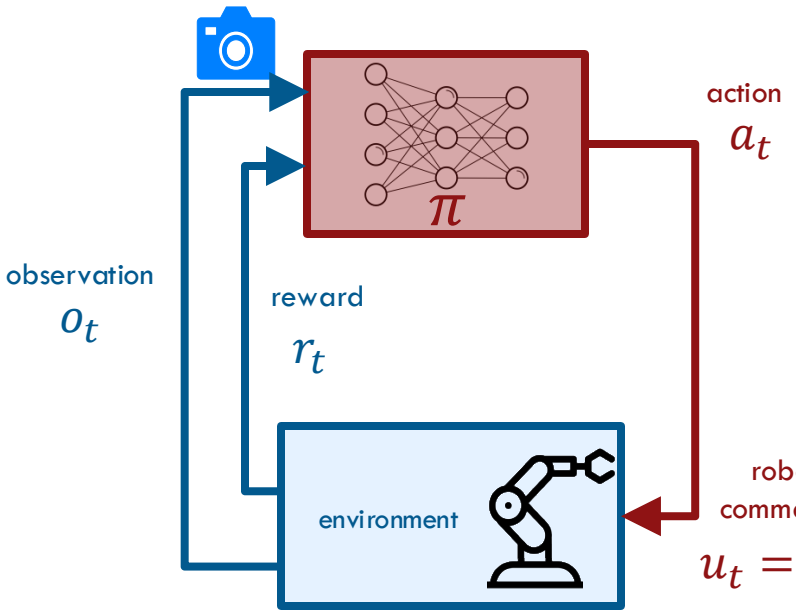
Goal: find policy $a_t = \pi(o_t) \to \mathrm{p}(\mathrm{a_t}|\mathrm{o_t})$

that maximizes $\mathbb{E}_{p_\pi(\tau)}[T(\tau)]$, $T(\tau) = \sum_{(o_t,a_t,r_t)\in\tau} \gamma^t r_t$

[Mahadevan & Connell 1992; Gullapalli et al. 1994; Atkeson 1994; Schaal 1996; Bagnell & Schneider 2001; Smart & Kaelbling 2002; Kohl & Stone 2004; Kober & Peters 2009; Theodorou et al. 2010; Levine et al. 2015; Gu et al. 2016; Peng et al. 2017; Kalashnikov et al. 2018]

Deep reinforcement learning to control physical interactions

End-to-end learning from pixel to torques



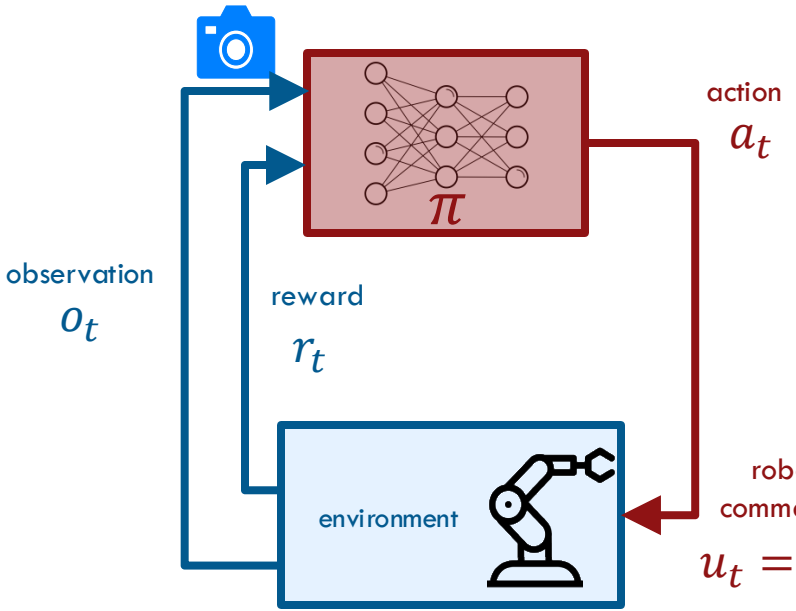
allows to use images directly as input

training and test conditions must be the same

[Agrawal et al. '16, Levine et al. '16, Peng et al. '17, Gu et al. '16, Chebotar et al. '17, Yahya et al. '16, James et al. '17, Popov et al. '17]

Deep reinforcement learning to control physical interactions

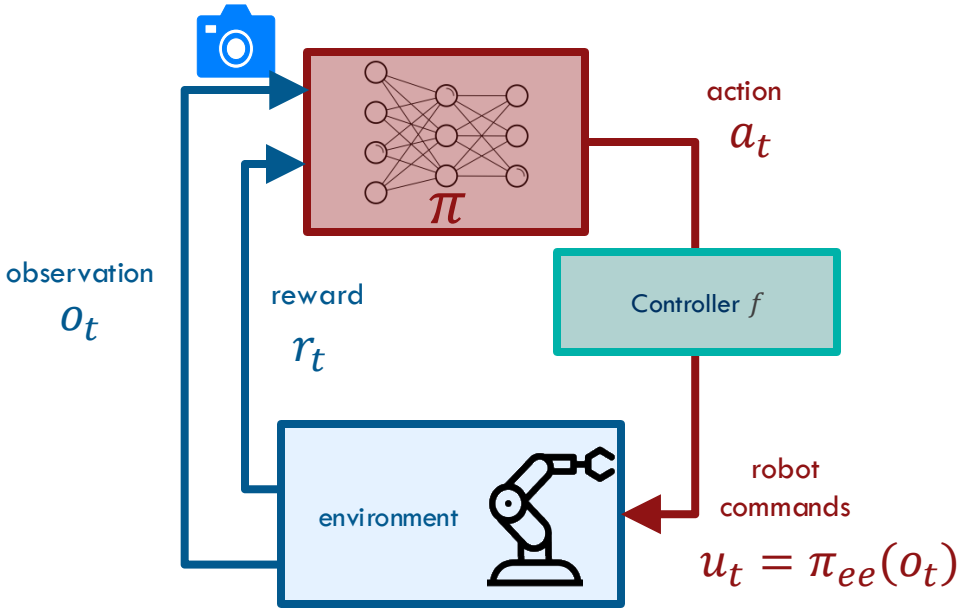
End-to-end learning from pixel to torques



- allows to use images directly as input
- training and test conditions must be the same
- data hungry: re-learn how to control robot's end-effector motion and forces for every task

[Agrawal et al. '16, Levine et al. '16, Peng et al. '17, Gu et al. '16, Chebotar et al. '17, Yahya et al. '16, James et al. '17, Popov et al. '17]

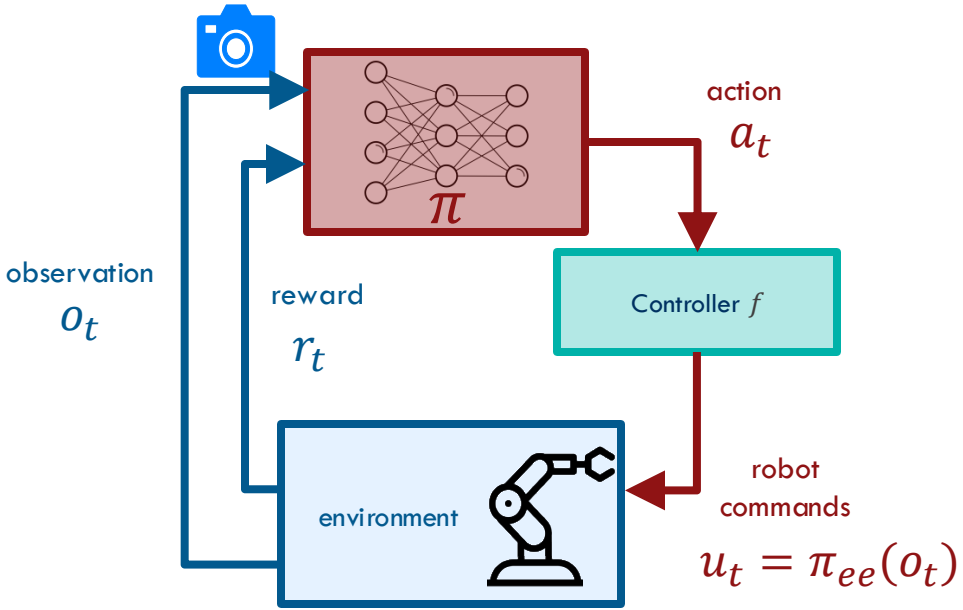
RL with Analytical Controller



$$a_t = \pi(o_t)$$

[Khatib.1987], [Bruyninckx et al. , '96], [Schaal '03], [Rodriguez et al. 2012], [Vijaykumar et al. '11],[Ijspreet et al. '13]][Li, Billard, et al. '14], [Lee, Abbeel et al.

RL with Analytical Controller

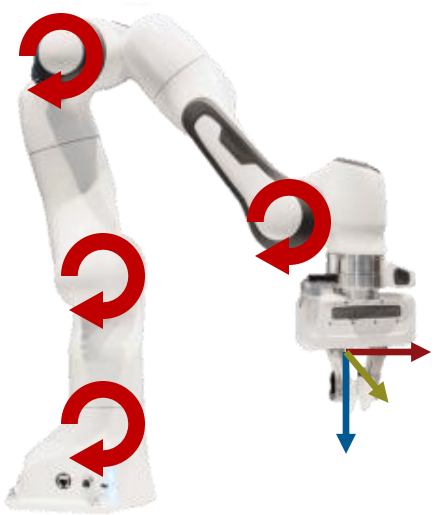


Reference Generator (learned)

$$u_t = f(\pi(o_t))$$

Robot Control (analytic)

A Study of Action Spaces in RL



Joint Space

Cartesian/Task Space

[Martin-Martin, IROS'19]

A Study of Action Spaces in RL

Joint Space

Cartesian/Task Space

Motion: Positions/Velocities/Accelerations

Forces and Torques

[Martin-Martin, IROS'19]

VICES: Variable Impedance in End-Effector Space

VICES

Joint Space

Motion: Positions/Velocities/Accelerations

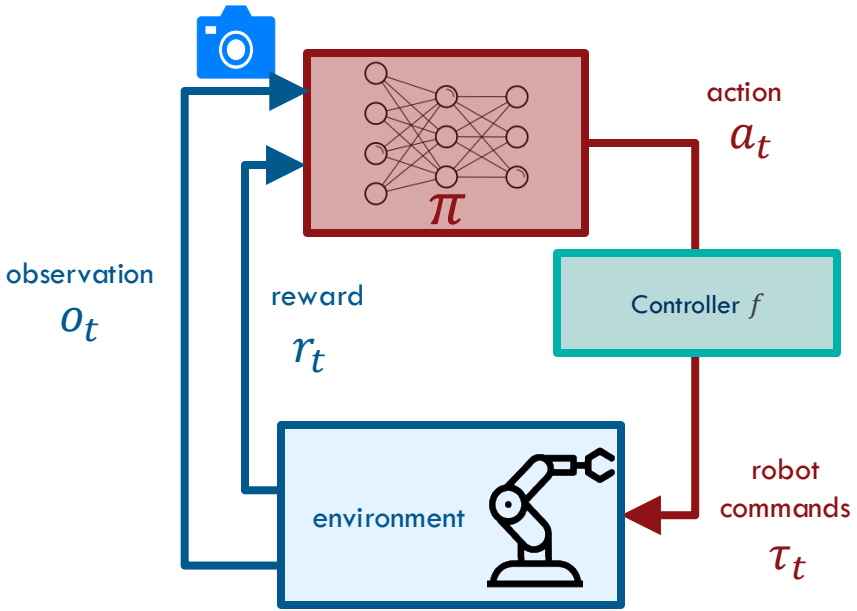
Cartesian/Task Space

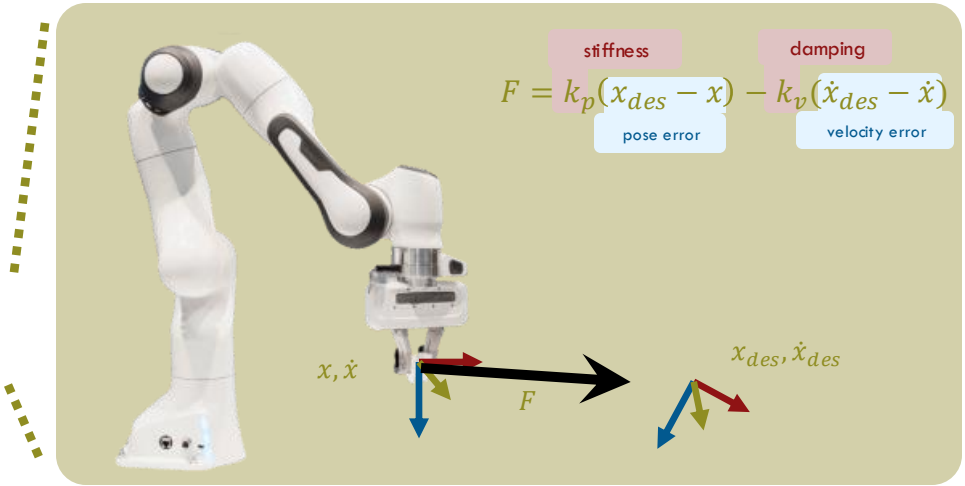
Forces and Torques

[Martin-Martin, IROS'19]

