



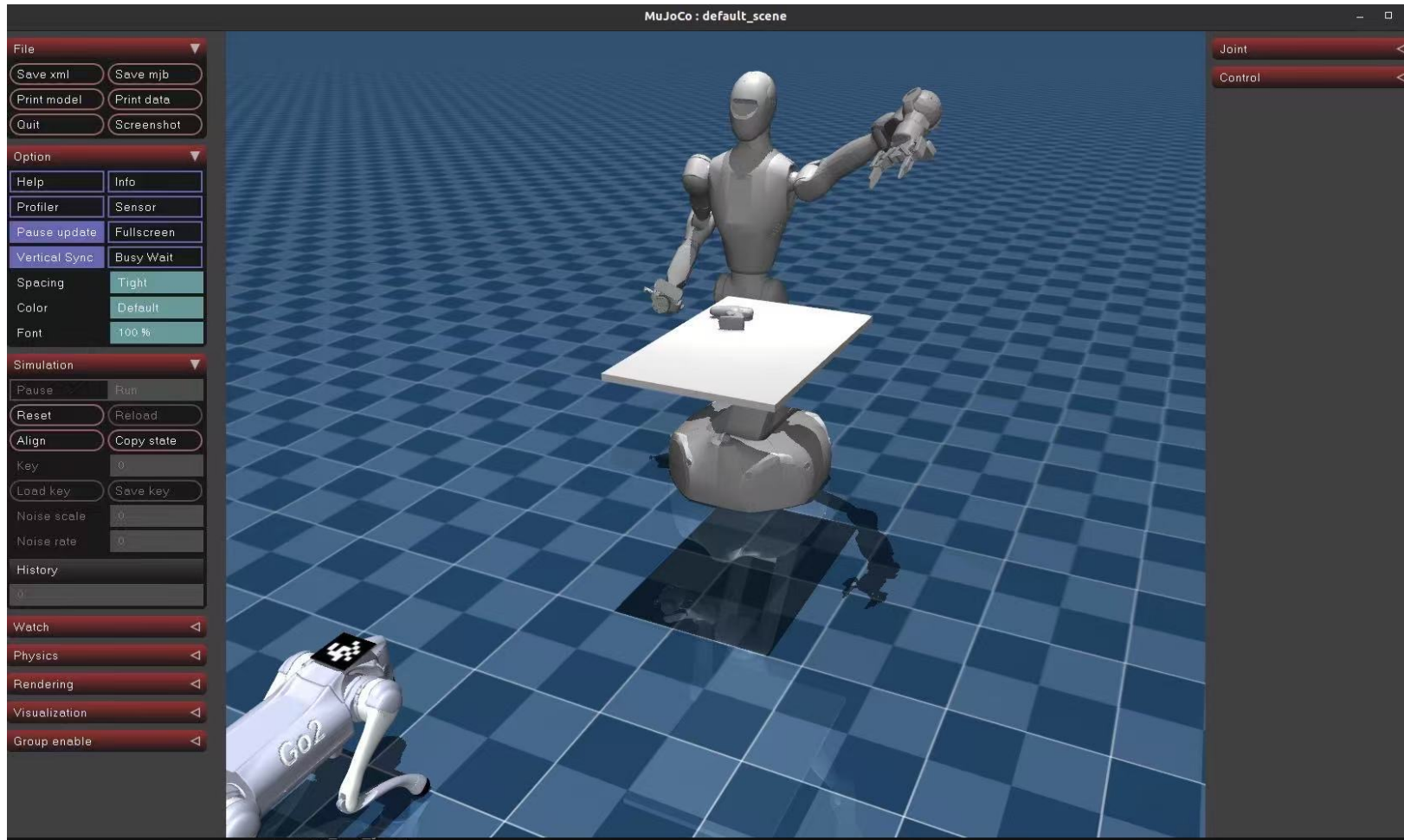
Introduction to Embodied AI

Lecture 13 - Robotic Task III

Prof. He Wang



Final Project



Task 1: grasp the object; Task 2: Go 2 navigation; Task 3: place the object.

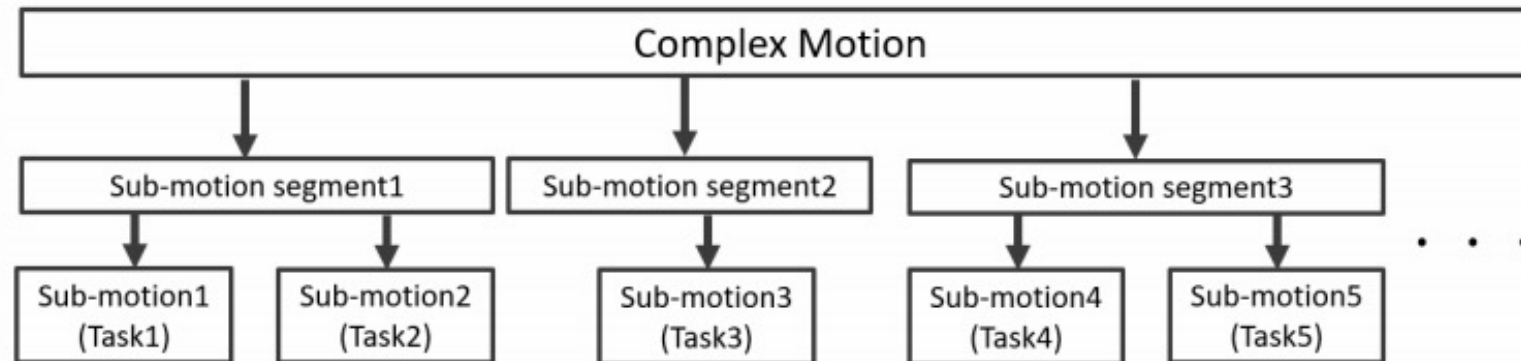
Final Project Timeline

- Release date: 05/21 Wednesday evening
- Real test data release date: 05/26
- Sim evaluation: 06/04 Wednesday evening
- Debugging real experiment (optional): 06/07-08
- Real experiment date: 06/09 Monday 3-6PM

Gripper Manipulation

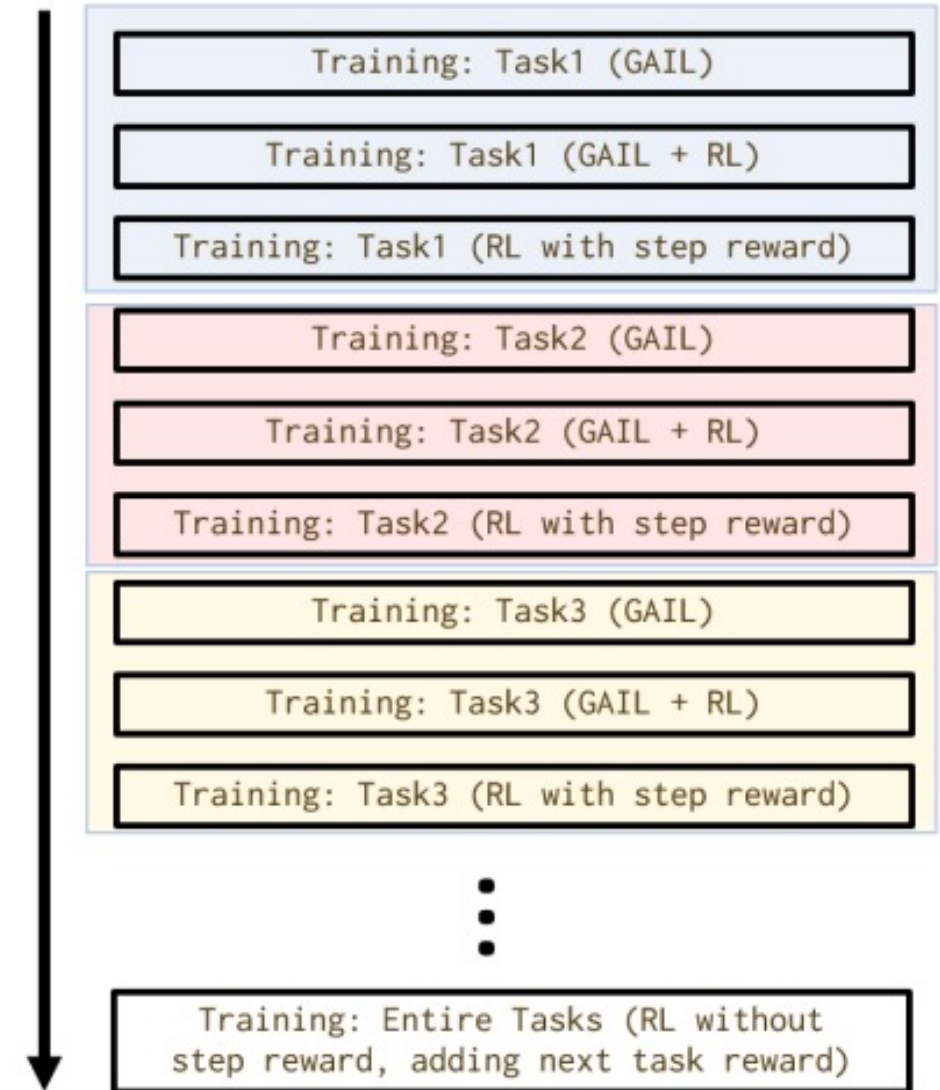
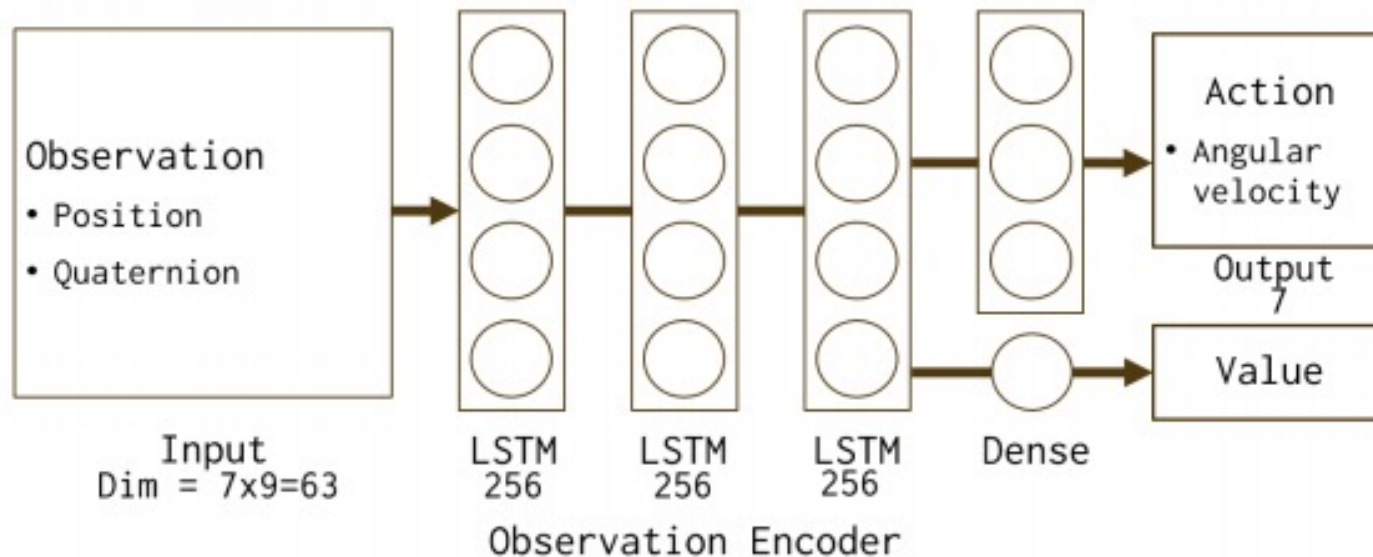
Learning 6DoF Grasping Using Reward-Consistent Demonstration

- On-policy
- The grasping task is divided into consecutive stages:
 - Orienting the end-effector, approaching the target, and closing the gripper
 - To simplify the reward design, and make it easy for humans to control the robot