Interaction controller in end-effector space

Direct control of motion and forces of robot's hand

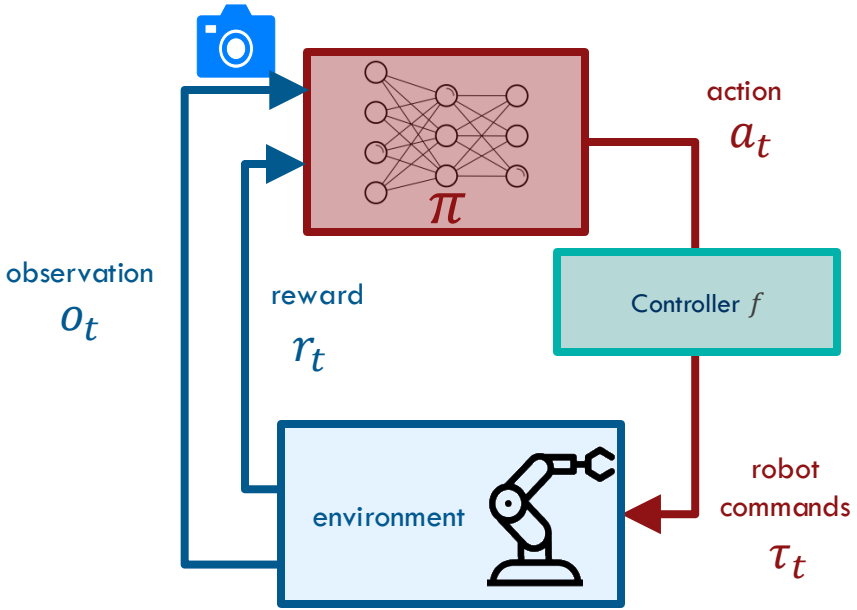


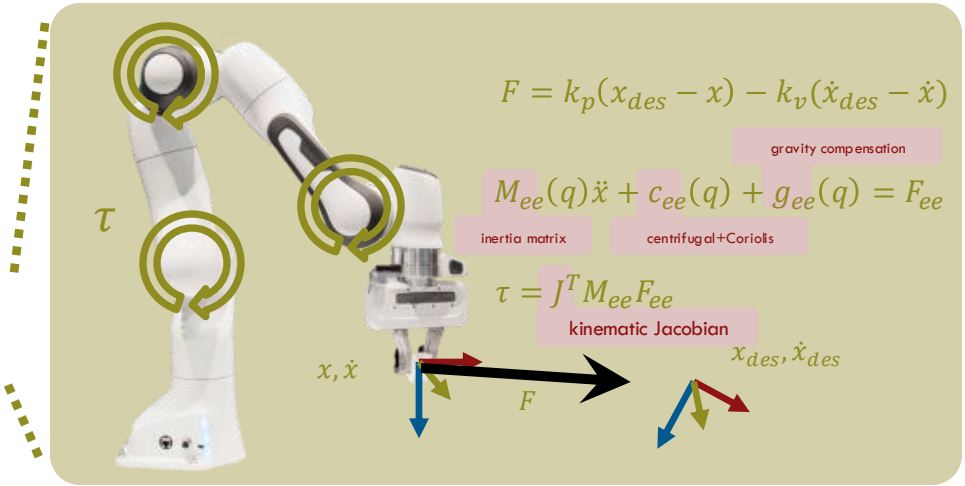


[Khatib '87, Bruyninckx et al. '96, Schaal '03, Rodriguez et al. '12, Vijaykumar et al. '11, Ijspreet et al. '13, Billard et al. '14, Lee, Abbeel et al. '15

Interaction controller in end-effector space

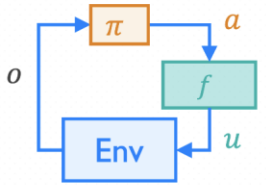
Direct control of motion and forces of robot's hand





[Khatib '87, Bruyninckx et al. '96, Schaal '03, Rodriguez et al. '12, Vijaykumar et al. '11, Ijspreet et al. '13, Billard et al. '14, Lee, Abbeel et al. '15

Studied Action Spaces



$$u = f(\underbrace{\pi(o)}_{a})$$

Action Space	$a = \pi(o) \in \mathcal{A}$	$u = f(a) = \tau$
Joint Torques	$a = \tau_{des}$	$u = a$
Joint Velocities	$a = \dot{q}_{des}$	$u = k_v(a - \dot{q}_{cur})$
Joint Positions	$a = q_{des}$	$u = M_j[k_p(a - q_{cur}) - k_v\dot{q}_{cur}]$
Variable Joint Impedance	$a = (q_{des}, k_p, k_v)$	
End-Effector Pose	$a = x_{des} \in SE(3)$	$u = J^T[M_{ee}[k_p(a - x_{cur}) - k_v\xi_{cur}]]$
VICES	$a = (x_{des}, k_p, k_v)$	

[Martin-Martin, IROS'19]

Experimental Evaluation

- effect of the choice of action space in the performance of RL

Experimental Evaluation

- effect of the choice of action space in the performance of RL
- transferability of policies in different action spaces across robots

Experimental Evaluation

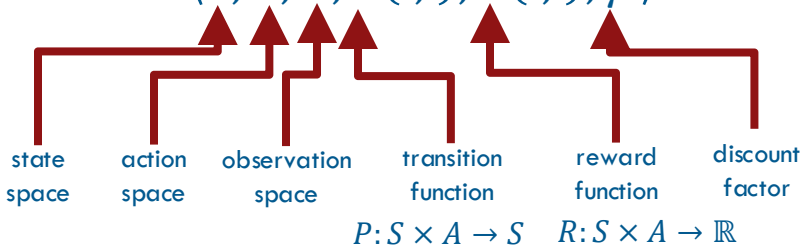
- effect of the choice of action space in the performance of RL
- transferability of policies in different action spaces across robots
- benefits of learning to control forces

Solving MDPs without Models

Model-free Reinforcement learning

When dynamics model is <u>unknown</u> and <u>hard to estimate</u>, we can **learn policy directly** from the agent's trajectories from interacting with an MDP

$$\mathcal{M} = \langle S, A, O, P(\cdot,\cdot), R(\cdot,\cdot), \gamma \rangle$$



agent's experience

$$\tau = \{(s_i, a_i, r_i) \,|\, i = 0, \ldots, H\}$$

model-free RL

optimal policy

$$\pi^* = \arg\max_{\pi} \mathbb{E}[\sum_{t \geq 0} \gamma^t r(s_t, \pi(s_t))]$$

Solving MDPs without Models – Policy Learning

policy parameterized by θ

Total reward of trajectory

Objective Function $J(\theta) = \mathbb{E}_{\tau \sim p(\tau|\theta)}[r(\tau)]$

trajectories under policy π_θ

Model-free Policy-Gradient RL

policy gradient theorem

$$\nabla_\theta J(\theta) = \mathbb{E}_\pi[Q^\pi(s,a)\nabla_\theta \log \pi_\theta(a|s)]$$

Different ways of computing the gradient above lead to different PG variants: Monte-Carlo

Main problems: Data inefficiency and training instability!

Trust Region Methods

TRPO, Trust Region Policy Optimization

- Intuition: the new policy (after update) should not be "too different" from the previous one → Stability
  - TRPO measures how different two policies are by computing the KL divergence and imposing it as hard constraint to the optimization
- Additionally, we can reuse the experiences from the old policy, allowing us to reuse data! → Data efficiency

π_θ →

$$\theta_{k+1} = \arg\max_\theta \mathcal{L}(\theta_k, \theta)$$
$$\text{s.t. } \bar{D}_{KL}(\theta||\theta_k) \leq \delta$$

$$\mathcal{L}(\theta_k, \theta) = \mathop{\mathrm{E}}_{s,a\sim\pi_{\theta_k}}\left[\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)} A^{\pi_{\theta_k}}(s,a)\right]$$

$$D_{KL}(\theta||\theta_k) = \mathop{\mathrm{E}}_{s\sim\pi_{\theta_k}}\left[D_{KL}\left(\pi_\theta(\cdot|s)||\pi_{\theta_k}(\cdot|s)\right)\right]$$

New problem: Computationally expensive

Proximal Policy Optimization

- Convert the constrained optimization problem into a simpler one
  - Use a “clipped” surrogate objective:

$$L(s, a, \theta_k, \theta) = \min\left(\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)} A^{\pi_{\theta_k}}(s, a),\ \ \text{clip}\left(\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)}, 1-\epsilon, 1+\epsilon\right) A^{\pi_{\theta_k}}(s, a)\right)$$

- Implicit constraint:

$$(1-\epsilon)\pi_{\theta_{\text{old}}} \leq \pi_\theta \leq (1+\epsilon)\pi_{\theta_{\text{old}}}$$

Improved Stability!

Experimental Setup

Embodiments

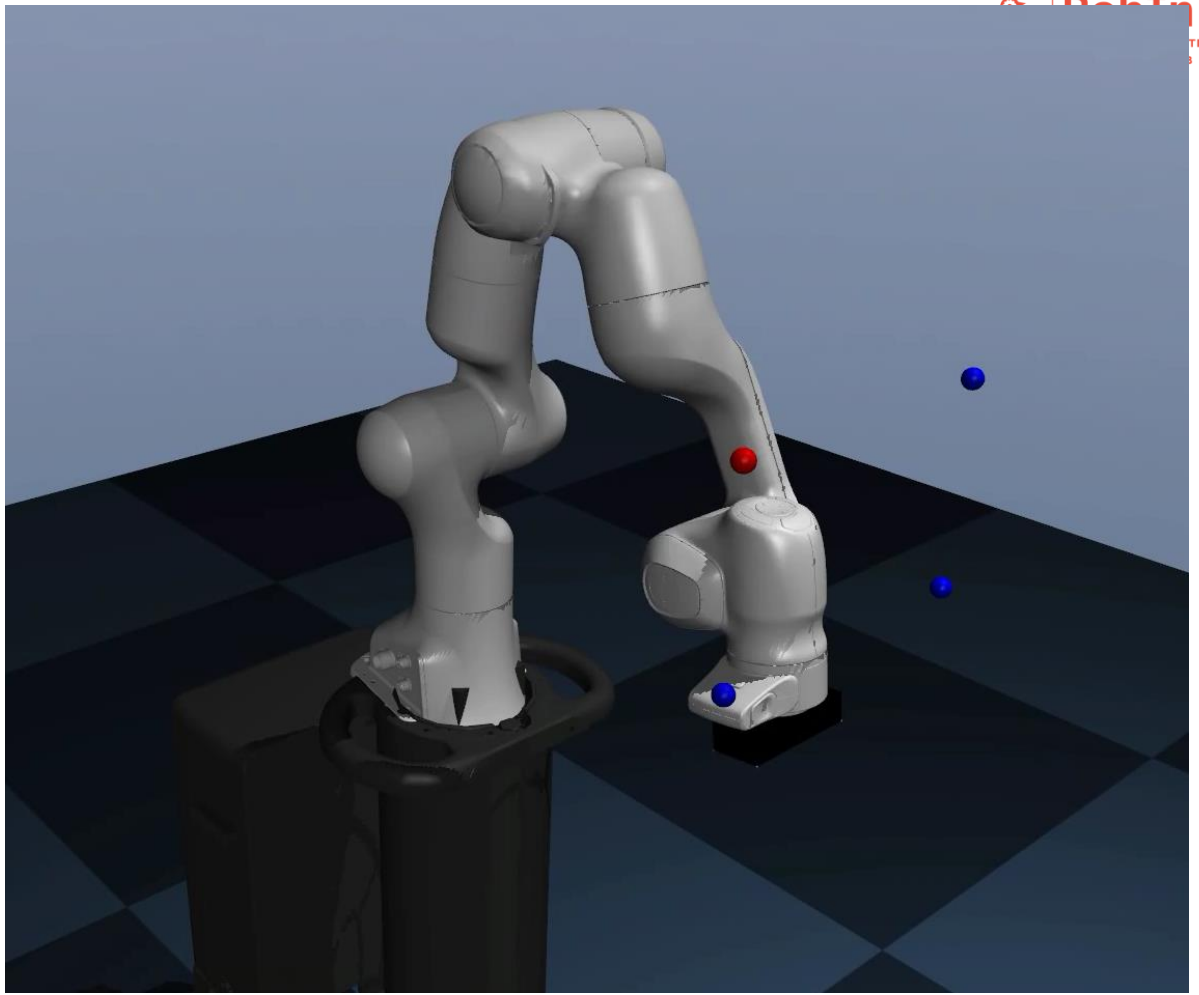
[Martin-Martin, IROS'19]

Path Following

No Contact

Input: Proprioception

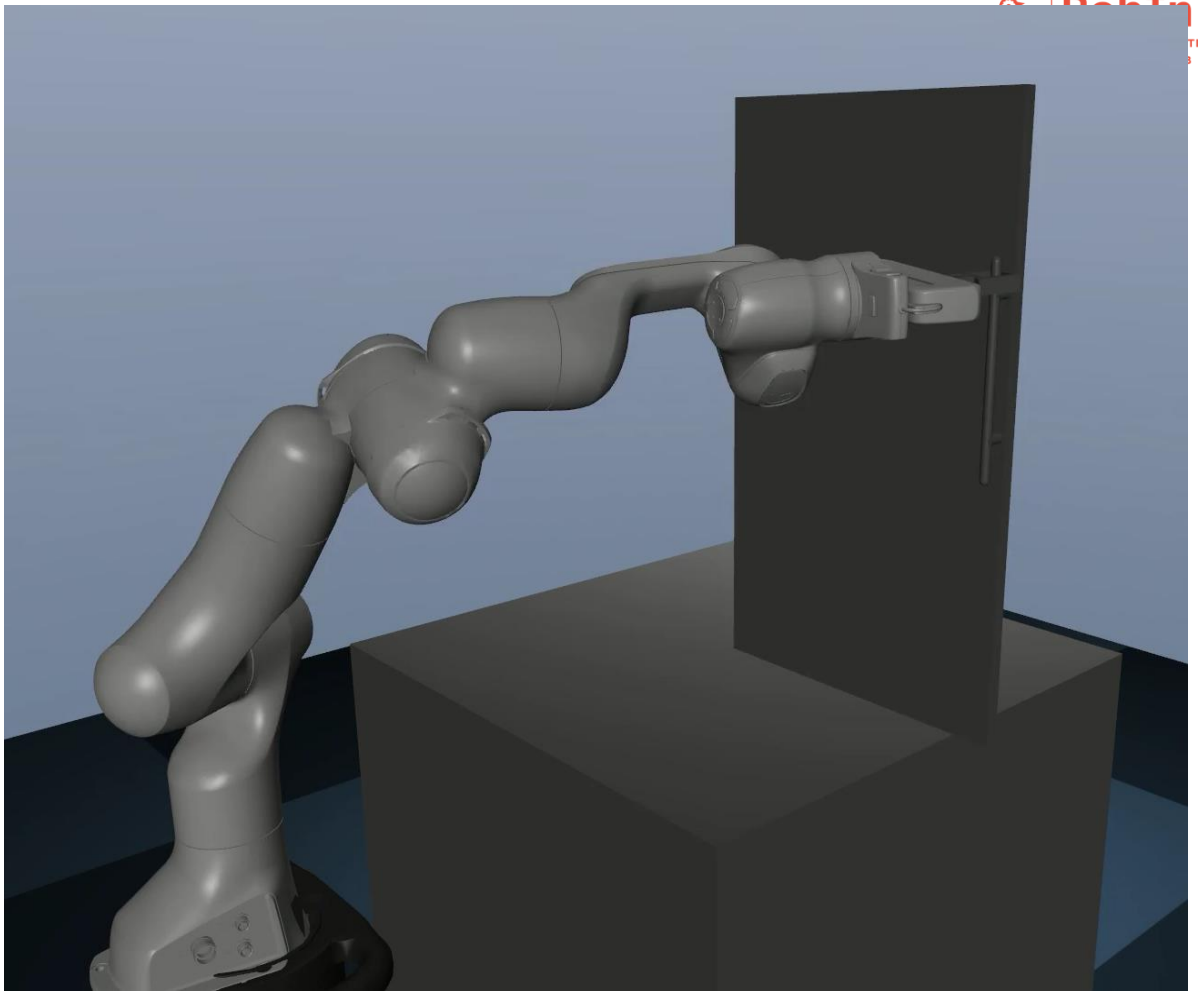
Trajectory State



[Martin-Martin, IROS'19]

Door Opening

Intermittent Contact
Input: Proprioception and
Door state

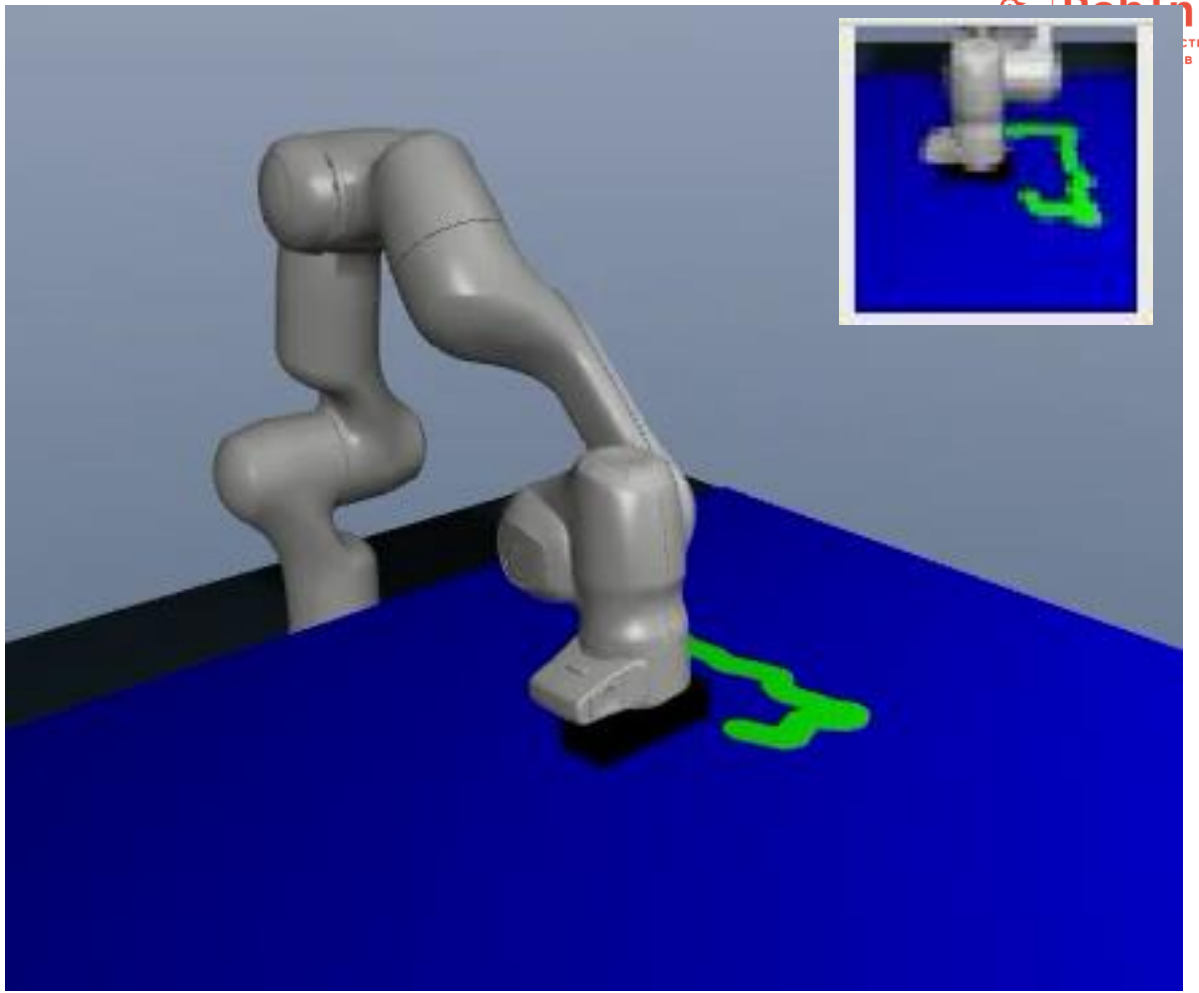


[Martin-Martin, IROS'19]

Surface Wiping

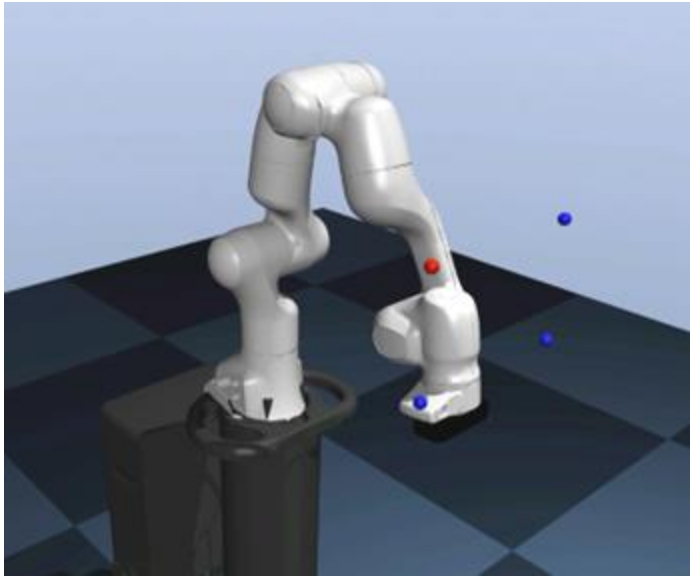
Contact Rich

Input: Proprioception and Images

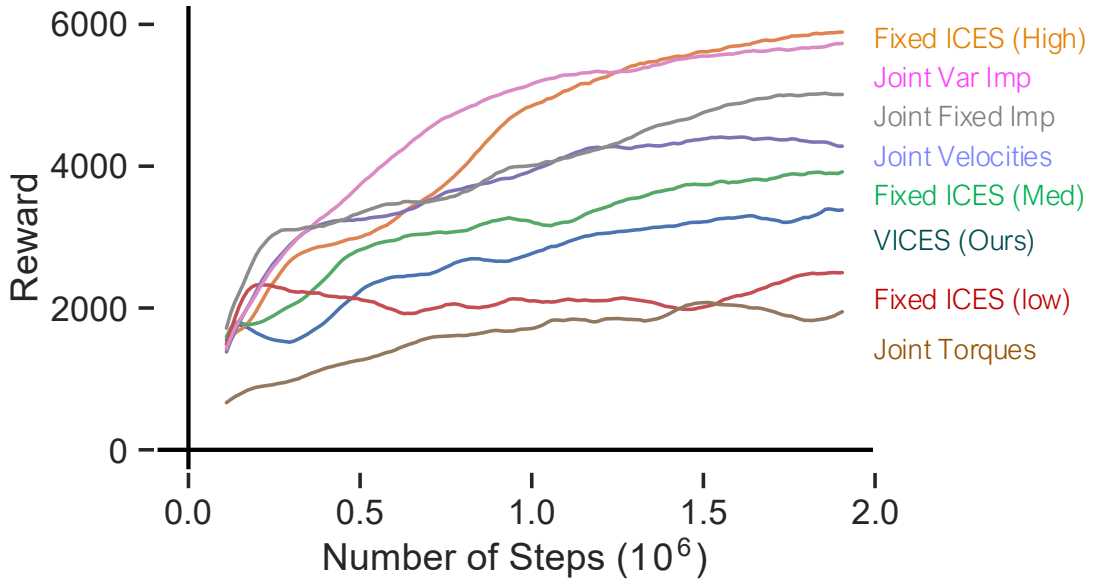


[Martin-Martin, IROS'19]

Sample Efficiency: Path Following

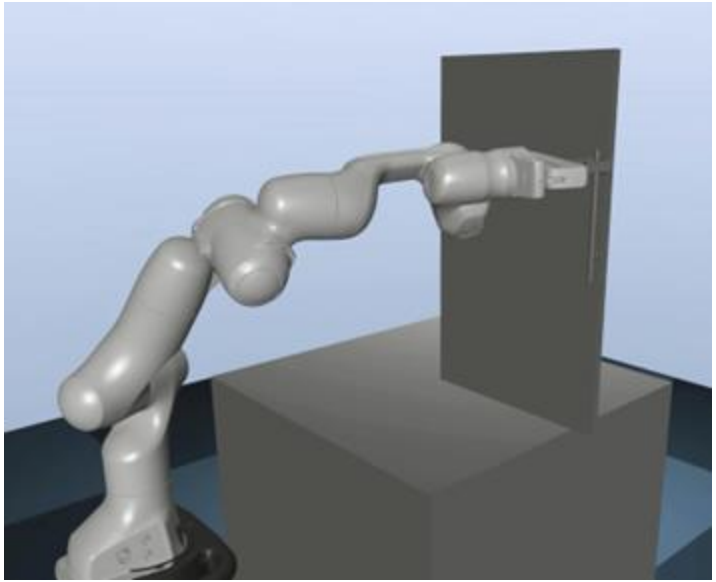


Trained Policy Rollout (VICES, Ours)

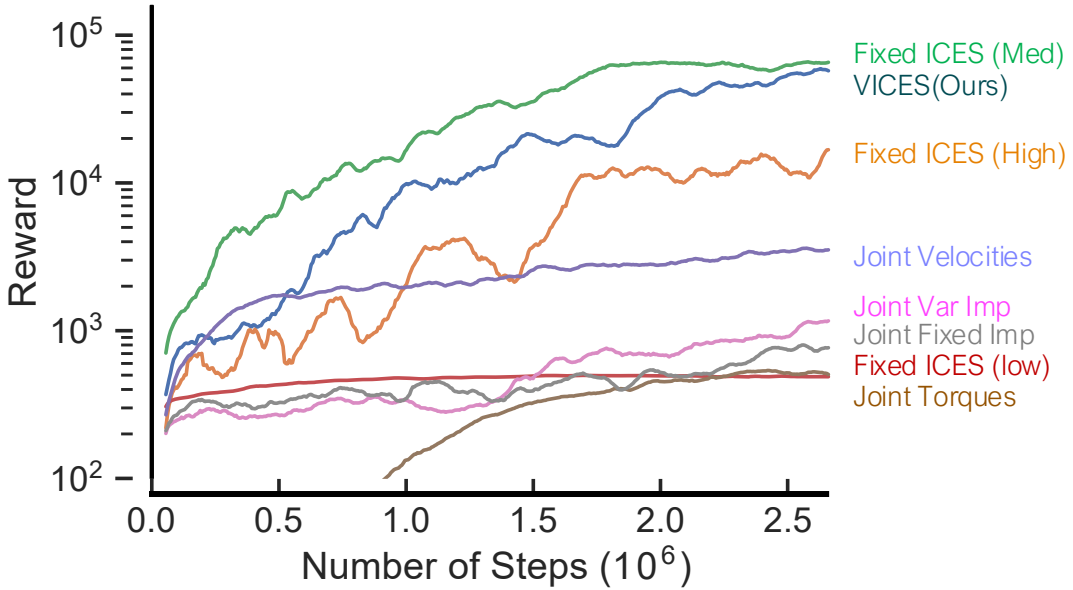


[Martin-Martin, IROS'19]

Sample Efficiency: Door Opening

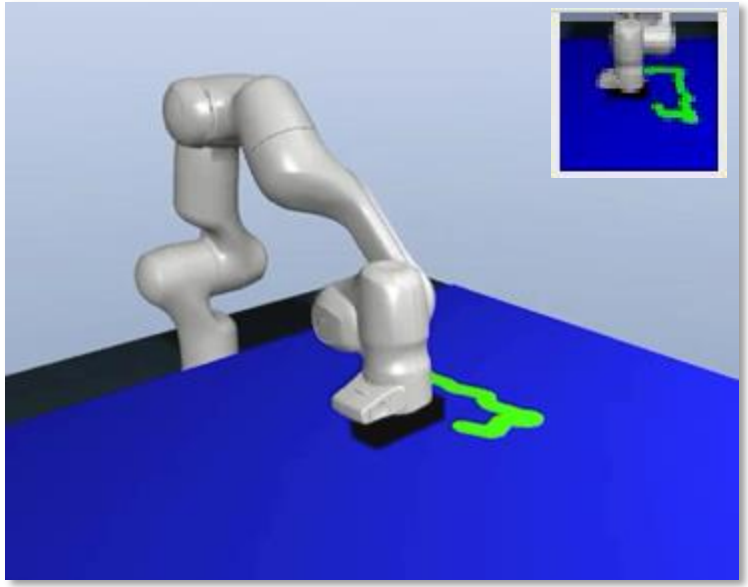


Trained Policy Rollout (VICES, Ours)

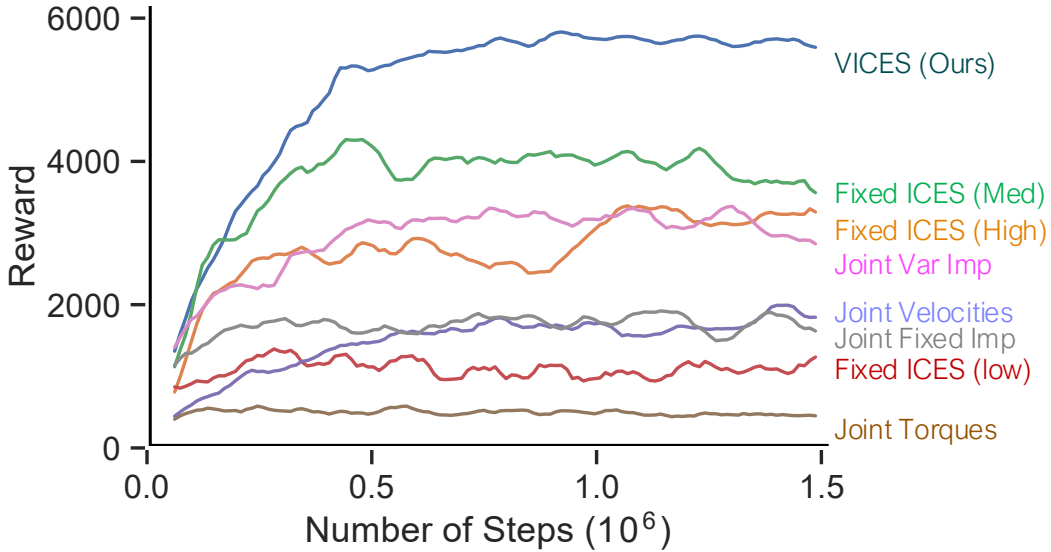


[Martin-Martin, IROS'19]