Learning 6DoF Grasping Using Reward-Consistent Demonstration

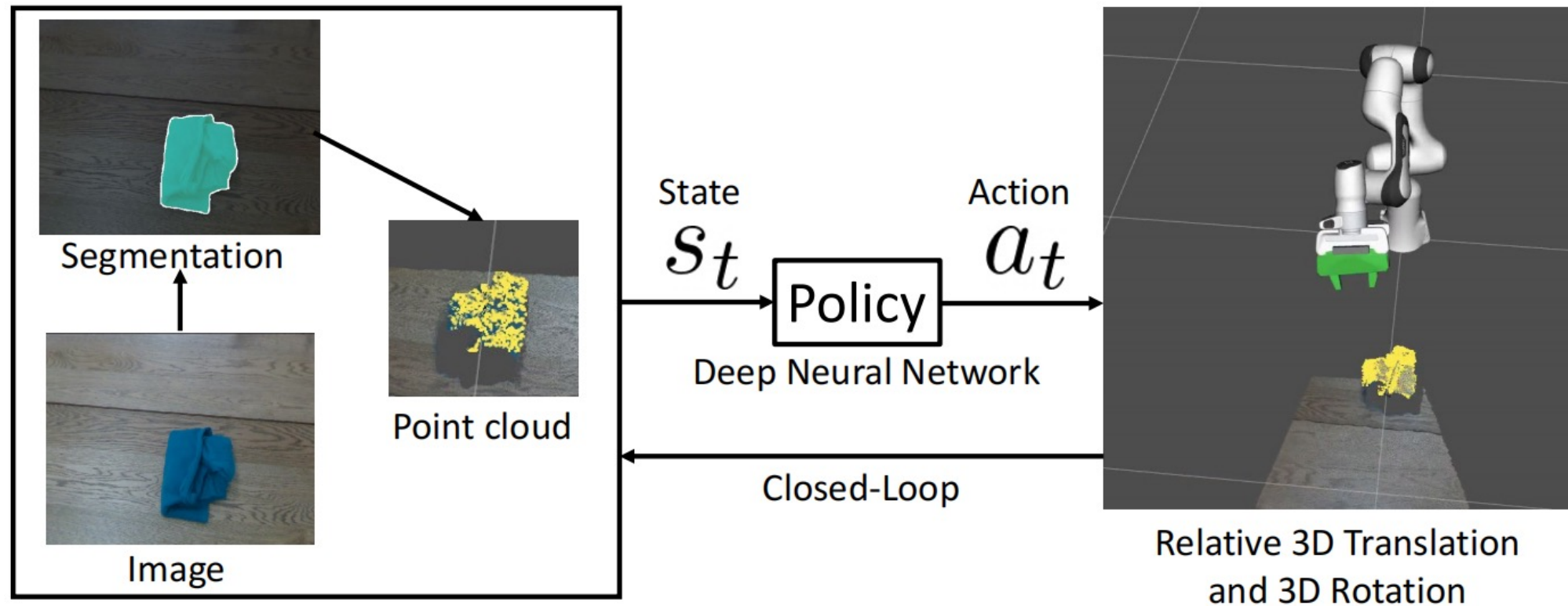
- A different RL model is trained for each stage
- Imitation learning method: **Generative adversarial imitation learning (GAIL)**
- Learning algorithm: **PPO**
- Policy network:



Goal-auxiliary actor-critic for 6D robotic grasping with point clouds (GA-DDPG)

Learning a closed-loop policy for 6D robotic grasping

- Off-policy



Deep Deterministic Policy Gradient (DDPG)

- Actor-critic algorithm
- Actor: learn the policy $\pi_{\theta}(s)$
- Critic: approximate the Q-function $Q_{\phi}(s, a)$
- Replay buffer: $D = \{(s, a, r, s')\}$, examples sampled from it are used to optimize the actor and critic alternately
- Minimize:
$$\mathbb{E}_{(s,a,r,s') \sim D} \left[\frac{1}{2} \left(Q_{\phi}(s, a) - (r + \gamma Q_{\phi}(s', \pi_{\theta}(s'))) \right)^2 \right]$$
- The deterministic policy π_{θ} is trained to maximize the learned Q-function with
$$\max_{\theta} \mathbb{E}_{s \sim D} (Q_{\phi}(s, \pi_{\theta}(s)))$$

6D Grasping Policy From Point Clouds

- **Goal:**

learn a 6D grasping policy $\pi : s_t \rightarrow a_t$ (map the state s_t at time t to an action a_t)

- **Action:**

the 3D translation and the 3D rotation of the robot end-effector

- **State:**

3D point clouds of objects

Demonstrations from a Joint Motion and Grasp Planner

- Expert
 - Optimization-based Motion and Grasp Planner (OMG Planner)
 - Given a planning scene and a set of pre-defined grasps on a target object from a grasp planner, the OMG Planner plans a trajectory of the robot to grasp the object
- Construct an offline dataset of demonstrations D_{expert}

Behavior Cloning and DAGGER

- Train a policy network π_θ by applying **Behavior Cloning (BC)** to predict an action from a state

- Loss:

$$L_{\text{POSE}}(\mathcal{T}_1, \mathcal{T}_2) = \frac{1}{|X_g|} \sum_{x \in X_g} \|\mathcal{T}_1(x) - \mathcal{T}_2(x)\|_1$$

- Point matching loss

- Jointly optimize translation and rotation $L_{\text{BC}}(a^*, a_\theta) = L_{\text{POSE}}(a^*, a_\theta)$

- DAGGER:

- Augment supervisions for rollouts from the learner
 - Construct a dataset D_{dagger}

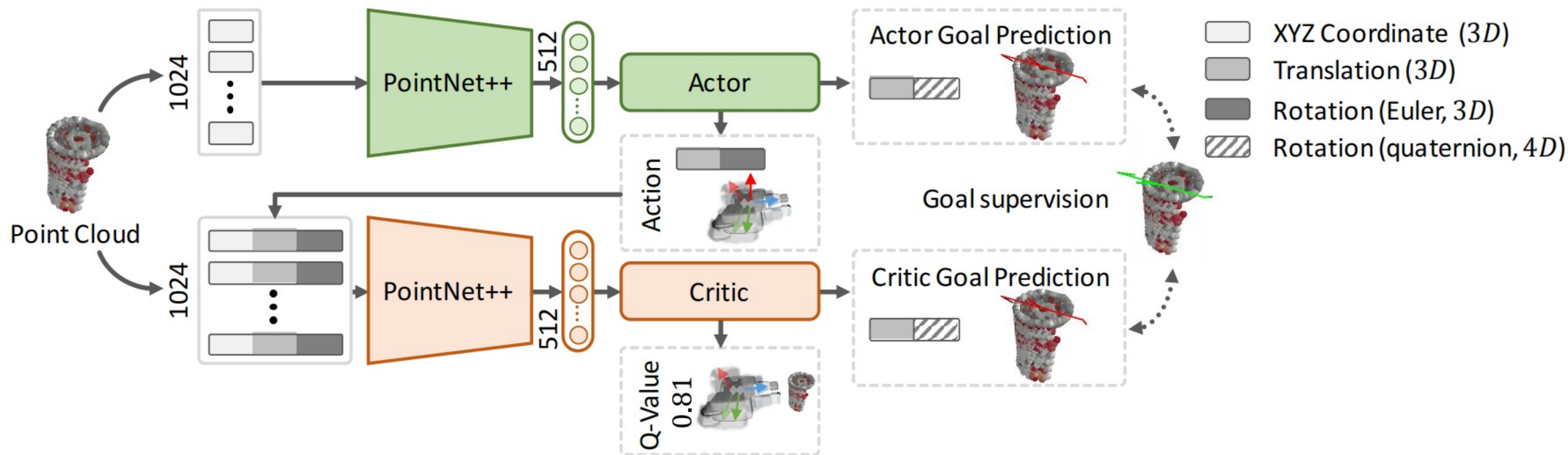
GA-DDPG

Goal-Auxiliary DDPG

- To handle contact-rich scenarios that are rare in the expert trajectories
- Improve the policy learning by combining RL with expert demonstrations
- Construct a dataset for DDPG D_{ddpg}
 - Similar to DAGGER, collect on-policy data for DDPG training
 - Difference: do not have expert demonstration for the action and goal for a state from the learner policy. Instead, we first find the nearest goal from a goal set G by $\tilde{g}_t = \arg \min_{g \in G} L_{\text{POSE}}(g, \mathcal{T}_t)$. Then the heuristic goal for s_t is $g_t = \tilde{g}_t \mathcal{T}_t^{-1}$
- The replay buffer of DDPG training: $D = D_{\text{expert}} \cup D_{\text{dagger}} \cup D_{\text{ddpg}}$

GA-DDPG

Goal-Auxiliary Actor-Critic Network



$$L_{\theta} = \lambda L_{BC}(a^*, a_{\theta}) + (1 - \lambda) L_{DDPG}(s, a_{\theta}) + L_{AUX}(g, g_{\theta})$$

$$L_{BC}(a^*, a_{\theta}) = L_{POSE}(a^*, a_{\theta}) \quad L_{DDPG}(s, a_{\theta}) = -Q_{\phi}(s, a_{\theta}) \quad L_{AUX}(g, g_{\phi}) = L_{POSE}(g, g_{\phi})$$

QT-Opt: Scalable Deep Reinforcement Learning for Vision-Based Robotic Manipulation

- Large-scale distributed optimization + QT-Opt (Q-learning algorithm)
- Stage:
 - a. Train a model entirely offline, using the already collected data
 - b. Then deploy and finetune that model on the real robot, further training it on newly collected data