Sample Efficiency: Surface Wiping



Trained Policy Rollout (VICES,Ours)

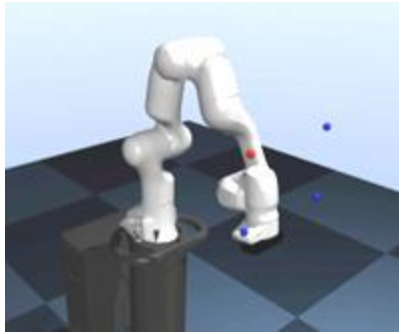


[Martin-Martin, IROS'19]

Transfer in Sim: Path Following

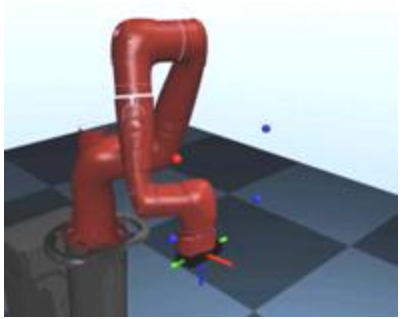
Trained on Panda

$u = f_{panda}(\pi_{panda}(o_t))$

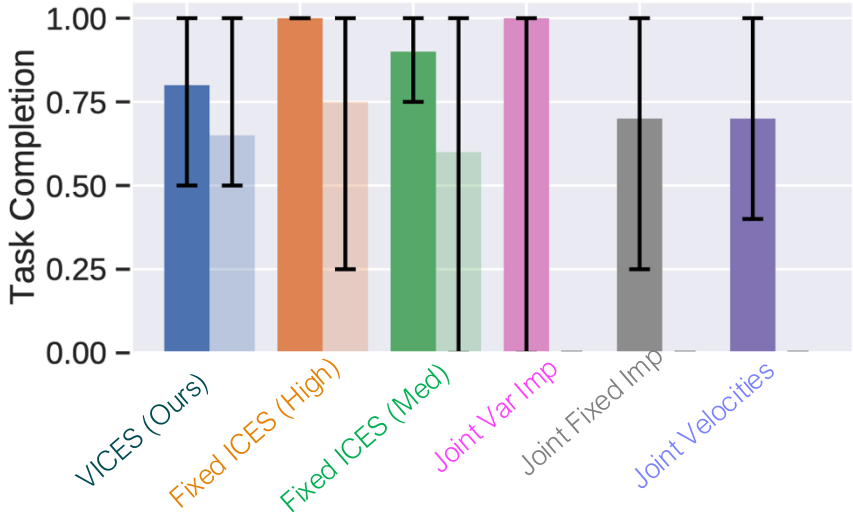


Tested on Sawyer

$u = f_{sawyer}(\pi_{panda}(o_t))$



Trained Policy Rollout (VICES,Ours)

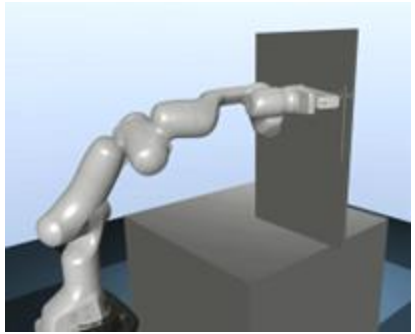


[Martin-Martin, IROS'19]

Transfer in Sim: Door Opening

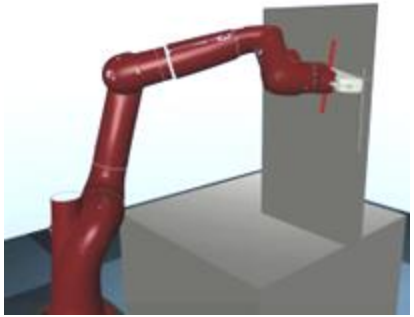
Trained on Panda

$u = f_{panda}(\pi_{panda}(o_t))$

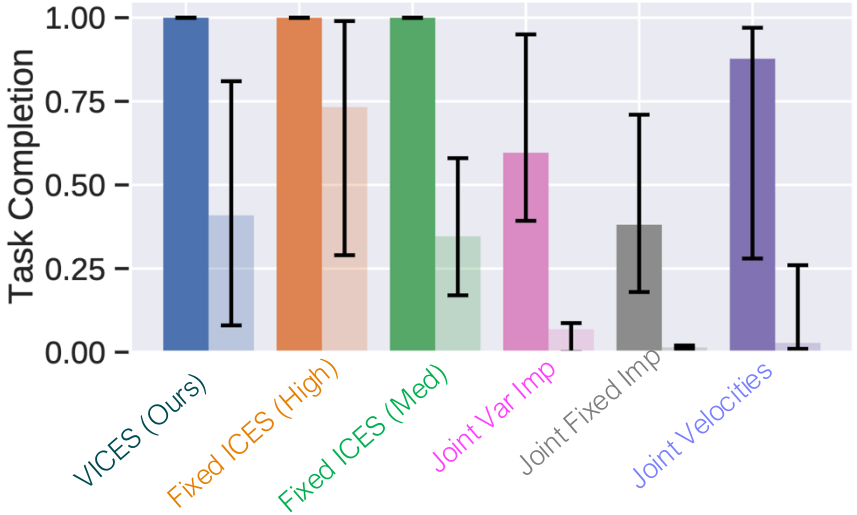


Tested on Sawyer

$u = f_{sawyer}(\pi_{panda}(o_t))$



Trained Policy Rollout (VICES,Ours)

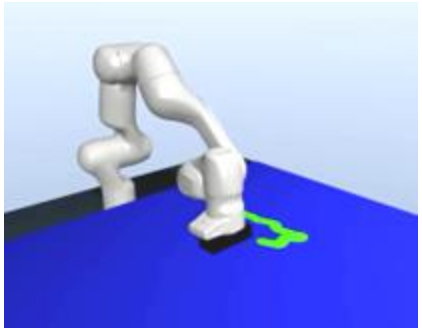


[Martin-Martin, IROS'19]

Transfer in Sim: Surface Wiping

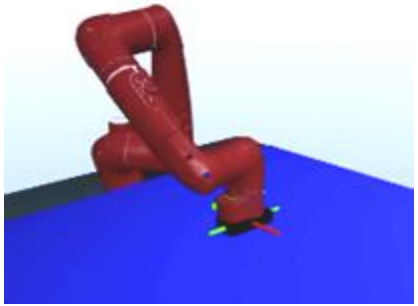
Trained on Panda

$u = f_{panda}(\pi_{panda}(o_t))$

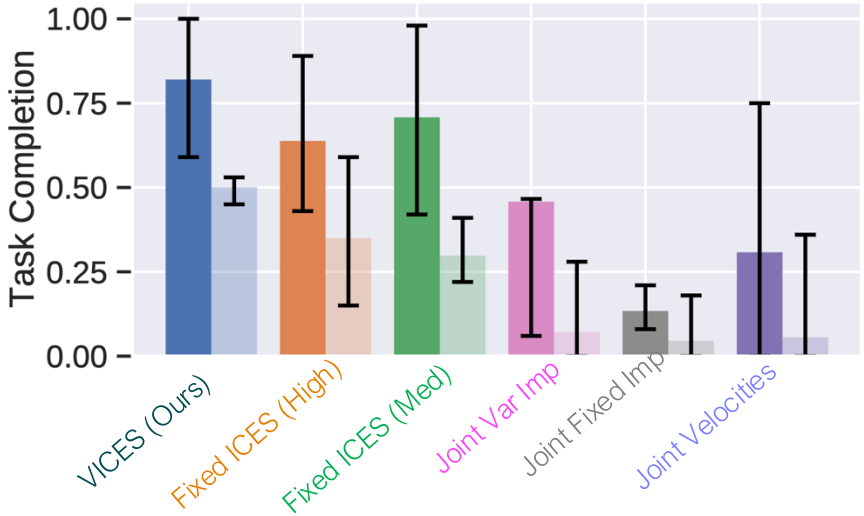


Tested on Sawyer

$u = f_{sawyer}(\pi_{panda}(o_t))$



Trained Policy Rollout (VICES,Ours)

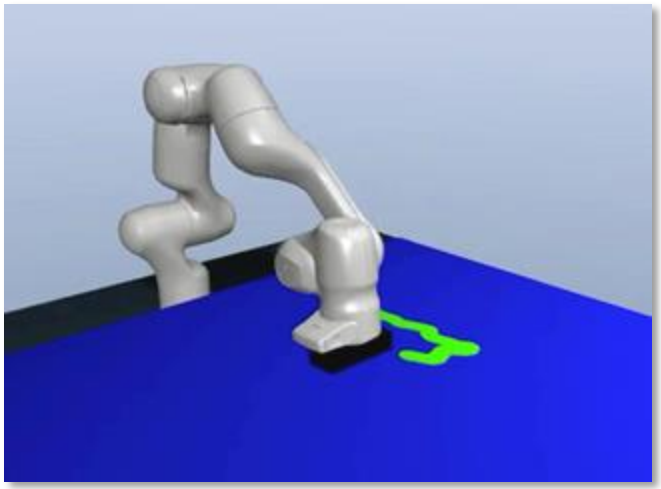


[Martin-Martin, IROS'19]

Transfer to Real: Surface Wiping

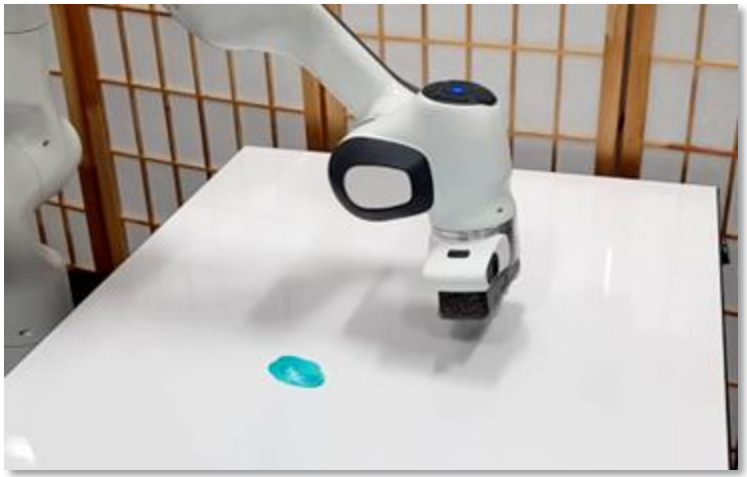
Trained on Simulation

$u = f_{sim}(\pi_{sim}(o_t))$



Tested on Real World

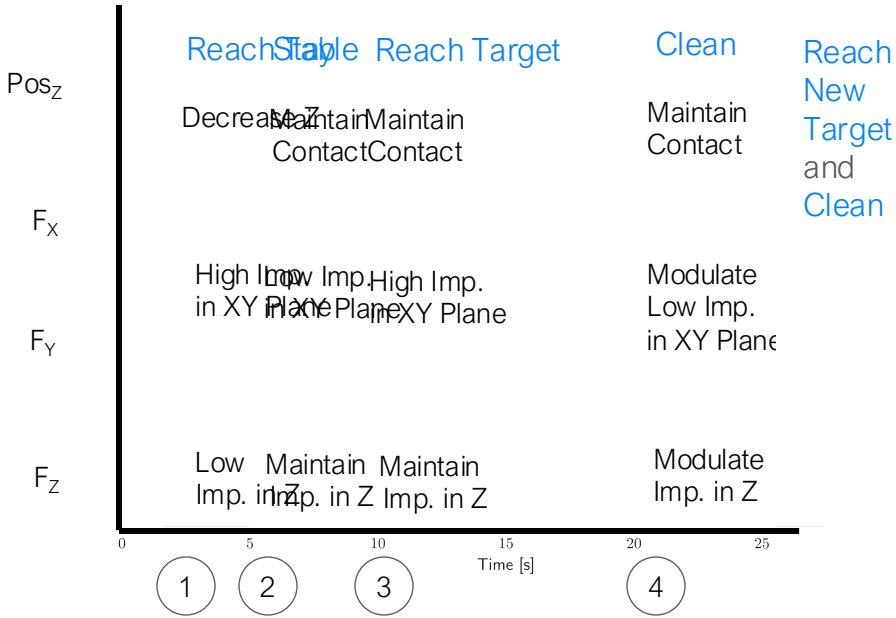
$u = f_{real}(\pi_{sim}(o_t))$

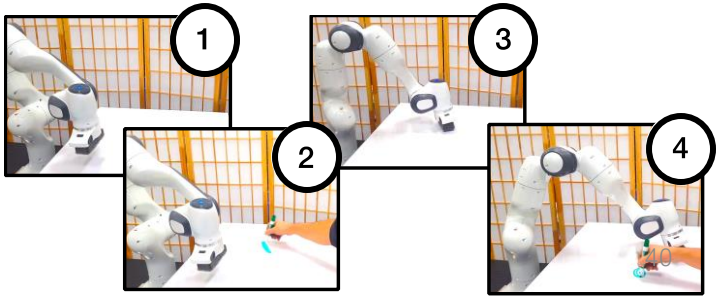


Success 80% (10 Trials)

[Martin-Martin, IROS'19]

Varying Impedance for Wiping





Key Takeaways

- the action space of the RL policy is crucial (more than the algorithm)

[Martin-Martin, IROS'19]

Key Takeaways

- the action space of the RL policy is crucial (more than the algorithm)
- transfer between robots is easier in task-space

[Martin-Martin, IROS'19]

Key Takeaways

- the action space of the RL policy is crucial (more than the algorithm)
- transfer between robots is easier in task-space
- in contact-rich tasks the policy needs to control forces → VICES

[Martin-Martin, IROS'19]

Robotics: **Representations**

"Solving a problem simply means representing it so as to make the solution transparent."

Herbert A. Simon, Sciences of the Artificial

Usually referring to the state representation but the action representation is key!

Learning action spaces?