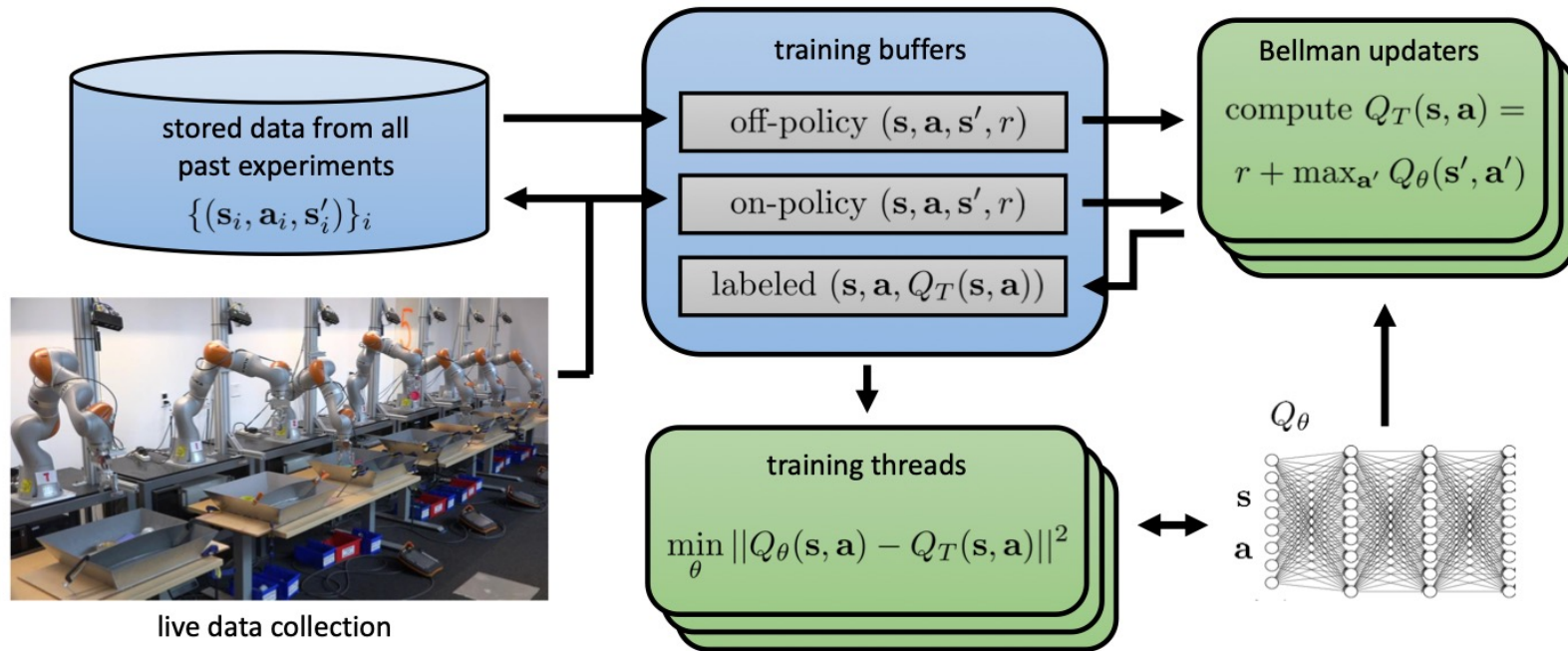


Seven robots collecting grasp data

QT-Opt

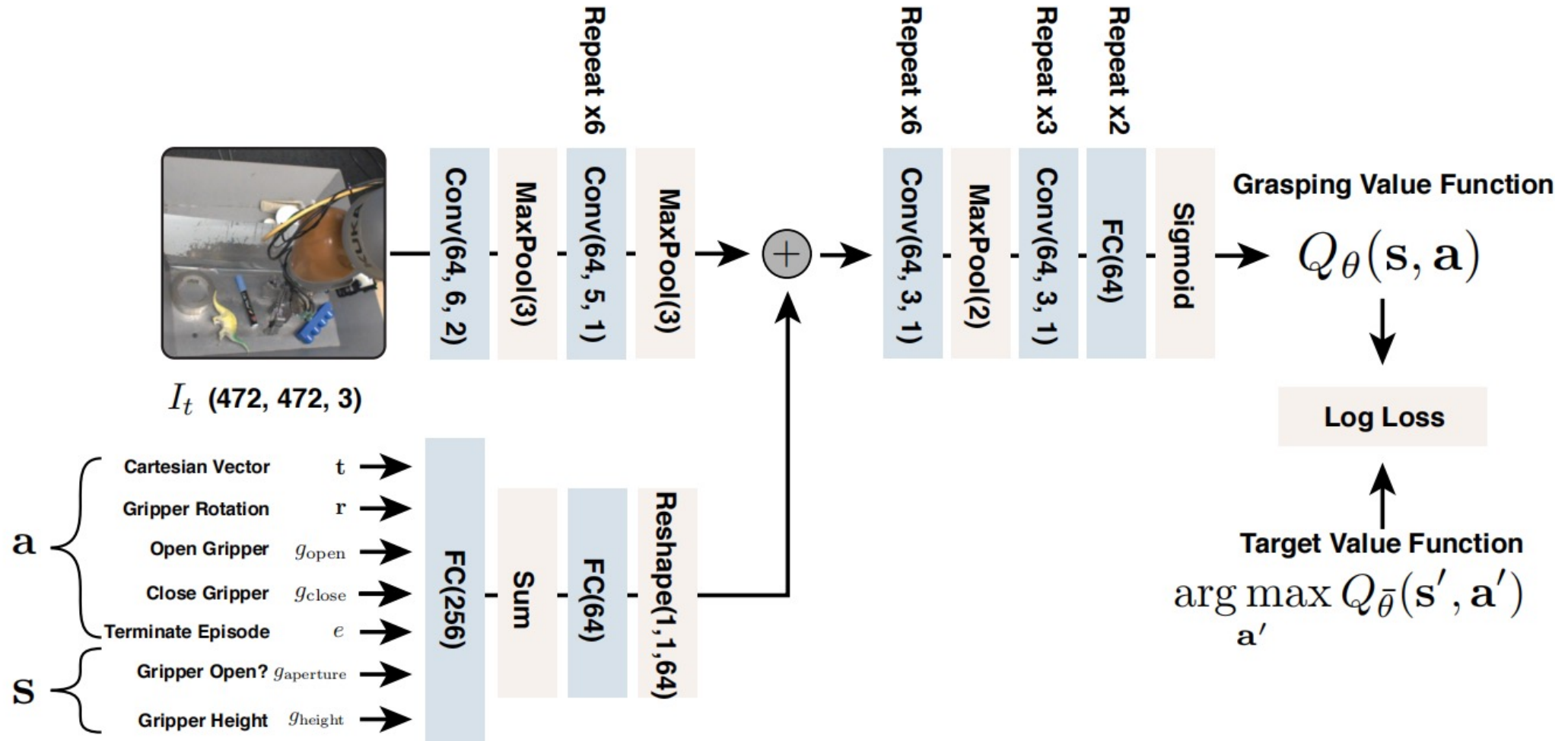
Distributed RL infrastructure for QT-Opt

- State-action-reward tuples are loaded from an offline data stored and pushed from online real robot collection.
- Bellman update jobs sample transitions and generate training examples, while training threads update the Q-function parameters.



QT-Opt

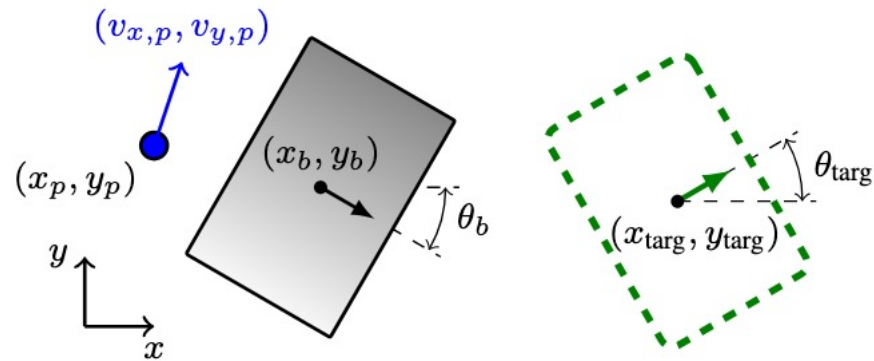
The architecture of grasping Q-Function



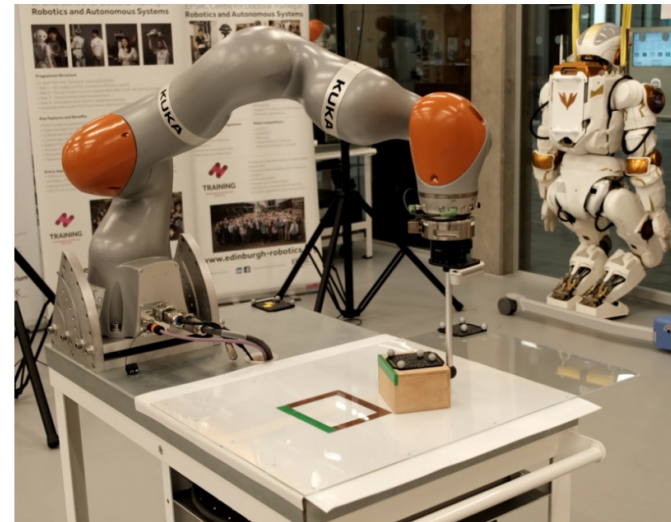
Non-prehensile Manipulation

Planar Pushing

- Parameterization: obs (x, y, θ) goal (x, y, θ) action (x, y, v_x, v_y)
- Solution: Goal-conditioned State-based RL

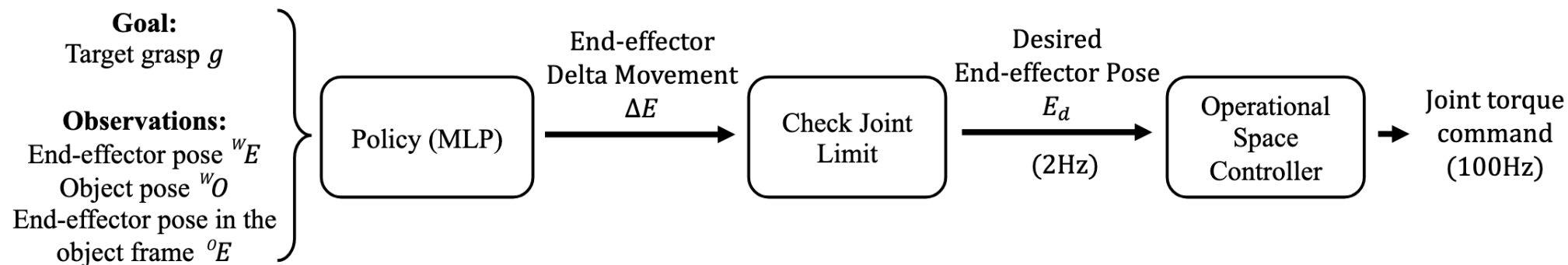


Observation and action



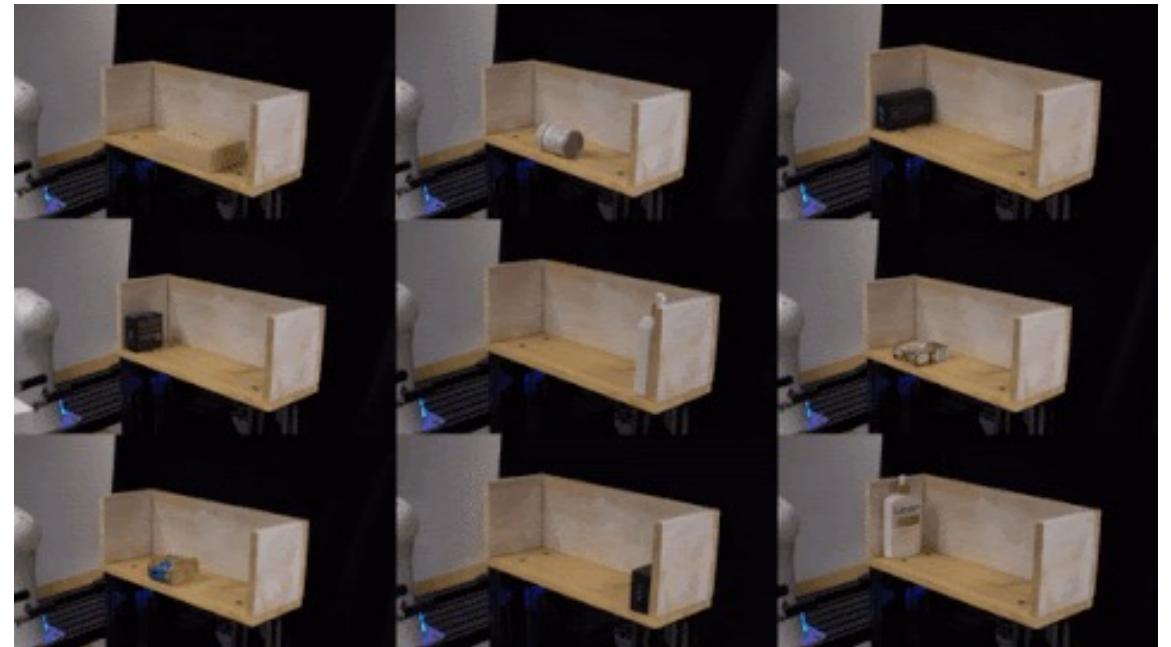
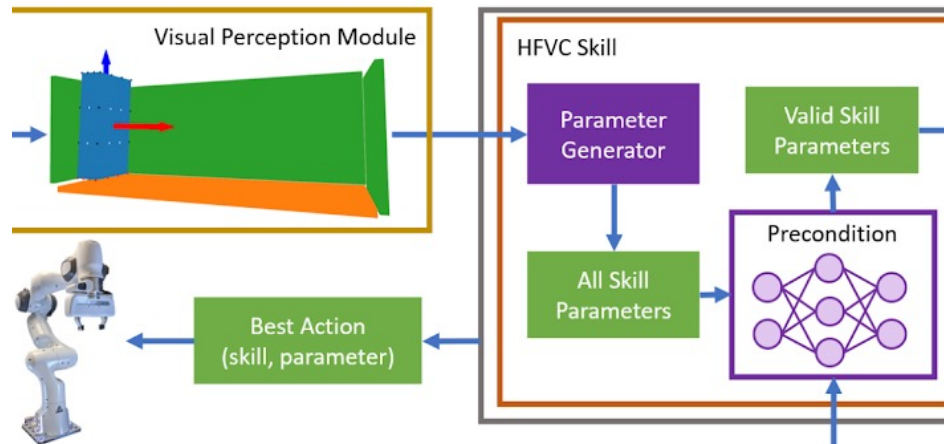
Extrinsic Dexterity

- Motivation: Utilize the external environment to help grasping
- Solution: Goal-conditioned state-based RL + OSC control



Non-prehensile Manipulation via Task Planning

- Skill: Parameterized Force-Velocity Controller
- Perception + Task Planning
- Limitation: Predefined skills for known instances



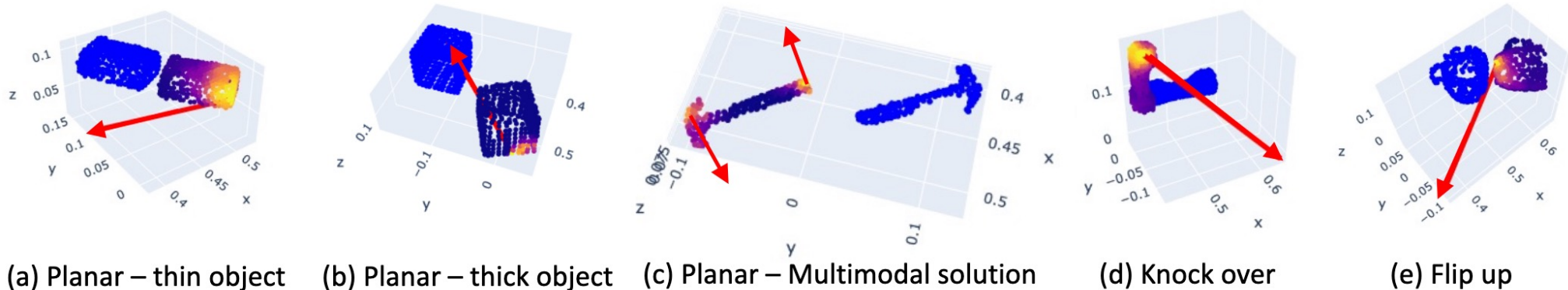
Generalizable 6D Non-prehensile Manipulation

- Previous works:
 - Instance-level
 - State-based policy
- Recent Trend
 - Vision-based policy
 - Generalization on Unseen Objects
 - 6D: Arbitrary Stable Goal Pose



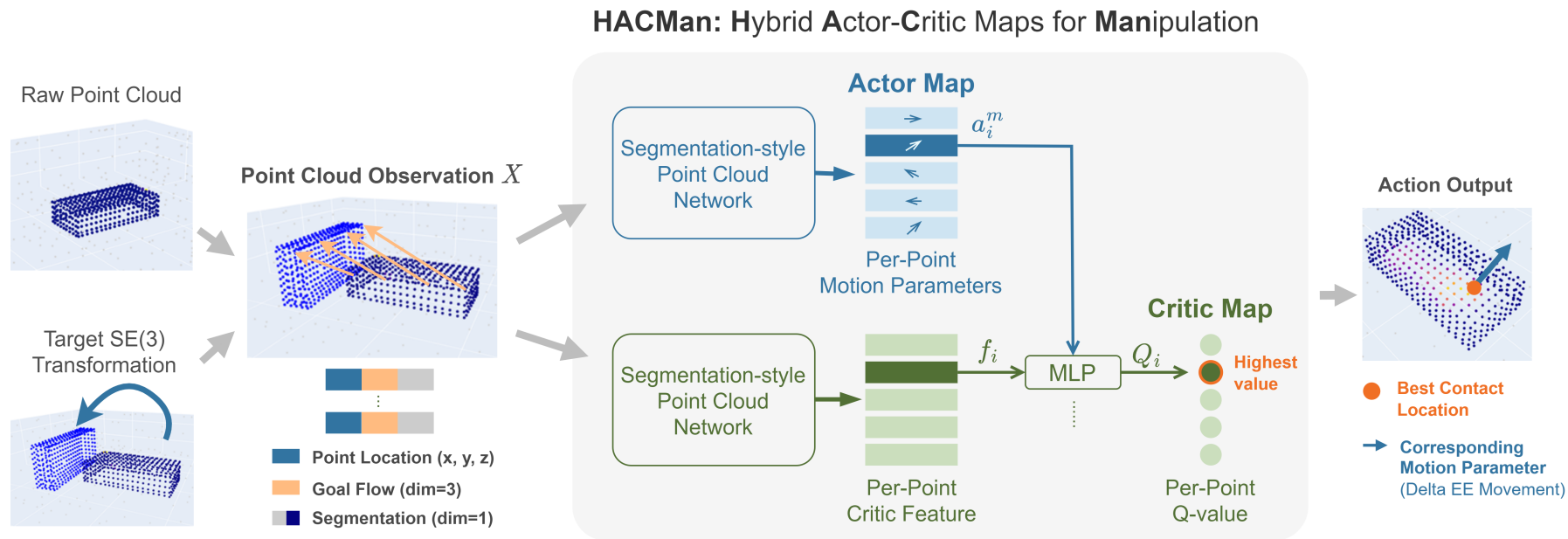
HACMan

- Action Parameterization: contact point and pushing direction
- High-level idea: Learning the action map via RL
- Challenges: Action space contains both discrete part (contact point) and continuous part (pushing direction)



HACMan

- Input: Point cloud + Goal Flow + Mask
- Concat critic feature with actor map (motion parameter)
- Argmax critic value for choosing the action



HACMan

- Advantage: Generalization across unseen objects
- Limitation:
 - leveraging motion primitive → cannot achieve continuous and smooth action
 - Cannot generalize to different table friction

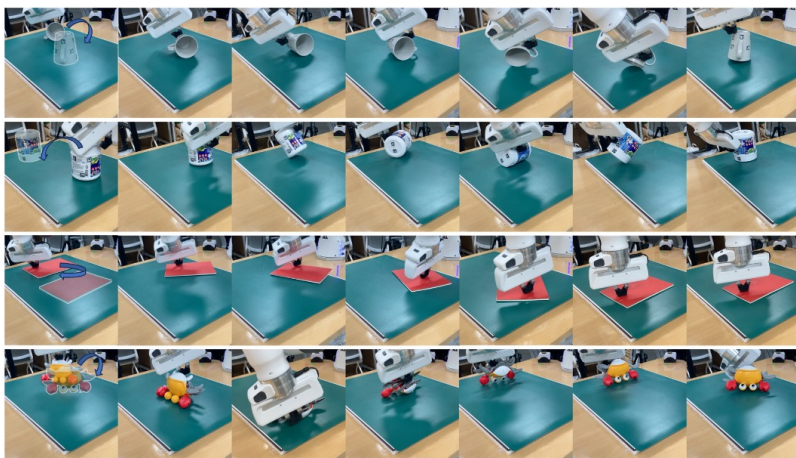
Closed-loop Non-prehensile Manipulation Policy

CORN: CONTACT-BASED OBJECT REPRESENTATION FOR NONPREHENSILE MANIPULATION OF GENERAL UNSEEN OBJECTS

Yoonyoung Cho*, Junhyek Han*, Yoontae Cho, Beomjoon Kim

Korea Advanced Institute of Science and Technology

{yoonyoung.cho, junhyek.han, yoontae.cho, beomjoon.kim}@kaist.ac.kr



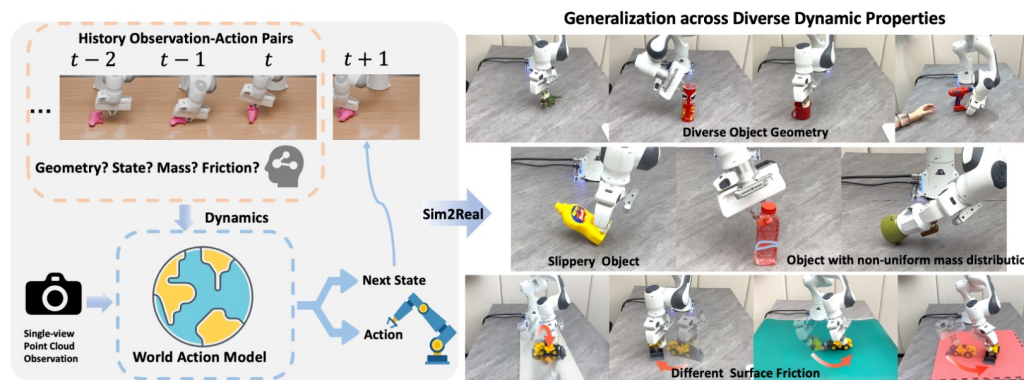
ICLR 2024

DyWA: Dynamics-adaptive World Action Model for Generalizable Non-prehensile Manipulation

Jiangran Lyu^{1,2}, Ziming Li^{1,2}, Xuesong Shi², Chaoyi Xu², Yizhou Wang^{1,†}, He Wang^{1,2,†}

¹CFCS, School of Computer Science, Peking University ²Galbot

<https://pku-epic.github.io/DyWA/>



ArXiv 2025

Dexterous Manipulation

Introduction: Different Type of Grippers



Suction-type gripper



Parallel gripper



Dexterous hand

Introduction: Manipulation Tasks

- Grasping
- Object relocation
- In-hand manipulation
- Tool use
 - Hammers, scissors ...



Only **Dexterous hand** has the potential for **complex manipulation tasks**

Hardware Example: Shadow Hand



Shadow hand

OVERVIEW

Degrees of Freedom (DOF)

Overall Weight

No. of Fingers

No. of Actuators

No. of Finger Joints

No. of Thumb Joints

No. of Wrist Joints

No. of Tip Pressure Sensors

No. of Tendon Load Sensors

SHADOW DEXTEROUS HAND

20 actuated DOF and a further 4 under-actuated movements for a total of 24 joints

4.3 Kg

4 + 1 (thumb)

20 DC motors

4

5

2

5

40

SHADOW DEXTEROUS HAND LITE

13 DOF across 16 joints

2.4 Kg

3 + 1 (thumb)

13 DC motors

4

4

0

4

26

SHADOW DEXTEROUS HAND EXTRA LITE

10 DOF across 12 joints

2.1 Kg

2 + 1 (thumb)

10 DC motors

4

4

0

3

20

SHADOW DEXTEROUS HAND SUPER LITE

7 DOF across 8 joints

1.8 Kg

1 + 1 (thumb)

7 DC motors

4

4

0

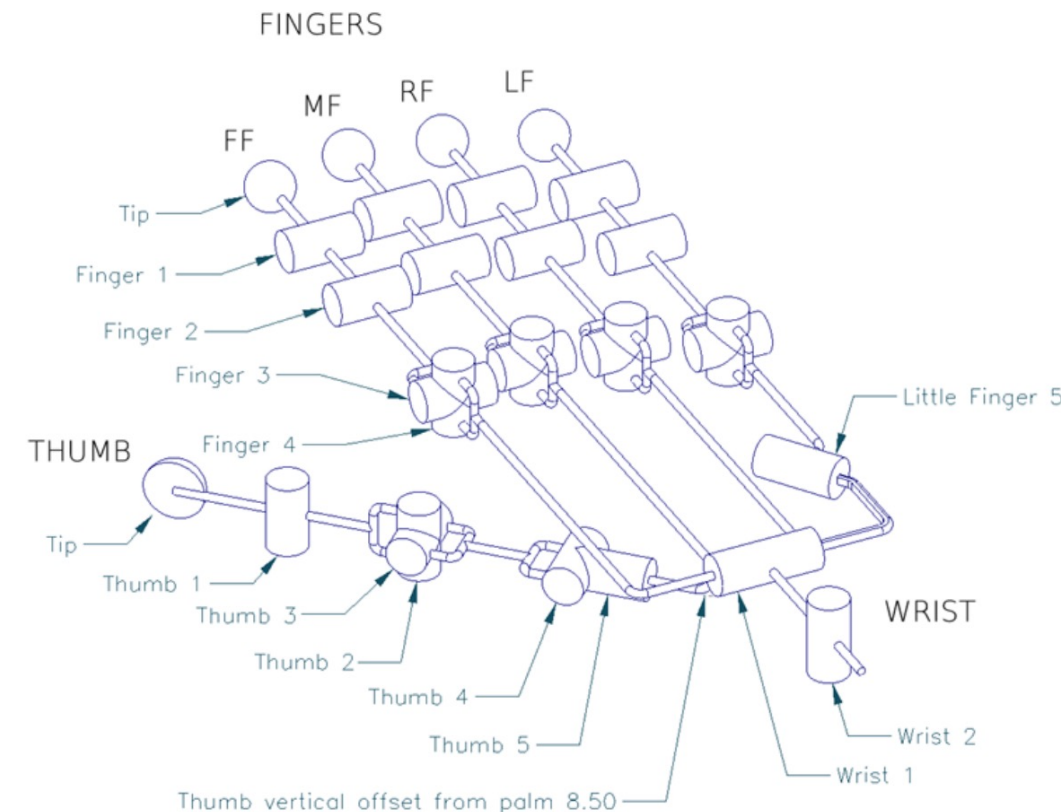
2

14

Hardware Example: Shadow Hand

2.3 Kinematic structure

The Dexterous Hand kinematics are optimized to be as close as possible (within engineering constraints) to the kinematics of the human hand.



Joint(s)	Degrees		Radians		Notes
	Min	Max	Min	Max	
FF1, MF1, RF1, LF1	0	90	0	1.571	Coupled
FF2, MF2, RF2, LF2	0	90	0	1.571	
FF3, MF3, RF3, LF3	-15	90	-0.262	1.571	
FF4, MF4, RF4, LF4	-20	20	-0.349	0.349	
LF5	0	45	0	0.785	
TH1	-15	90	-0.262	1.571	
TH2	-30	30	-0.524	0.524	
TH3	-12	12	-0.209	0.209	
TH4	0	70	0	1.222	
TH5	-60	60	-1.047	1.047	
WR1	-40	28	-0.698	0.489	
WR2	-28	8	-0.489	0.140	

Challenge 1: Sample Inefficient

- Sample inefficient for RL algorithms
 - Extreme high Degree-of-Freedom (e.g., 20 to 30 DoF)

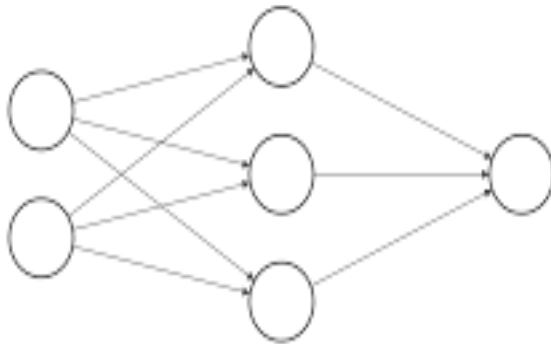
RL for continuous Control



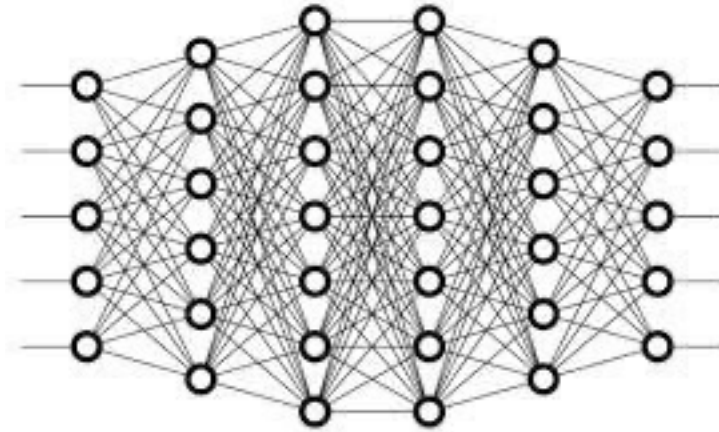
✗ Prohibitively large number of samples and safety concerns