LASER: Learning a Latent Action Space for Efficient Reinforcement Learning

Arthur Allshire*†, Roberto Martín-Martín*‡, Charles Lin‡, Shawn Manuel‡, Silvio Savarese‡, Animesh Garg†◇

***Abstract*— The process of learning a manipulation task depends strongly on the action space used for exploration: posed in the incorrect action space, solving a task with reinforcement learning can be drastically inefficient. Additionally, similar tasks or instances of the same task family impose latent manifold constraints on the most effective action space: the task family can be best solved with actions in a manifold of the entire action space of the robot. Combining these insights we present LASER, a method to learn latent action spaces for efficient reinforcement learning. LASER factorizes the learning problem into two sub-problems, namely action space learning and policy learning in the new action space. It leverages data from similar manipulation task instances, either from an offline expert or online during policy learning, and learns from these trajectories a mapping from the original to a latent action space. LASER is trained as a variational encoder-decoder model to map raw actions into a disentangled latent action space while maintaining action reconstruction and latent space dynamic consistency. We evaluate LASER on two contact-rich robotic tasks in simulation, and analyze the benefit of policy learning in the generated latent action space. We show improved sample efficiency compared to the original action space from better alignment of the action space to the task space, as we observe with visualizations of the learned action space manifold. Additional details: pair.toronto.edu/laser**

I. Introduction

Deep Reinforcement Learning (RL) has fueled rapid progress in robot manipulation by enabling learning of closed loop visuomotor control policies that integrate perception and control in a single system [1]. However, the focus of end-to-end policy learning has been on the complexity of

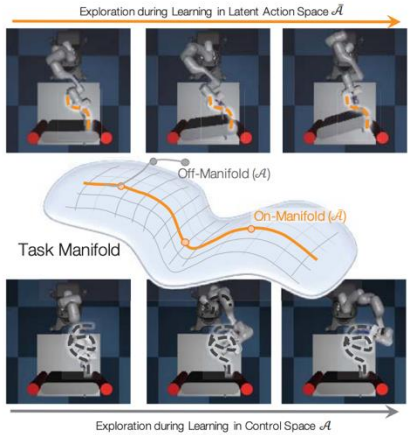
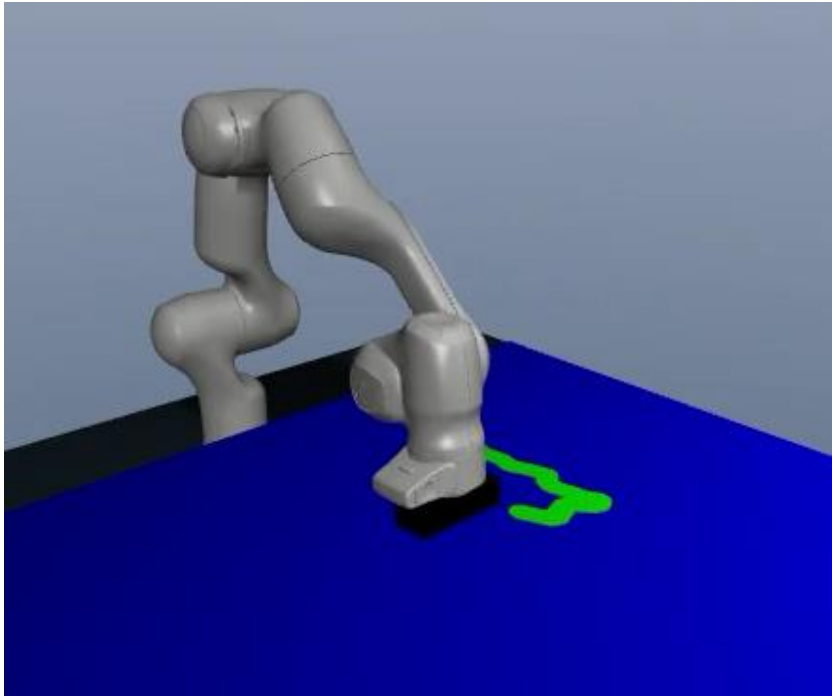


Fig. 1: **Learning Latent action spaces for efficient reinforcement learning**. Manipulation tasks, such as opening a door, are often structured and do not require exploration in the entire action space, only on certain manifold. LASER learns this action space manifold from data, either offline (expert) or online (training with LASER), enabling faster learning in subsequent novel instances of the task by transferring the knowledge via an efficient latent action space.

- Can we directly learn good action spaces?

Next Step: Generalizing Contact-Rich Manipulation

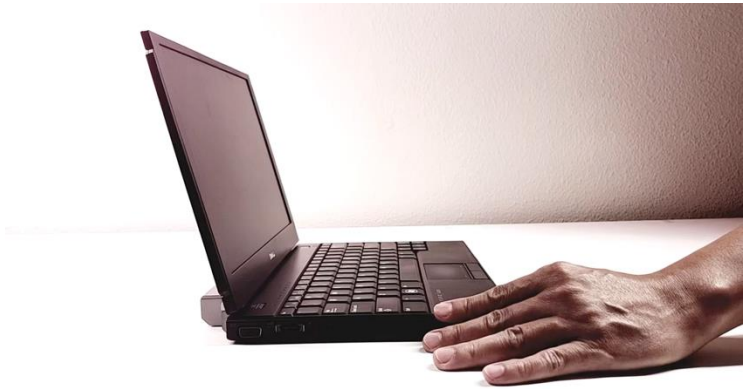


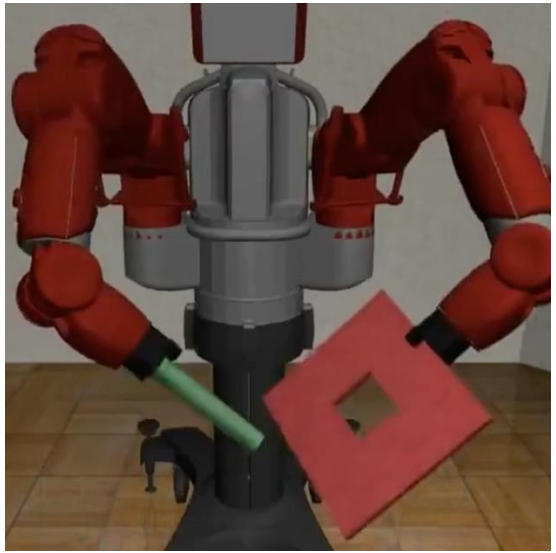
• • •


wikiHow


www.jamieshomecookingskills.com

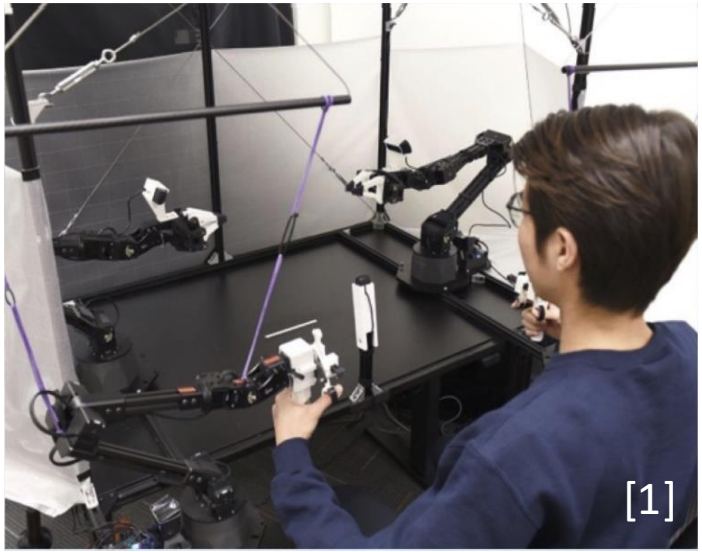

MODI ADAMS





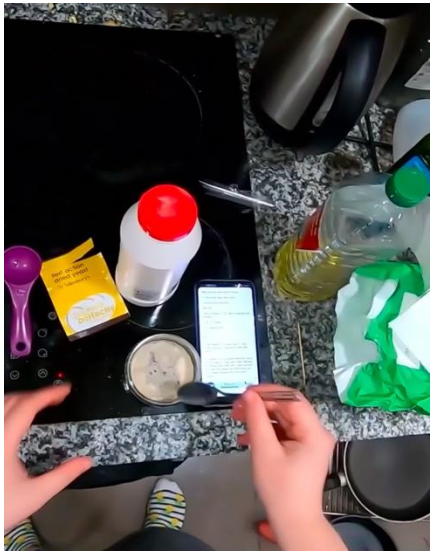
Exploring to Learn

✘ Large action space



Using Robot Demos to Learn

✘Difficult to collect



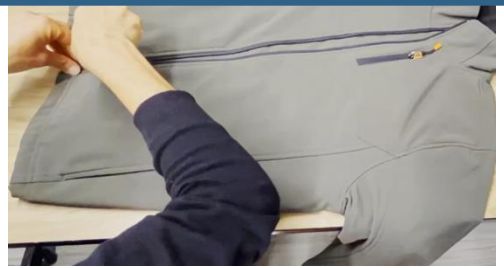
Learning from Human Demonstration

[1] Zhao et. al. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware, 2023

Bimanual Manipulation as Screw Motions



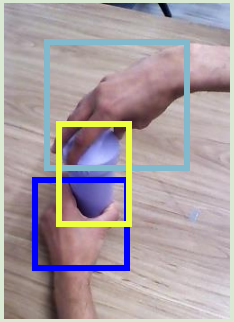
- ✔Helps to cleanly parse human demonstrations for bimanual tasks
- ✔Provides a more efficient action space for exploration



ScrewMimic

Extracting Screw Action from a Human Demonstration

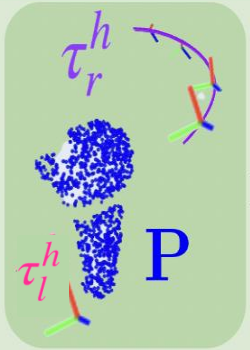
Human Video Demo (RGBD)



Hand-Object Detection

Hand Pose Detection

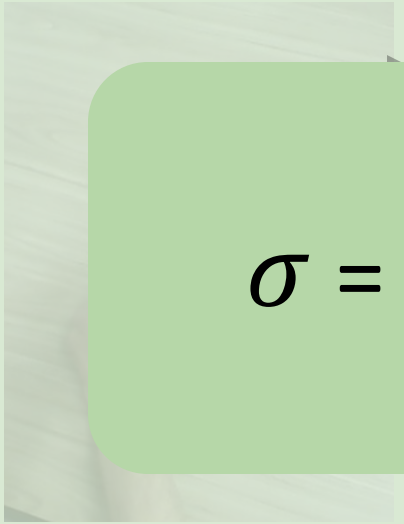
Contact Point Extraction



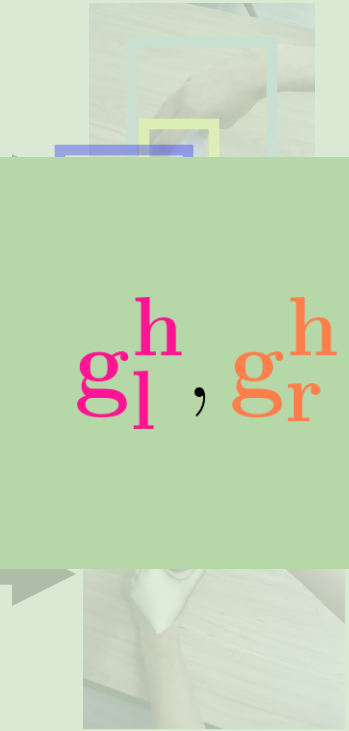
Wrist Pose Extraction

Axis Extraction

Extracting Screw Action from a Human Demonstration



Human Video Demo (RGBD)



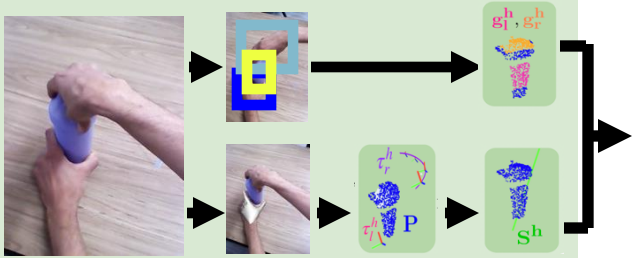
Hand Pose Detection

Wrist Pose Extraction

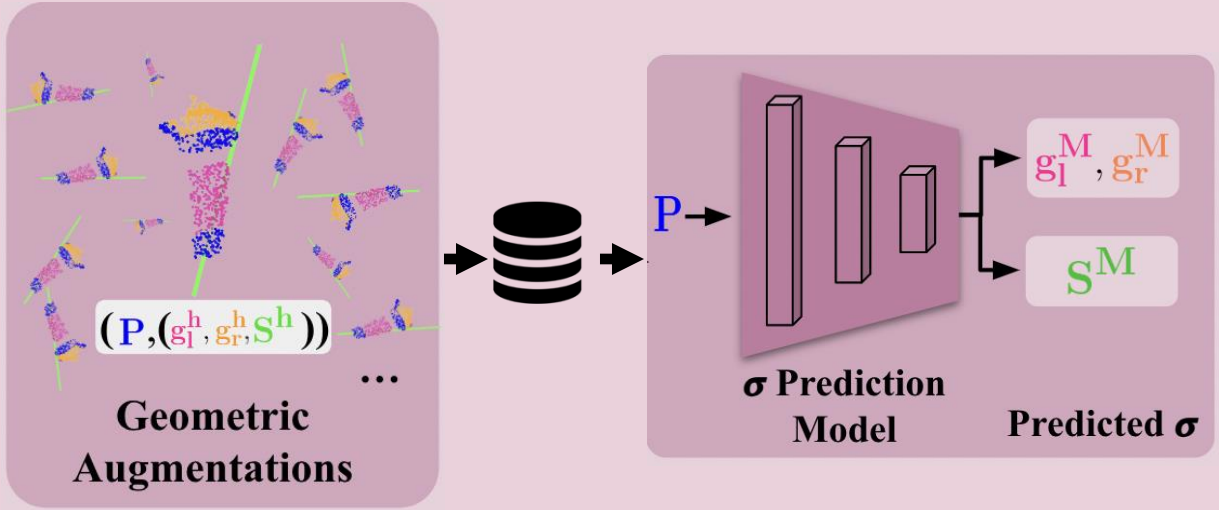
Axis Extraction

$$\sigma = (\quad \mathbf{g}_{\mathbf{l}}^{\mathbf{h}}, \mathbf{g}_{\mathbf{r}}^{\mathbf{h}} \quad \mathbf{S}^{\mathbf{h}} \quad \tau_l^h$$