Challenge 2: Visual Observation

- Sample inefficient for RL algorithms
- Visual observation
 - High dimension (>1000 dim) and partial
 - Hard to learn vision backbone



small backbone:
no enough capacity



large backbone:
hard to optimize

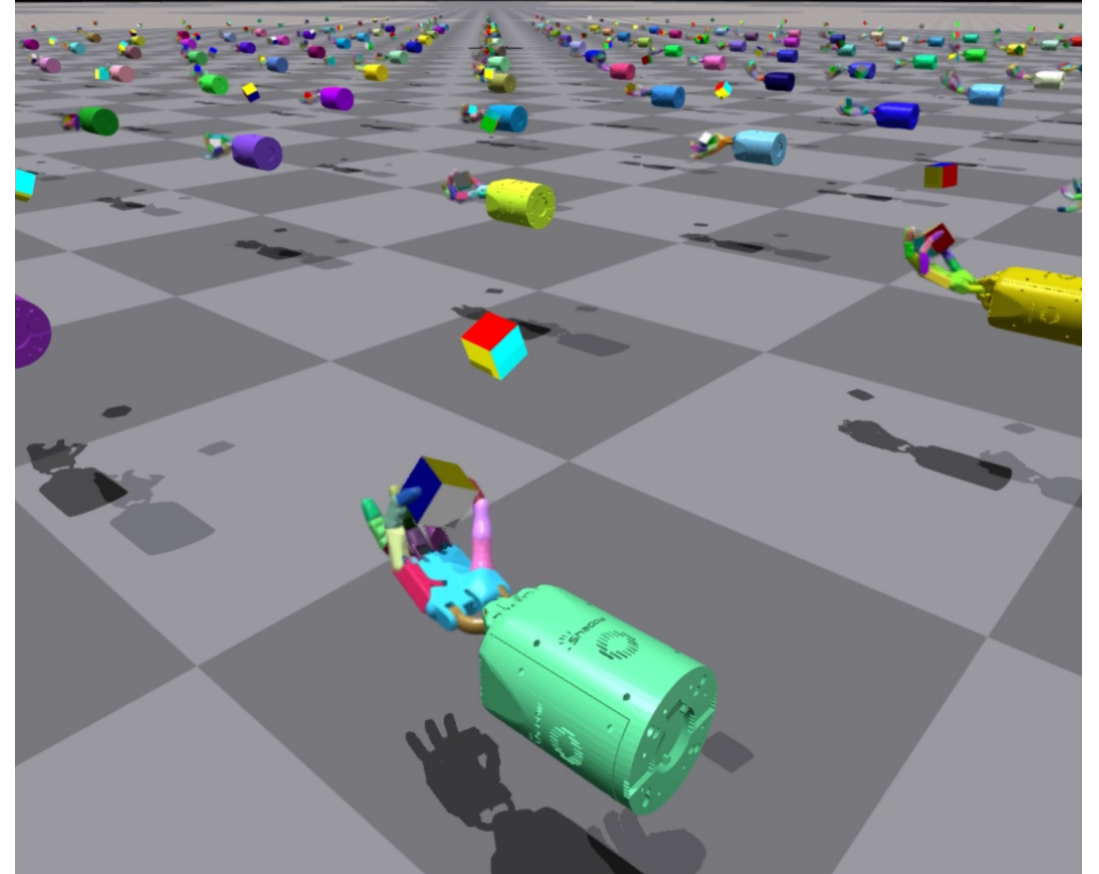
Challenge 3: Sim-to-Real Gap

- Sample inefficient for RL algorithms
- Visual observation
- Sim-to-real gap
 - Directly learn in real world is unaffordable
 - Contact-rich task is hard for accurate simulation

Sim-to-Real Gap



Google robot “farm”



Simulation
(Isaac Gym)

Research on Dexterous Grasping

Synthesizing Diverse and Physically Stable Grasps with Arbitrary Hand Structures using Differentiable Force Closure Estimator

Tengyu Liu¹, Zeyu Liu¹, Ziyuan Jiao¹, Yixin Zhu^{2,3,†}, and Song-Chun Zhu^{2,3,4}

R-AL 2021

DexGraspNet: A Large-Scale Robotic Dexterous Grasp Dataset for General Objects Based on Simulation

Ruicheng Wang^{1*}, Jialiang Zhang^{1*}, Jiayi Chen^{1,2}, Yinzhen Xu^{1,2}, Puhao Li^{2,3}, Tengyu Liu², He Wang^{1†}

ICRA 2023

Grasp Data: DexGraspNet



Force closure grasping data synthesis: Solved ✓

What We can Do: Grasp Generation

UniDexGrasp: Universal Robotic Dexterous Grasping via Learning Diverse Proposal Generation and Goal-Conditioned Policy

Yinzhen Xu^{*1,2,3}, Weikang Wan^{*1,2}, Jialiang Zhang^{*1,2}, Haoran Liu^{*1,2}, Zikang Shan¹,
Hao Shen¹, Ruicheng Wang¹, Haoran Geng^{1,2}, Yijia Weng⁴, Jiayi Chen¹,
Tengyu Liu³, Li Yi⁵, He Wang^{†1,2}

¹ Center on Frontiers of Computing Studies, Peking University ² School of EECS, Peking University

³ Beijing Institute for General AI ⁴ Stanford University ⁵ Tsinghua University

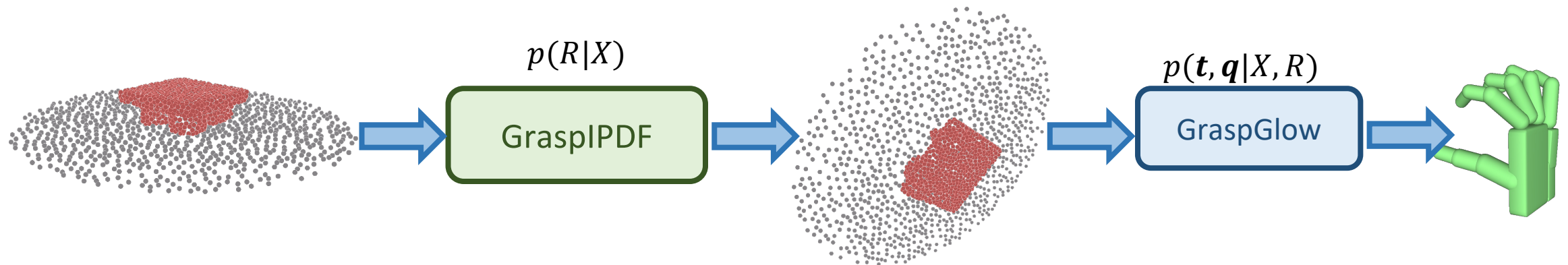
<https://pku-epic.github.io/UniDexGrasp/>

CVPR 2023

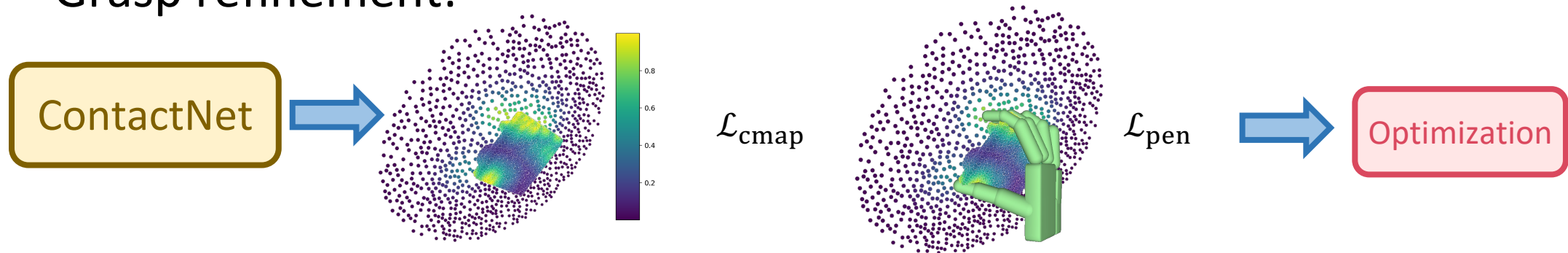


Grasp Generation: UniDexGrasp

- Grasp Generation: $p(R, t, q|X) = p(R|X)p(\mathbf{t}, \mathbf{q}|X, R)$



- Solve mode collapse in diverse grasp generation
- Grasp refinement:



Grasp Generation: UniDexGrasp

- Pros:
 - Model diverse grasp distribution even with partial observation
- Cons:
 - Raw output of network has penetration and inexact contact
 - Refinement requires full observation and iterative optimization

State of Research: Grasp Policy in Simulation

UniDexGrasp: Universal Robotic Dexterous Grasping via Learning Diverse Proposal Generation and Goal-Conditioned Policy

Yinzhen Xu^{*1,2,3}, Weikang Wan^{*1,2}, Jialiang Zhang^{*1,2}, Haoran Liu^{*1,2}, Zikang Shan¹,
Hao Shen¹, Ruicheng Wang¹, Haoran Geng^{1,2}, Yijia Weng⁴, Jiayi Chen¹,
Tengyu Liu³, Li Yi⁵, He Wang^{†1,2}

¹ Center on Frontiers of Computing Studies, Peking University ² School of EECS, Peking University

³ Beijing Institute for General AI ⁴ Stanford University ⁵ Tsinghua University

<https://pku-epic.github.io/UniDexGrasp/>

CVPR 2023

UniDexGrasp++: Improving Dexterous Grasping Policy Learning via Geometry-aware Curriculum and Iterative Generalist-Specialist Learning

Weikang Wan^{1*} Haoran Geng^{1,3*} Yun Liu²
Zikang Shan¹ Yaodong Yang^{1,3} Li Yi² He Wang^{1†}

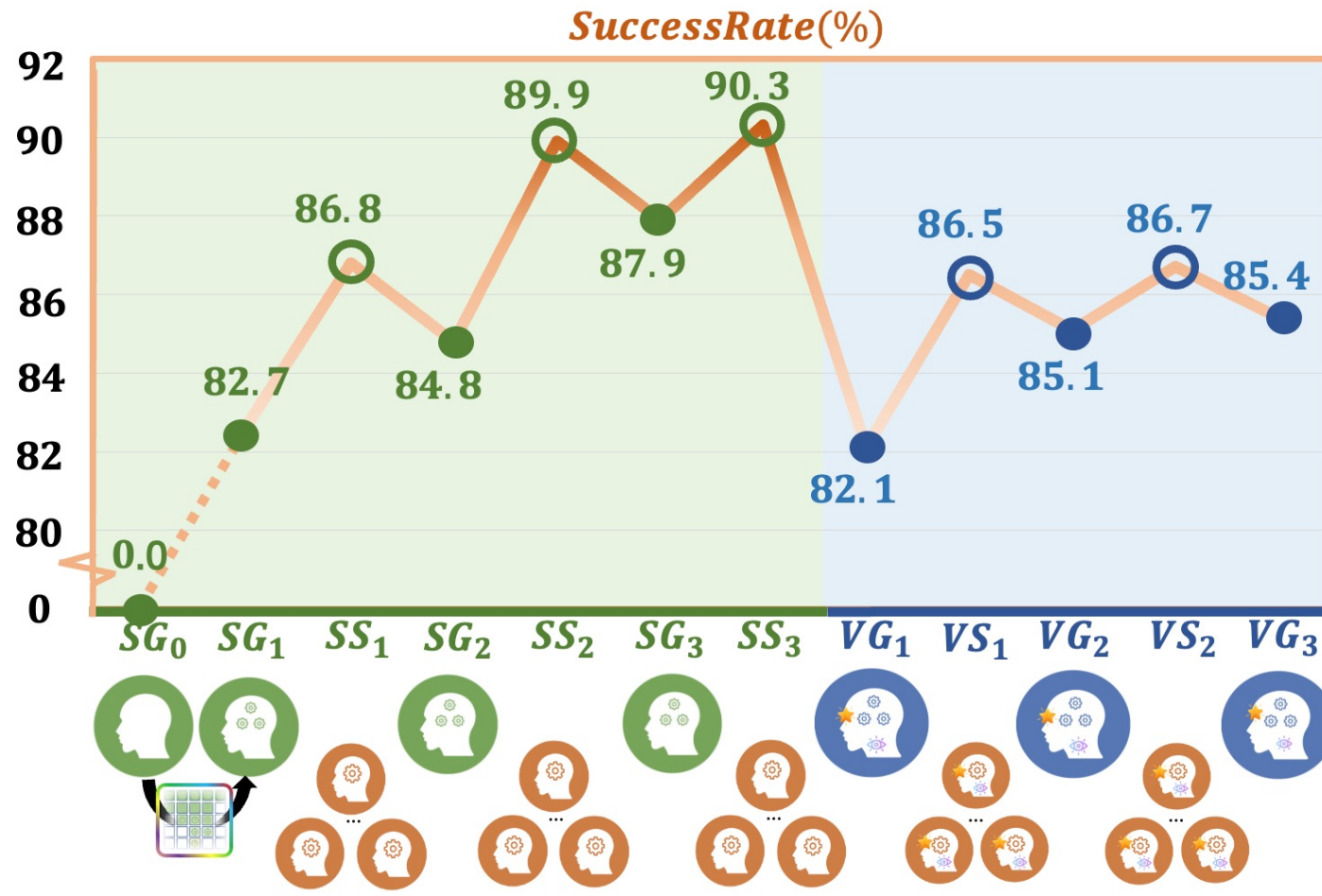
¹Peking University ²Tsinghua University

³Beijing Institute for General Artificial Intelligence

ICCV 2023

Grasp Policy in Simulation: UniDexGrasp++

- Key idea: use geometry-aware iterative generalist-specialist learning (**curriculum**) to boost generalizability