Extracting Screw Action from a Human Demonstration

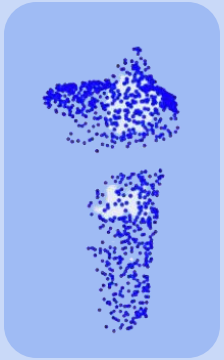


Learning to Predict a Screw Action from a Point Cloud

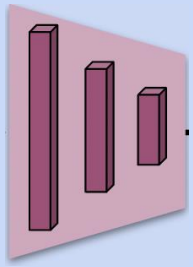


Self-Supervised Screw-Action Policy Fine-Tuning

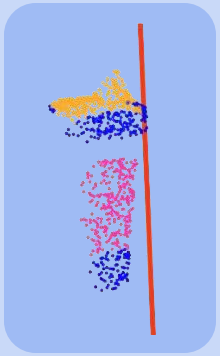
New Trial



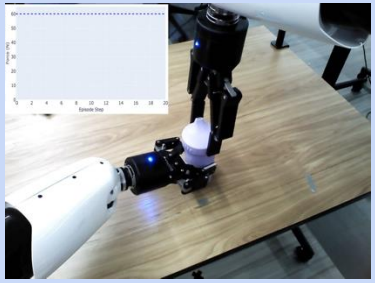
Robot Observation



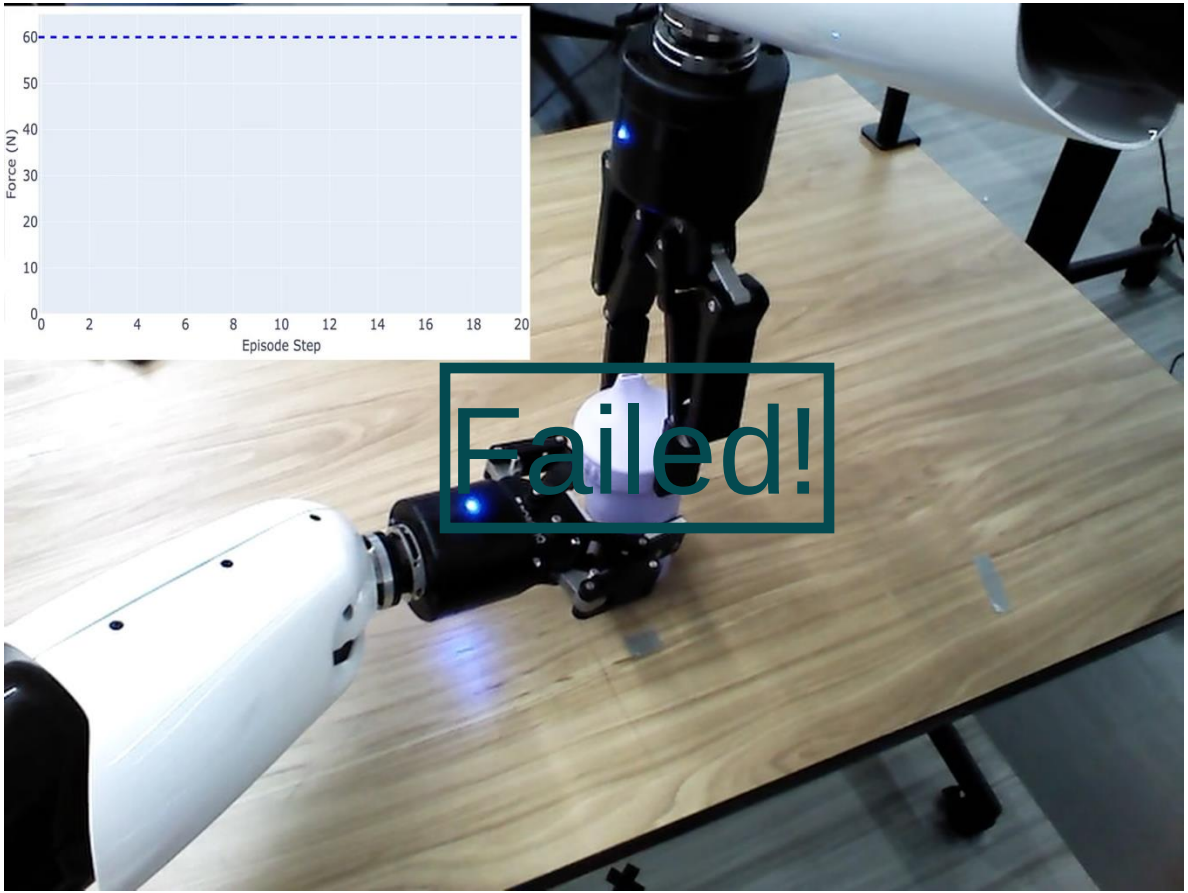
σ Prediction Model



Predicted σ

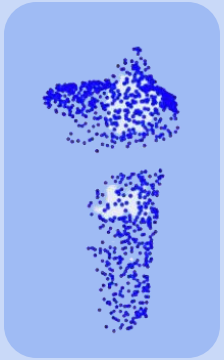


Initial Strategy

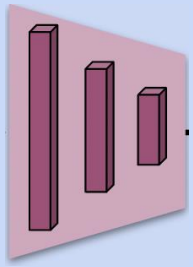

60
50
40
30
20
10
0
Force (N)
0
2
4
6
8
10
12
14
16
18
20
Episode Step
Failed!

Self-Supervised Screw-Action Policy Fine-Tuning

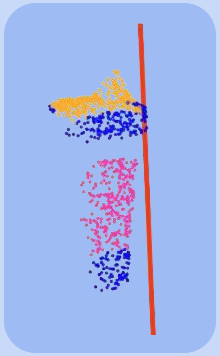
New Trial



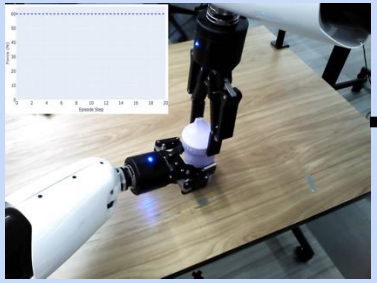
Robot Observation



σ Prediction Model



Predicted σ



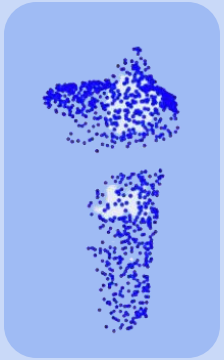
Initial Strategy

CEM-Based Fine-Tuning

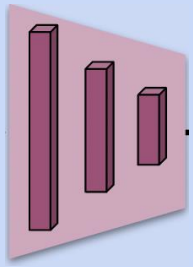

CEM-Based Fine-Tuning
Success!

Self-Supervised Screw-Action Policy Fine-Tuning

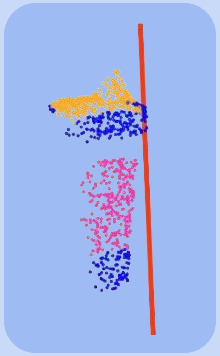
New Trial



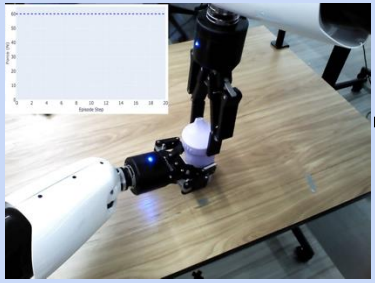
Robot Observation



σ Prediction Model

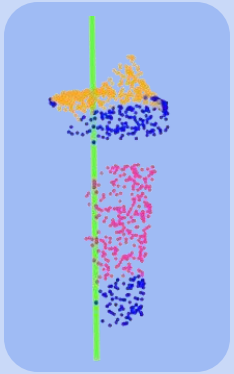


Predicted σ



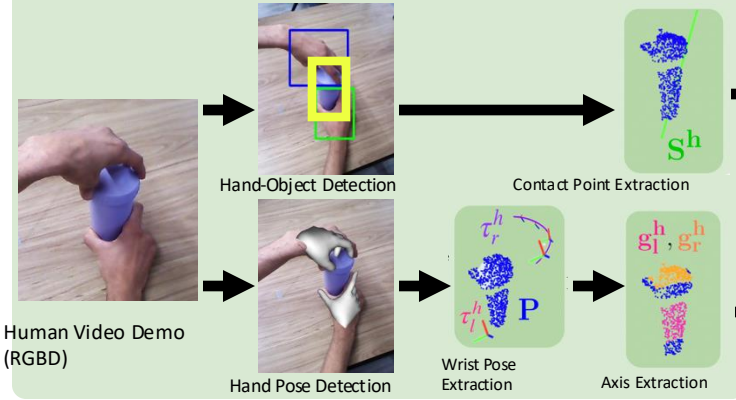
Initial Strategy

Successful Strategy

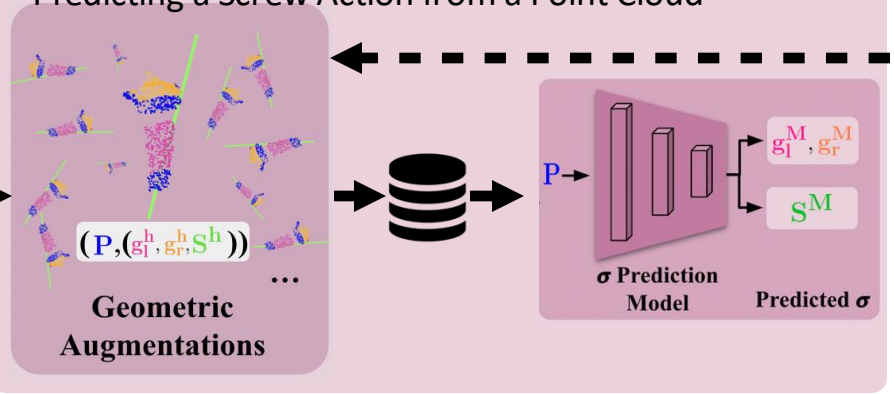


Successful σ

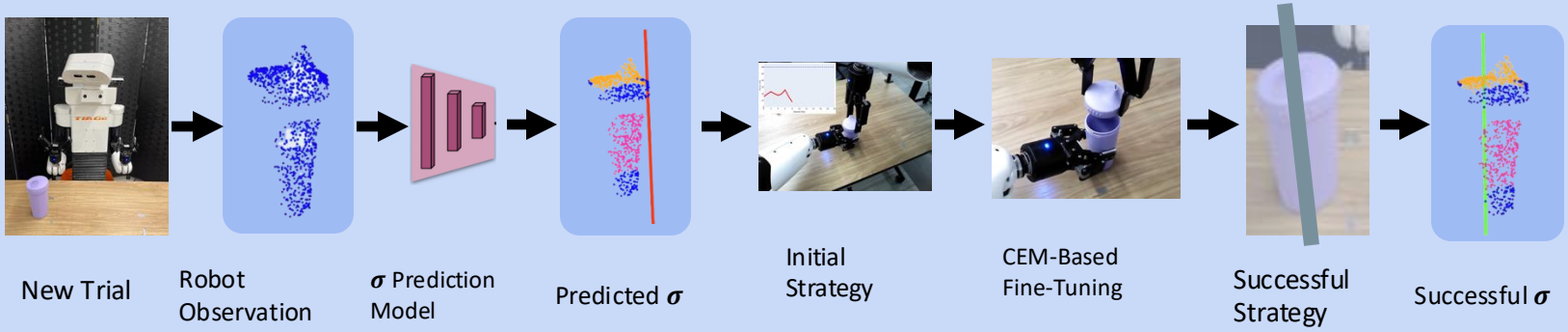
Extracting Screw Action from a Human Demo