State-based

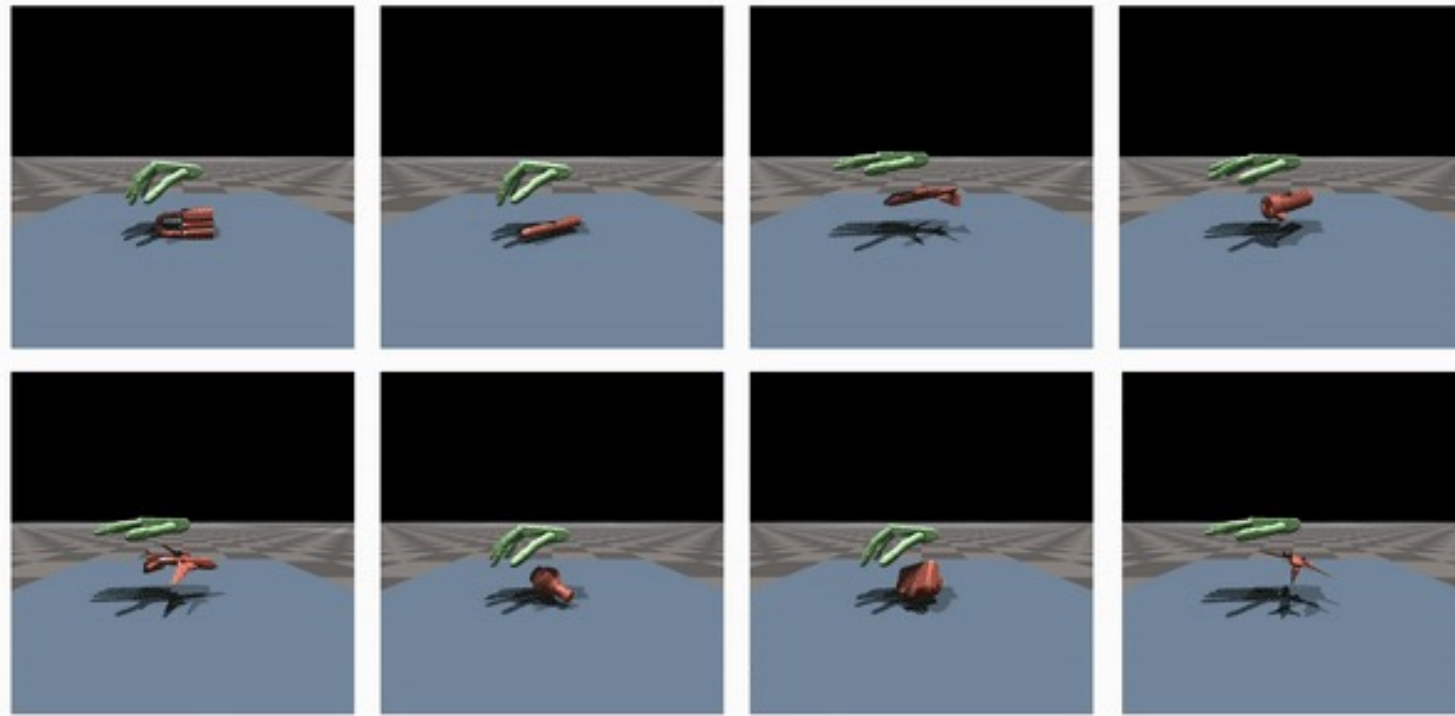
- Step1: GeoCurriculum
- Step2: GiGSL

Vision-Based

- Step3: GiGSL

Grasp Policy in Simulation: UniDexGrasp++

- 78.2% succ. for novel obj.
- Completely solves dexterous grasping in the simulator



- Can not directly sim2real (to be analyzed later)

State of Research: Grasp Policy in Real World

Learning Robust Real-World Dexterous Grasping Policies via Implicit Shape Augmentation

Zoey Qiuyu Chen^{1*} Karl Van Wyk² Yu-Wei Chao² Wei Yang² Arsalan Mousavian²

Abhishek Gupta¹ Dieter Fox^{1,2}

¹University of Washington ²NVIDIA

{qiuyuc, abhgupta, fox}@cs.washington.edu

{kvanwyk, ychao, weiy, amousavian}@nvidia.com

(ISAGrasp)

CoRL 2022

DexPoint: Generalizable Point Cloud Reinforcement Learning for Sim-to-Real Dexterous Manipulation

CoRL 2022

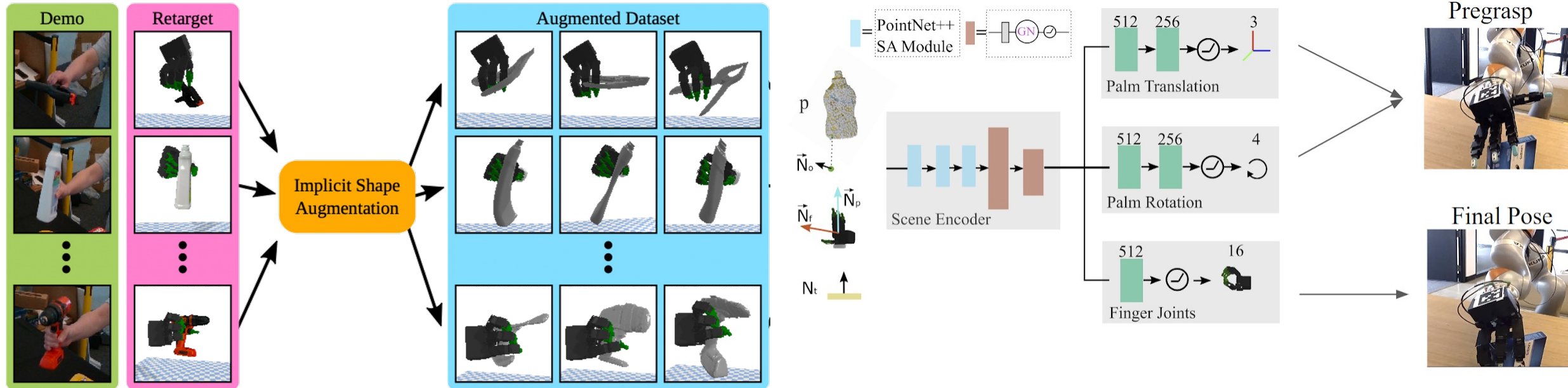
Yuzhe Qin^{*1} Binghao Huang^{*1} Zhao-Heng Yin²

Hao Su¹ Xiaolong Wang¹

UC San Diego¹ HKUST²

Grasp Policy in Real World: ISAGrasp

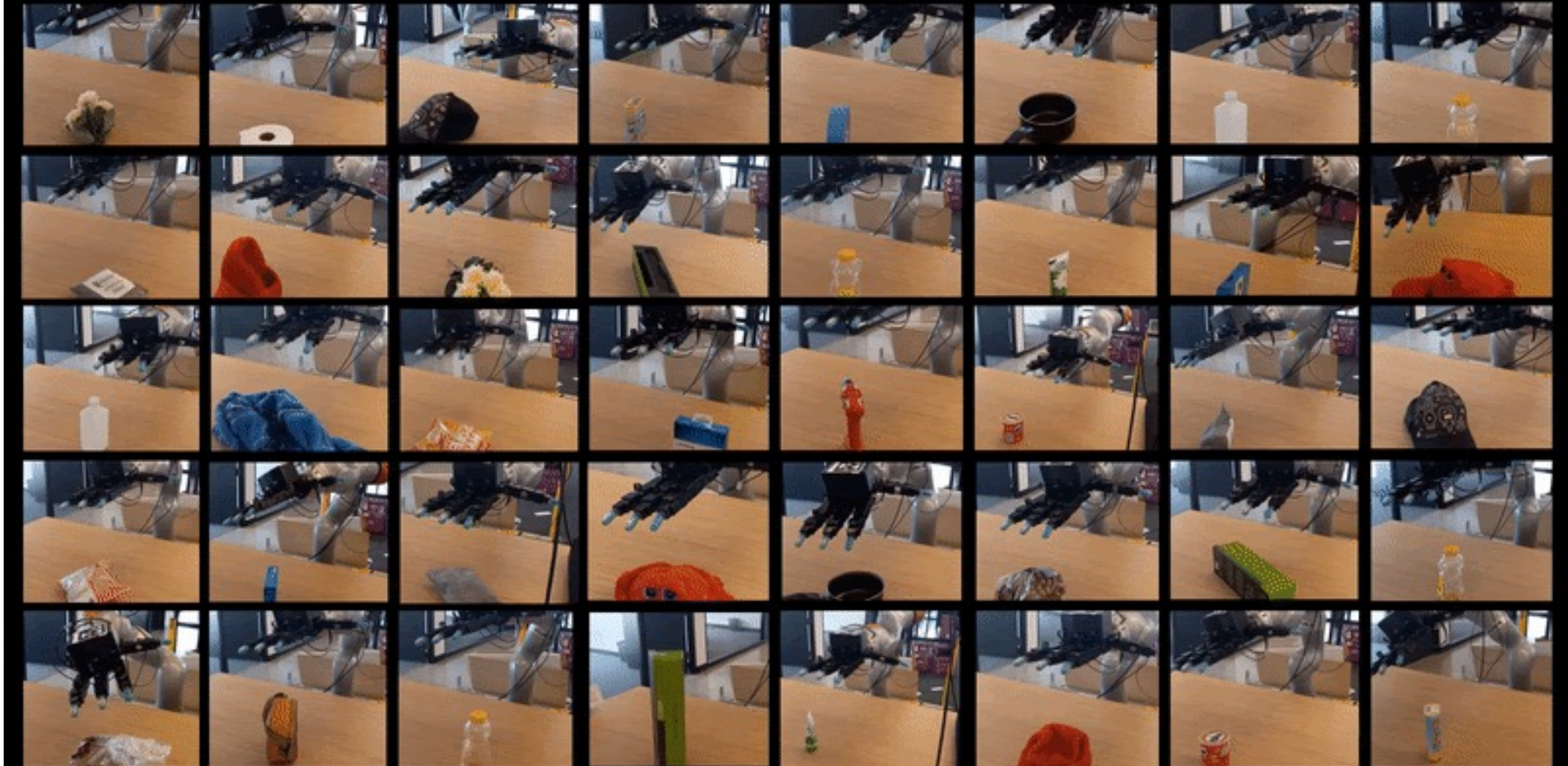
- Step1: Retarget human grasp dataset and augment with ISA



- Step2: Train grasp prediction network
- Step3: Move the wrist to target 6d pose, and move finger joints

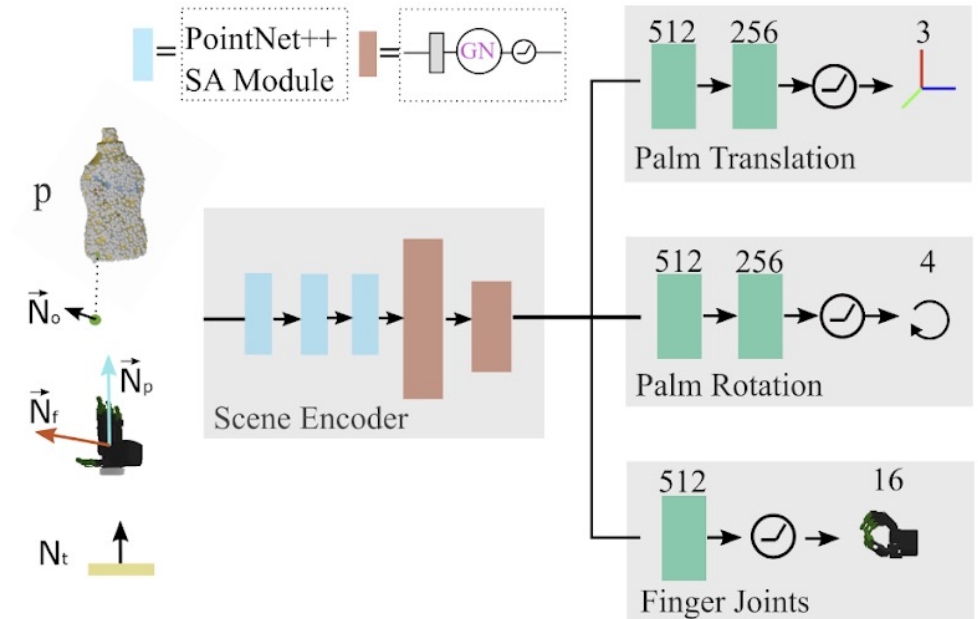
Grasp Policy in Real World: ISAGrasp

79% success rate



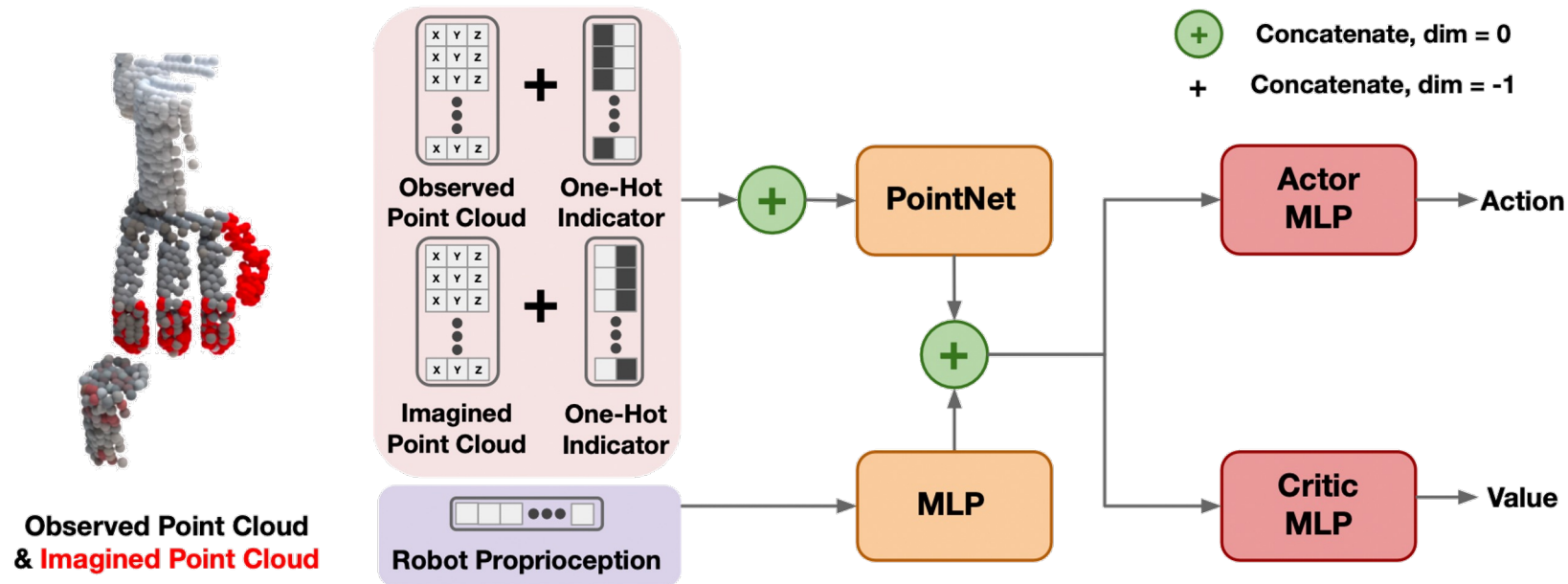
Grasp Policy in Real World: ISAGrasp

- Contributions
 - Data augmentation method
- Problems
 - Flexibility and partially observability
 - Network is way too simple
 - Lacks evaluations on grasp quality
 - Dexterity is doubtable
 - Not open-sourced



Grasp Policy in Real World: DexPoint

- Train PPO directly on visual observation
- Key argument: imagined point cloud + noise -> direct sim2real



Grasp Policy in Real World: DexPoint

- Trained on 10 bottles
- Claims 73% success rate on 10 novel objects in the real world
- Sim2real with only point cloud tricks
- Problem
 - Experiments are not thorough
- Why their RL can sim2real and ours can not?



Dexterous Manipulation

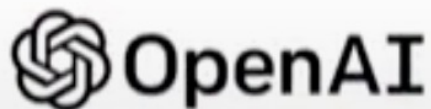
- Challenge 1: sample inefficient
 - RL + Imitation learning
 - ...
- Challenge 2: visual observation
 - Teacher-student distillation
 - ...
- Challenge 3: sim-to-real gap
 - Domain randomization
 - Model-based RL
 - ...

Topic 1: RL + IL

Learning Complex Dexterous Manipulation with Deep Reinforcement Learning and Demonstrations

Aravind Rajeswaran*, Vikash Kumar*, Abhishek Gupta, Giulia Vezzani,
John Schulman, Emanuel Todorov, Sergey Levine

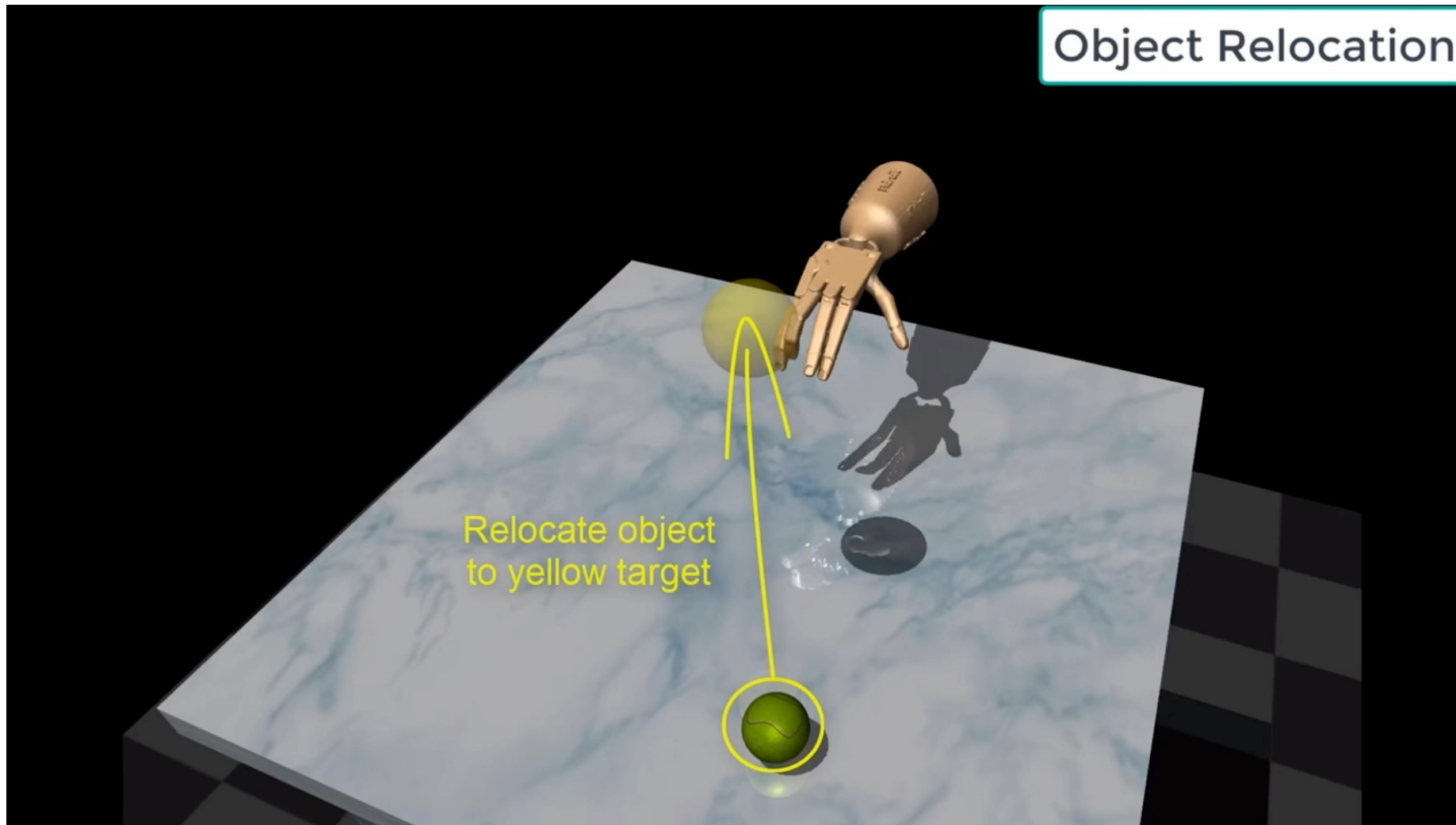
(* equal contributions)



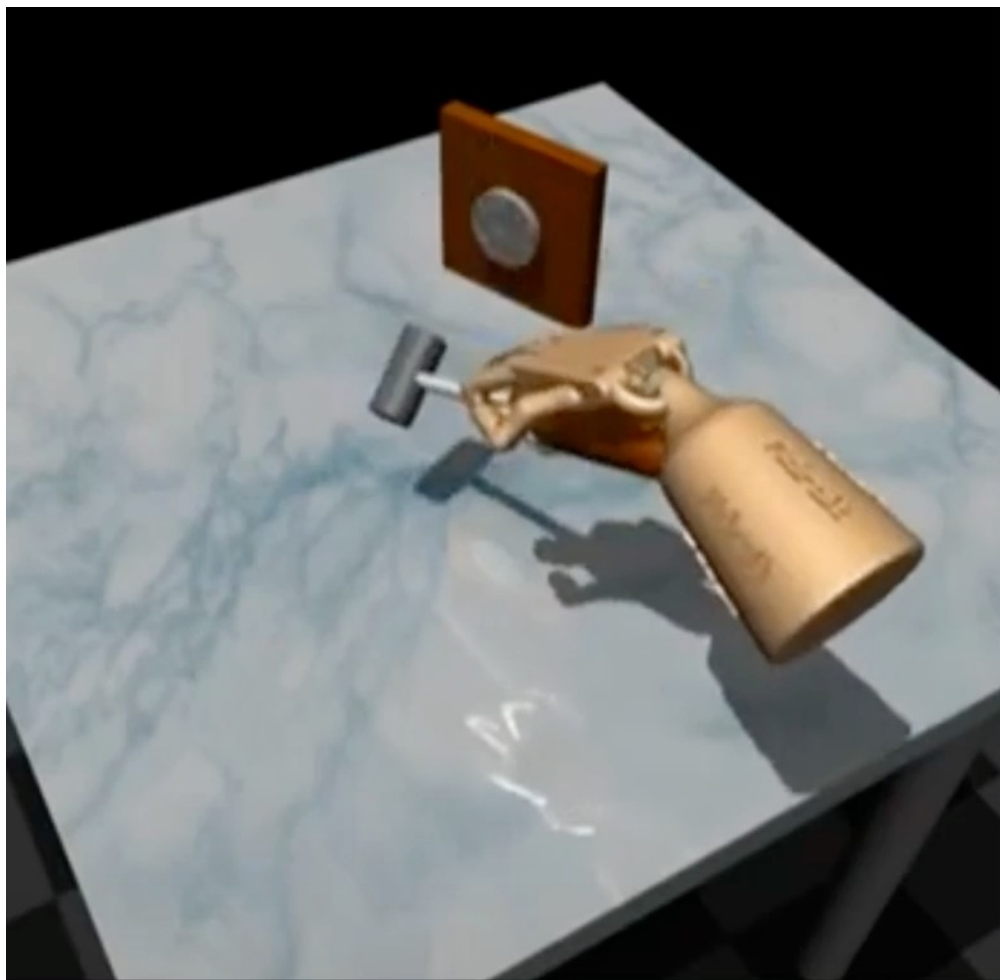
PAUL G. ALLEN SCHOOL
OF COMPUTER SCIENCE & ENGINEERING