Predicting a Screw Action from a Point Cloud



Self-Supervised Screw-Action Policy Fine-Tuning



RobIn
ROBOTIC INTERACTIVE INTELLIGENCE LAB

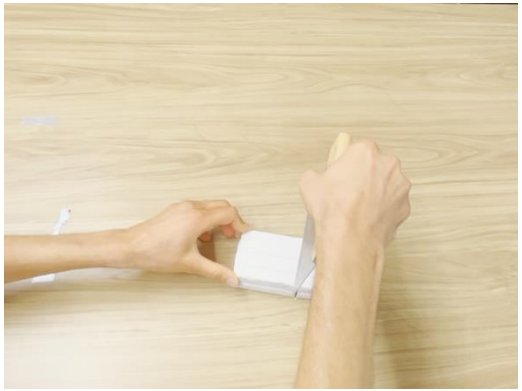
Open Bottle

Close Zipper

Insert Roll

Close Laptop

Stir



Cut

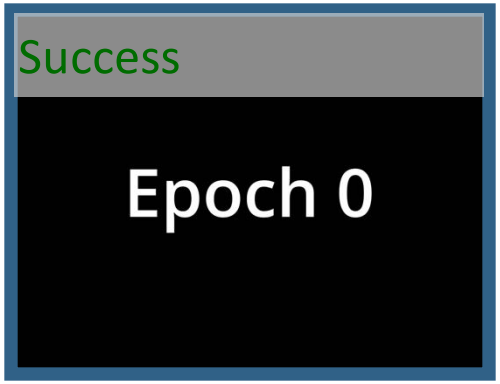
Success

Initial Strategy

Open Bottle

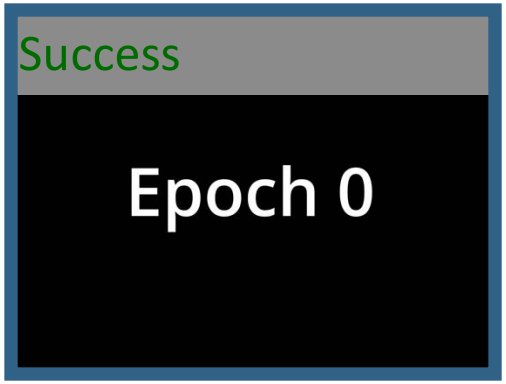
Close Zipper

Insert Roll



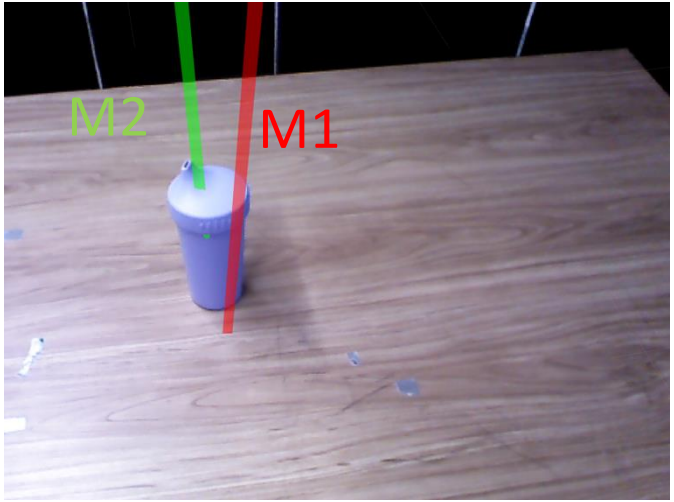
Close Laptop

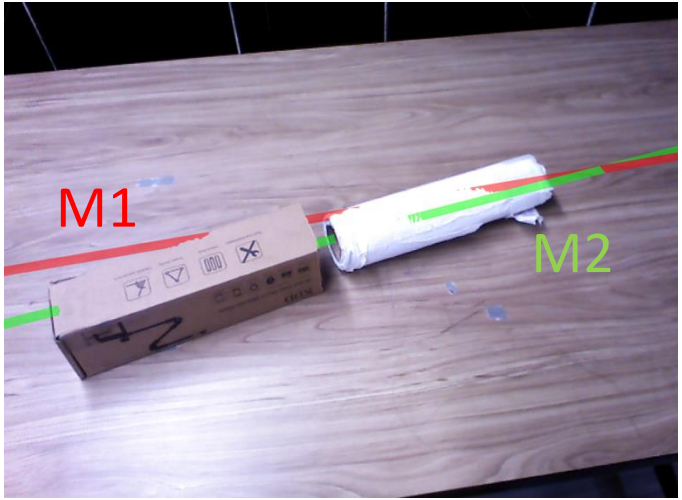
Stir



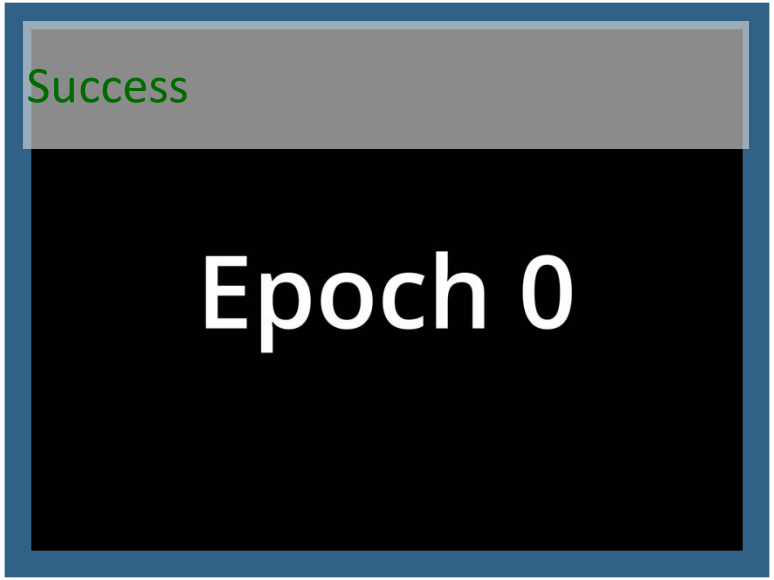
Cut

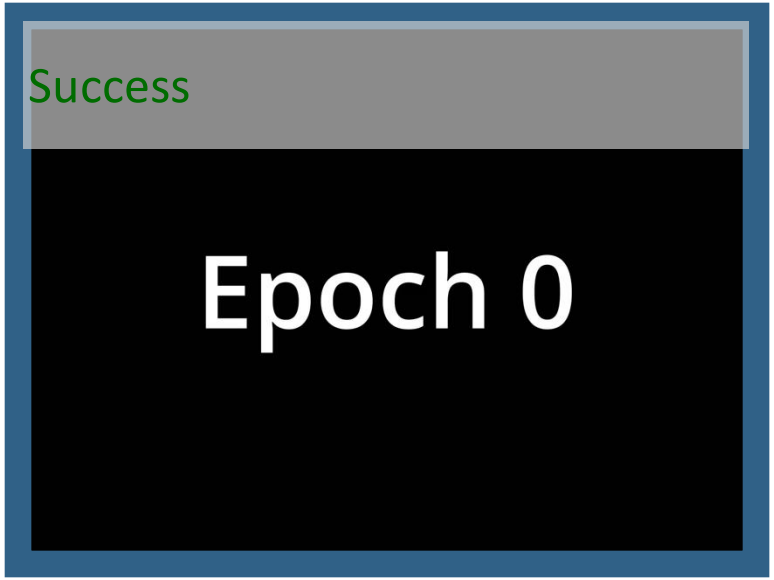
Does the screw action prediction model improve from the interactions?





Can ScrewMimic generalize to new objects?



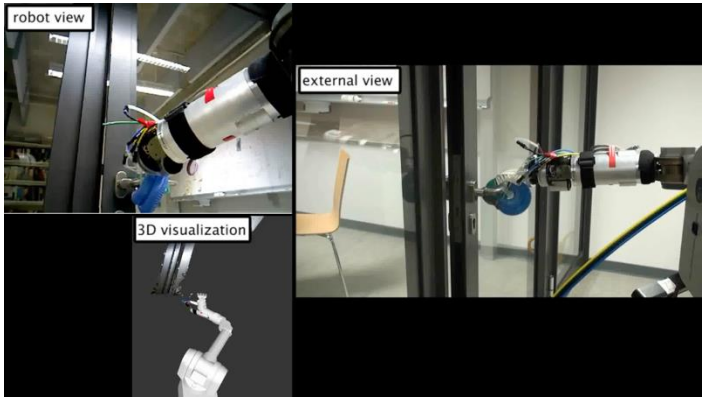


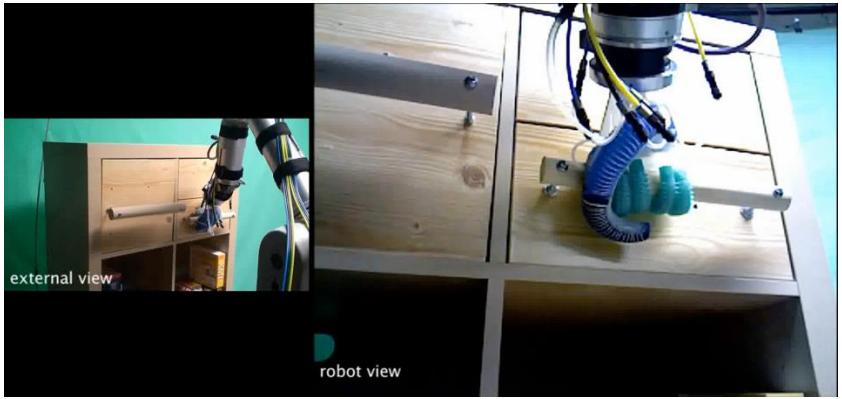
Does the Screw Action generalize when there is left-hand motion?

Key Takeaways of ScrewMimic

- Many bimanual contact-rich tasks can be represented by a simple 1-DoF screw joint
- Our screw action representation helps in more accurate human demo interpretation and efficient exploration
- ScrewMimic can continually expand its manipulation capabilities to new objects

The role of embodiment in contact-rich manipulation





[**Martín-Martín** and Brock, IJRR'18]

[Martín-Martín et al. IROS'14, ICRA'16,

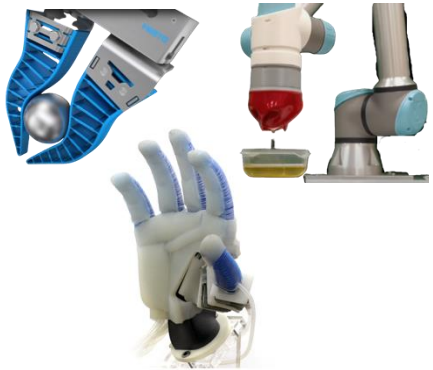
Existing robotic hand designs

Covering both extremes of the spectrum rigid-soft

✓ **Compliance**

✓ **Collision Tolerance**

✗ **Precision**



Soft

BaRiFlex

Rigid

✗ **Compliance**

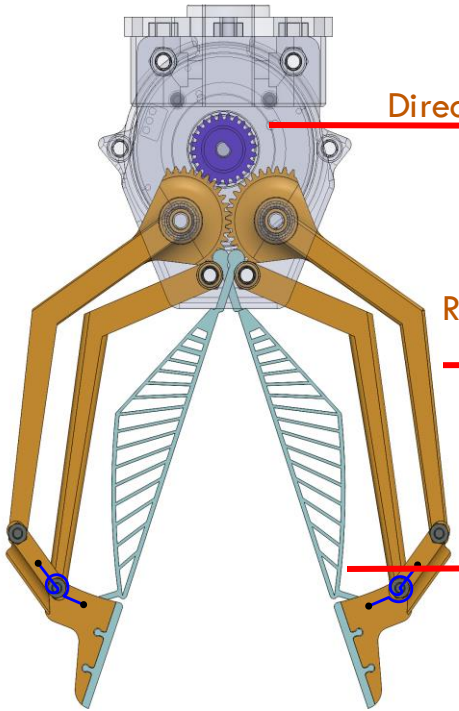
✗ **Collision Tolerance**

✓ **Precision**

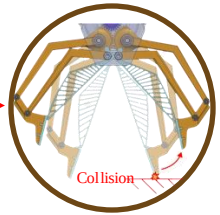
✓ **Strength**

[Festo. Co., Bui et al. '23, Puhlmann et al. '22, Franka Emika Co., On Robot Co.]

BaRiFlex Design

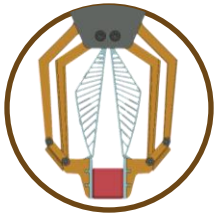


Direct-Drive Actuator



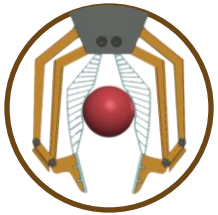
Collision Mitigation

Rigid 4-Bar Linkages



Precise Grasping

Flexible Fin-Ray Linkage



Compliant Grasping

less than $500!!

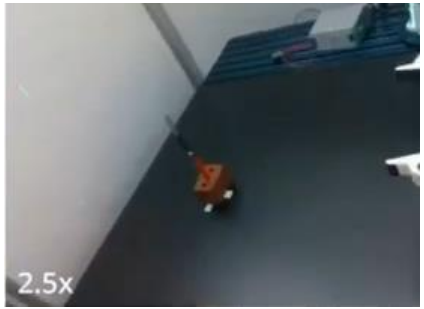
[Jeong et al., IROS'24 best mechanisms paper award]

Grasping Versatility Test (BaRiFlex)

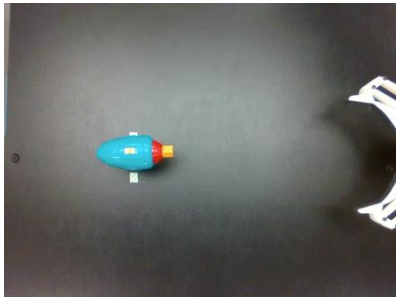
Expected Object Pose

Noise in Position

Noise in Orientation



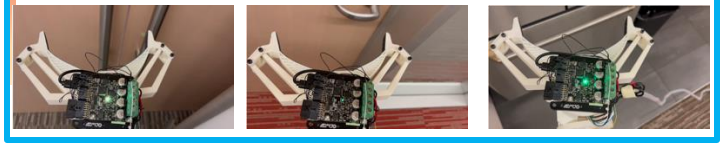
Precise/Pinch Grasping



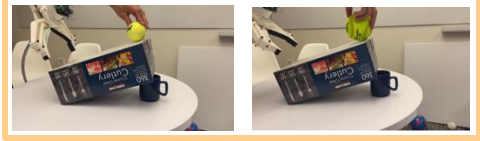
Compliant Grasping

[Jeong et al., IROS'24 best mechanisms paper award]

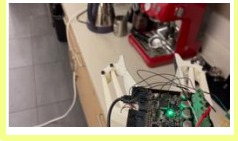
Task Versatility Test



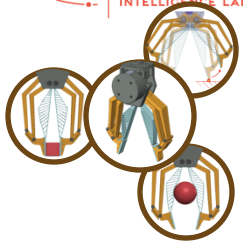
Manipulating Heavy Objects



High-Speed Catching



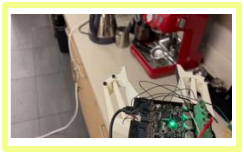
Long-Horizon Manipulation

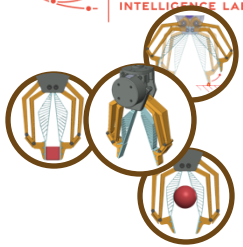


[Jeong et al., IROS'24 best mechanisms paper award]

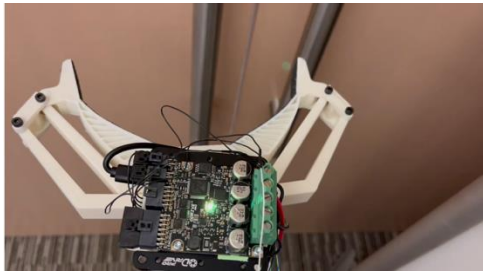
Task Versatility Test

RobIn
ROBOTIC INTERACTIVE INTELLIGENCE LAB

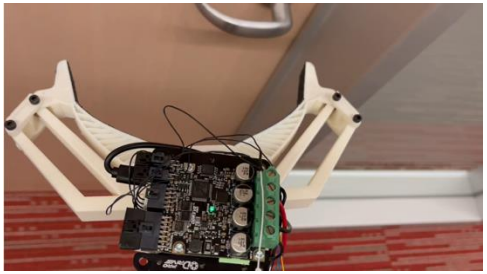




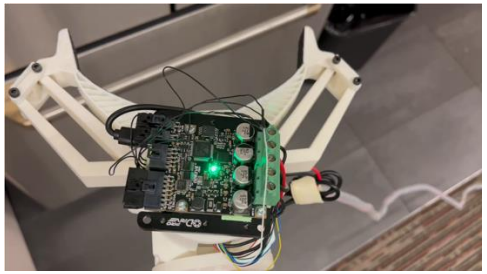
Long-Horizon Manipulation



Open Sliding Door



Open Door with Lever Handle



Open Fridge

Manipulating Heavy Objects

Catch Tennis Ball

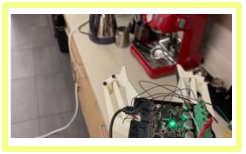
Catch Softball

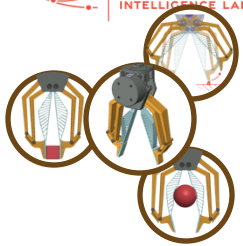
High-Speed Catching

[Jeong et al., IROS'24 best mechanisms paper award]

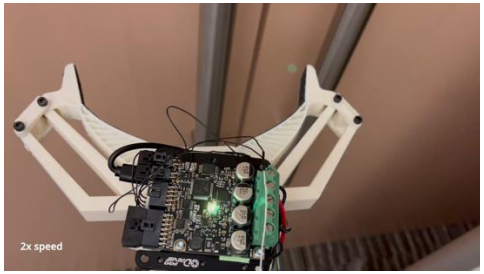
Task Versatility Test

RobIn
ROBOTIC INTERACTIVE INTELLIGENCE LAB

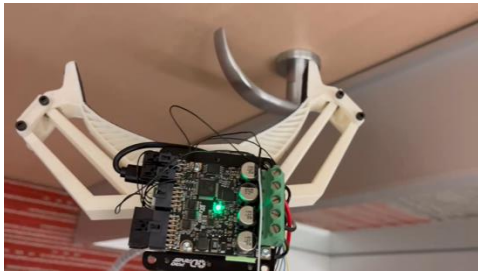




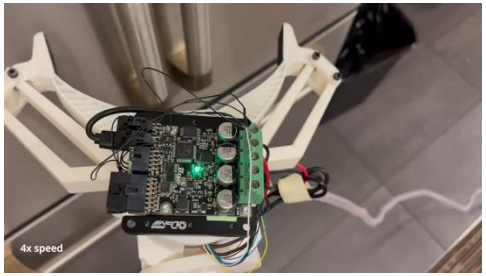
Long-Horizon Manipulation



Open Sliding Door



Open Door with Lever Handle



Open Fridge

Manipulating Heavy Objects

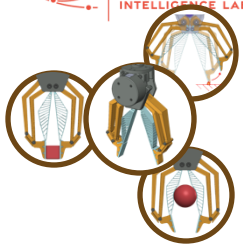
Catch Tennis Ball

Catch Softball

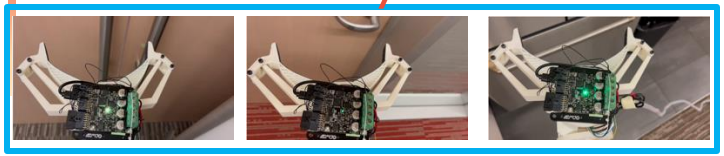
High-Speed Catching

[Jeong et al., IROS'24 best mechanisms paper award]

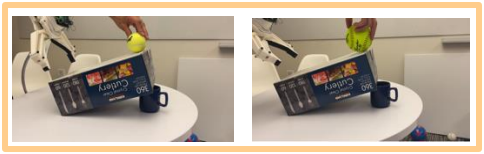
RobIn
ROBOTIC INTERACTIVE INTELLIGENCE LAB



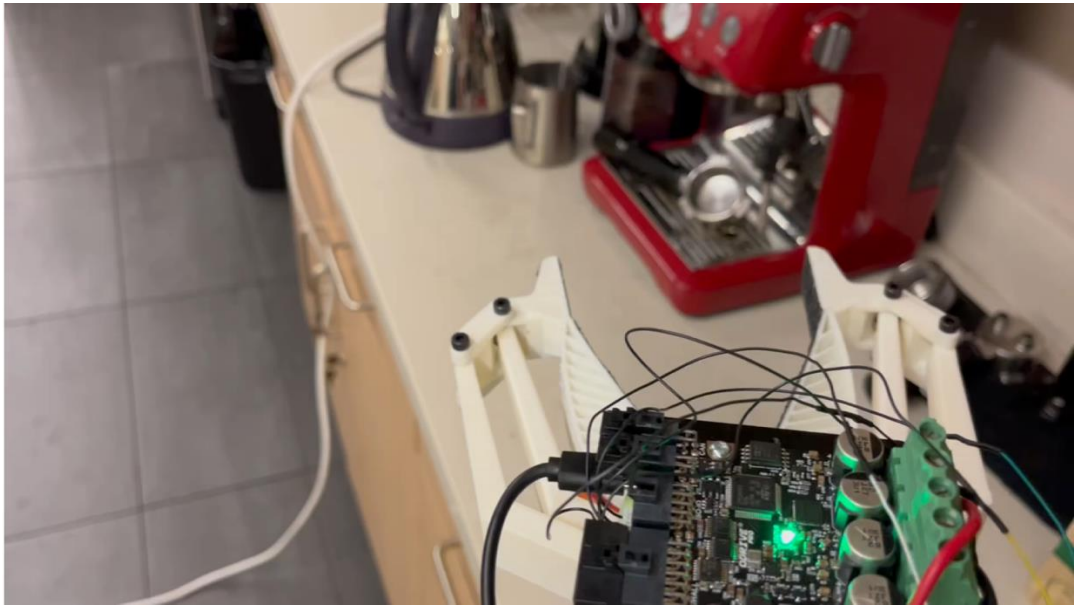
Task Versatility Test



Manipulating Heavy Objects



High-Speed Catching

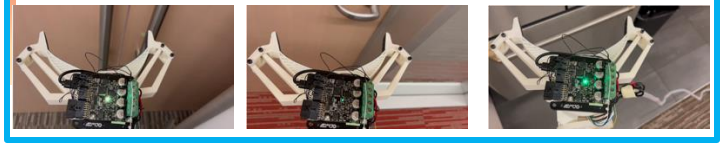


Make Instant Coffee

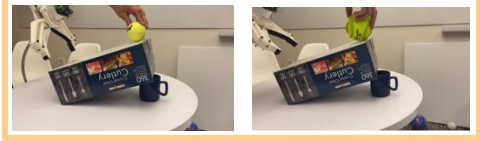
Long-Horizon Manipulation

[Jeong et al., IROS'24 best mechanisms paper award]

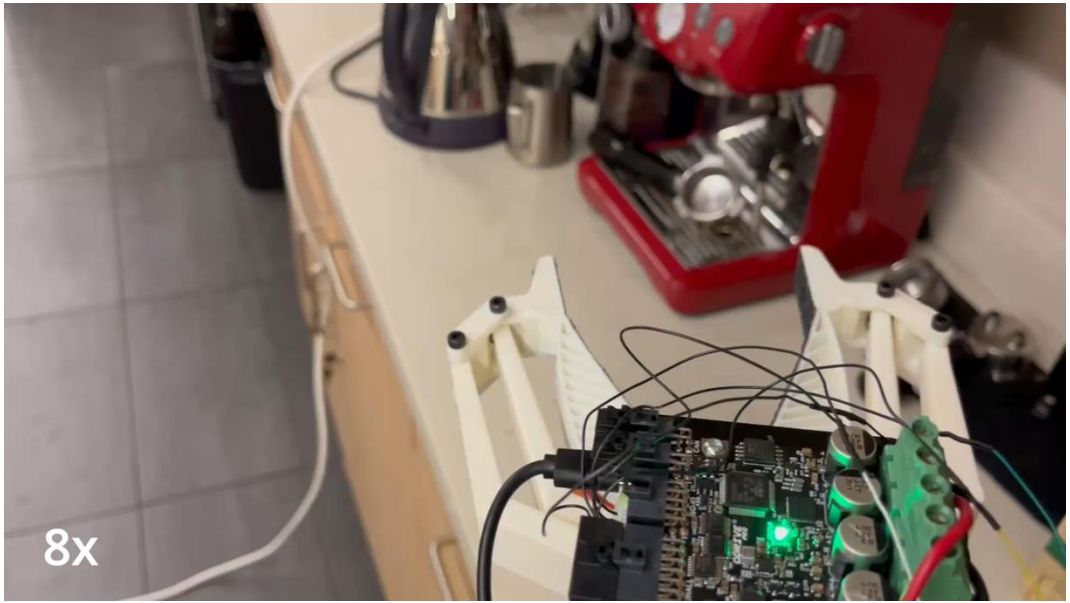
Task Versatility Test



Manipulating Heavy Objects



High-Speed Catching

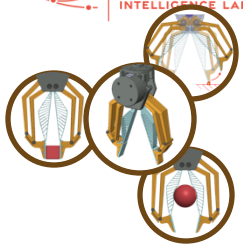


Make Instant Coffee

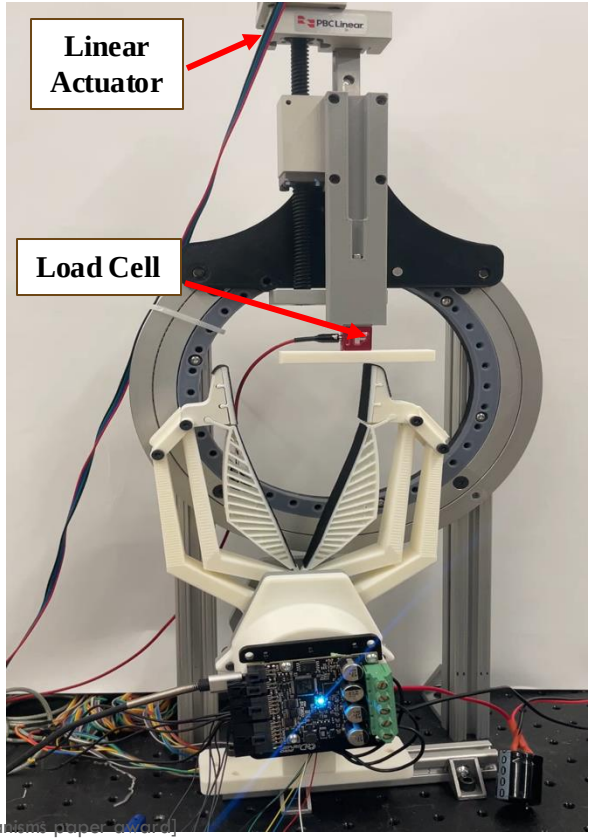
Long-Horizon Manipulation

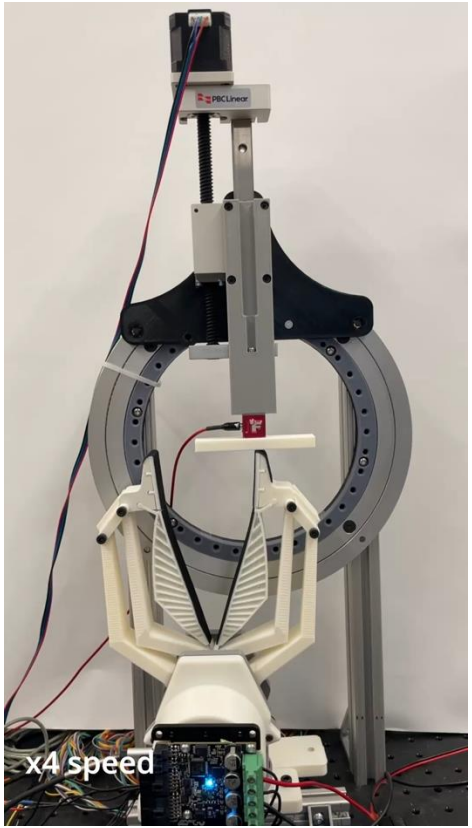
[Jeong et al., IROS'24 best mechanisms paper award]

RobIn
ROBOTIC INTERACTIVE INTELLIGENCE LAB



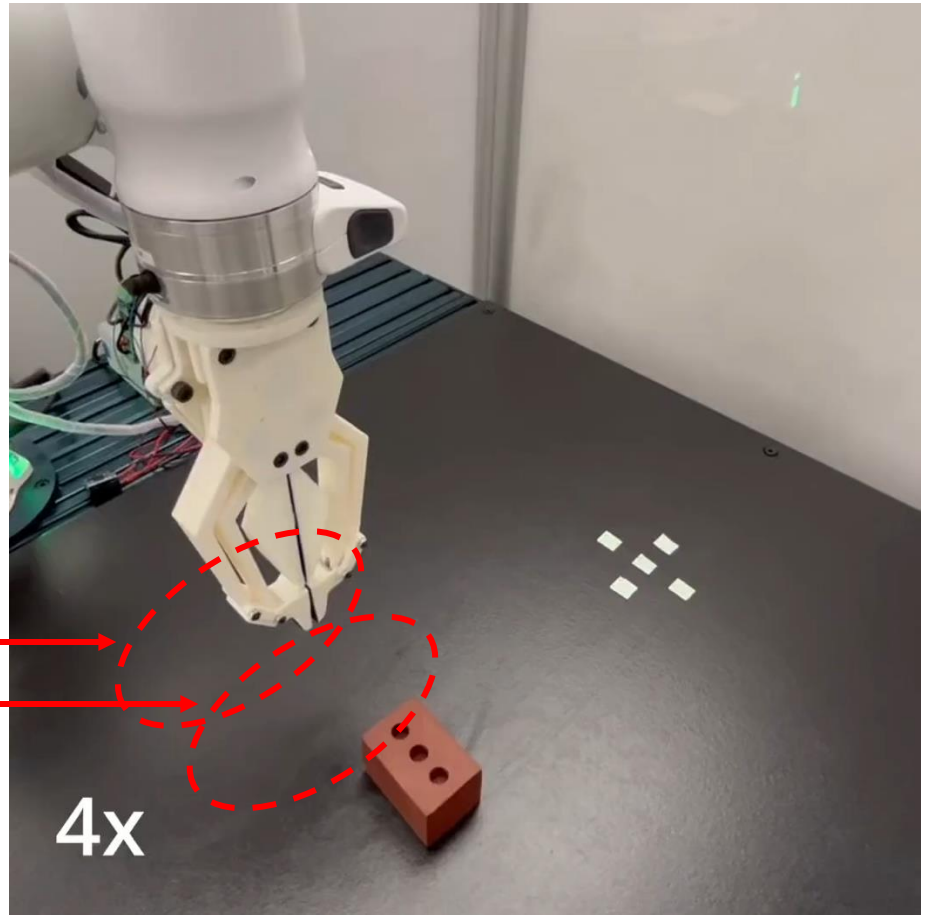
Robustness and Durability Tests



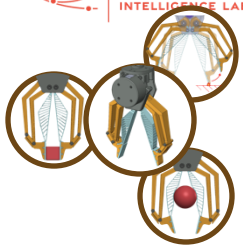


[Jeong et al., IROS'24 best mechanisms paper award]

Real World Reinforcement Learning Test



RobIn
ROBOTIC INTERACTIVE INTELLIGENCE LAB



Gu-Cheol Jeong

Arpit Bahety

Ashish Despande

[Jeong et al., IROS'24 best mechanisms paper award]

Final Thoughts

- Last challenge in robotics is to control contact-rich interactions
- Variable impedance facilitates learning to control contact-rich manipulation
- Bimanual manipulation of contact-rich tasks can be approximated by 1DoF screw axes

Thank you!

robertomm@cs.utexas.edu

RobIn
ROBOT INTERACTIVE INTELLIGENCE LAB

THE UNIVERSITY OF TEXAS AT AUSTIN