Results in simulation



Only RL or Only IL



Model-free policy gradient



Behavior cloning

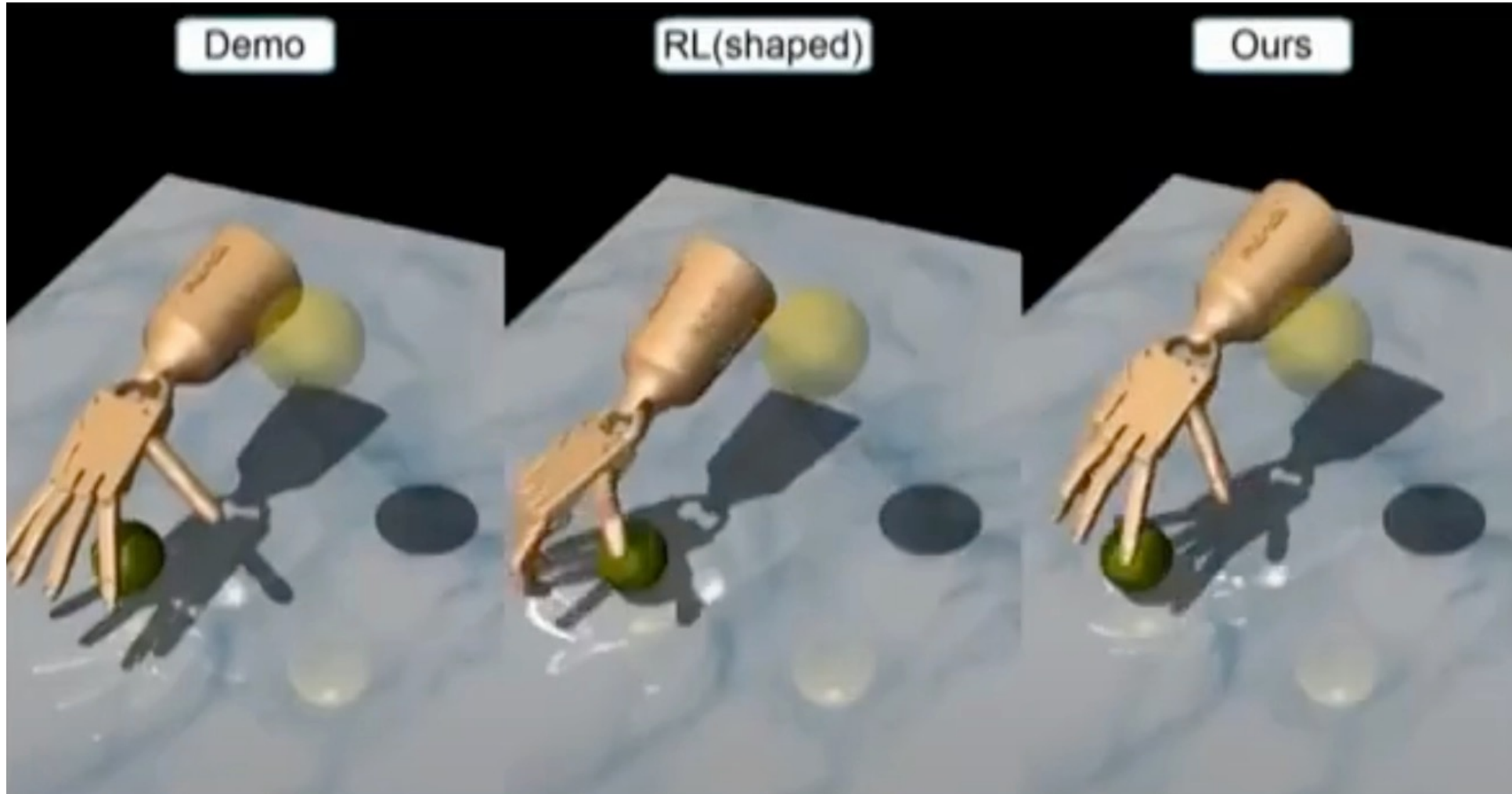
DAPG: Policy Gradient + Imitation

$$\eta(\pi_\theta) = \underbrace{\mathbb{E}_{\pi_\theta} [Q^{\pi_\theta}(s, a)]}_{\text{Reinforcement Learning}} + \lambda \underbrace{\mathbb{E}_{\pi^e} [\ln \pi_\theta(a|s)]}_{\text{Behavior Cloning}}$$

$$\theta_{k+1} = \theta_k + \underbrace{\alpha F_{\theta_k}^{-1} \nabla_\theta \eta(\pi_{\theta_k})}_{\text{natural gradient}} \quad \lambda \rightarrow 0 \text{ as } k \rightarrow \infty$$

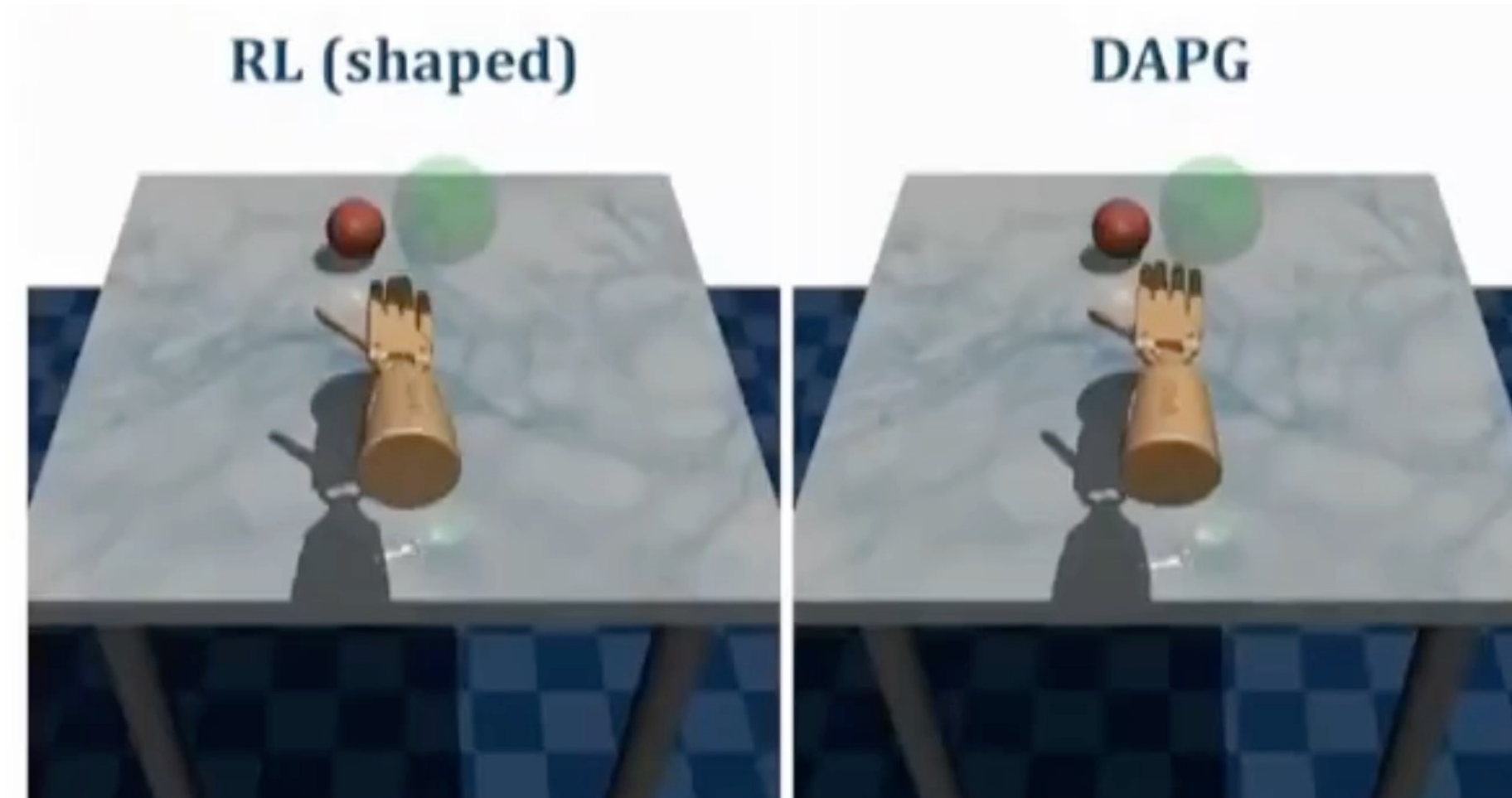
DAPG Benefits

Better quality and more “human-like” motions



DAPG Benefits

More robust to perturbations in the environment



Topic 2: Teacher-Student Distillation

A System for General In-Hand Object Re-Orientation

Tao Chen, Jie Xu, Pulkit Agrawal

MIT Improbable AI Lab



Massachusetts
Institute of
Technology

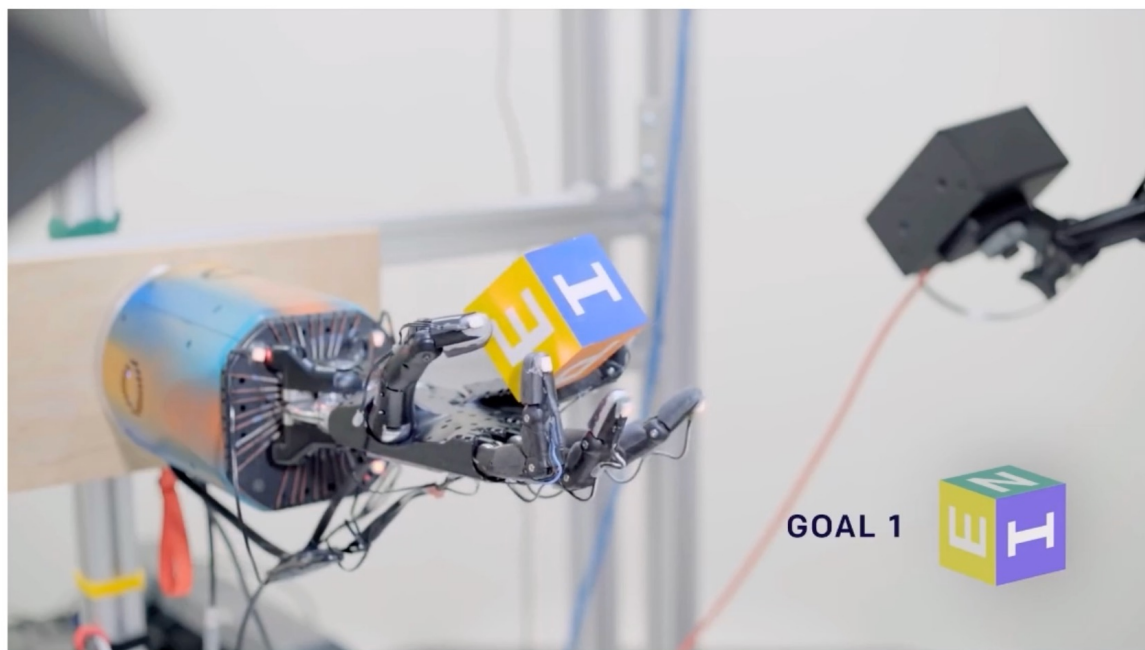


Comparison with Prior Work

Prior Work

Reorient one object, limited generalization to different objects

Only works when the **hand is facing upward**

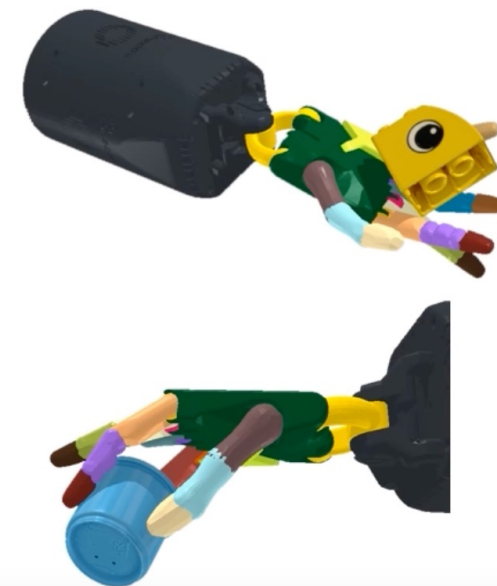


Their work

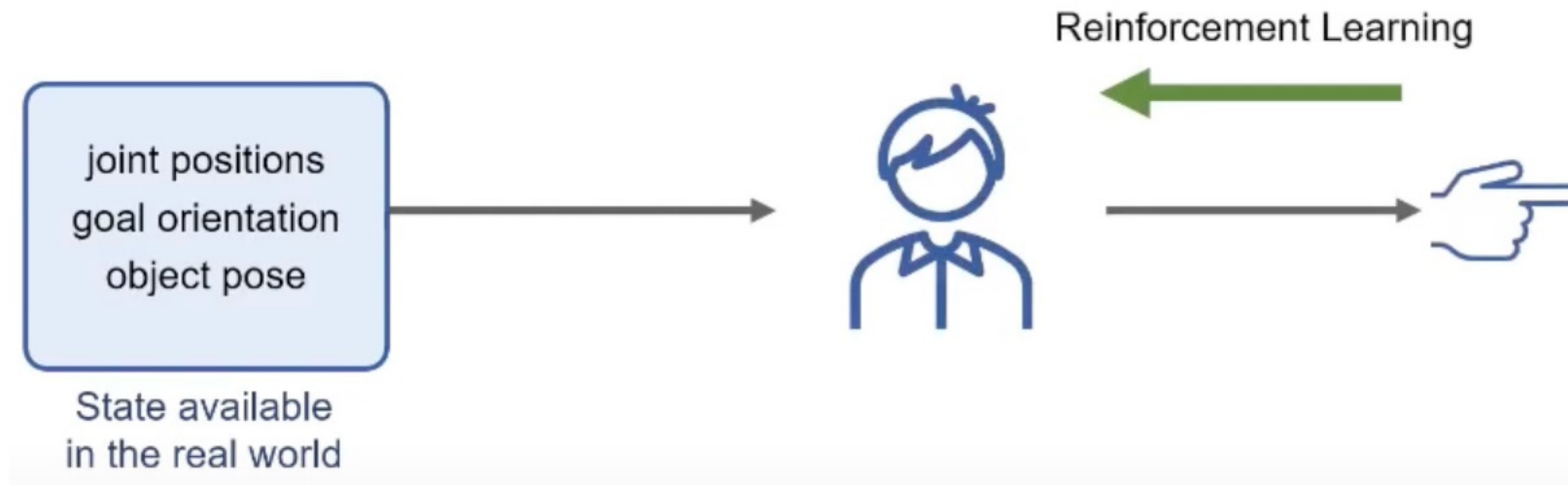
Reorient more than 2000 geometrically different objects



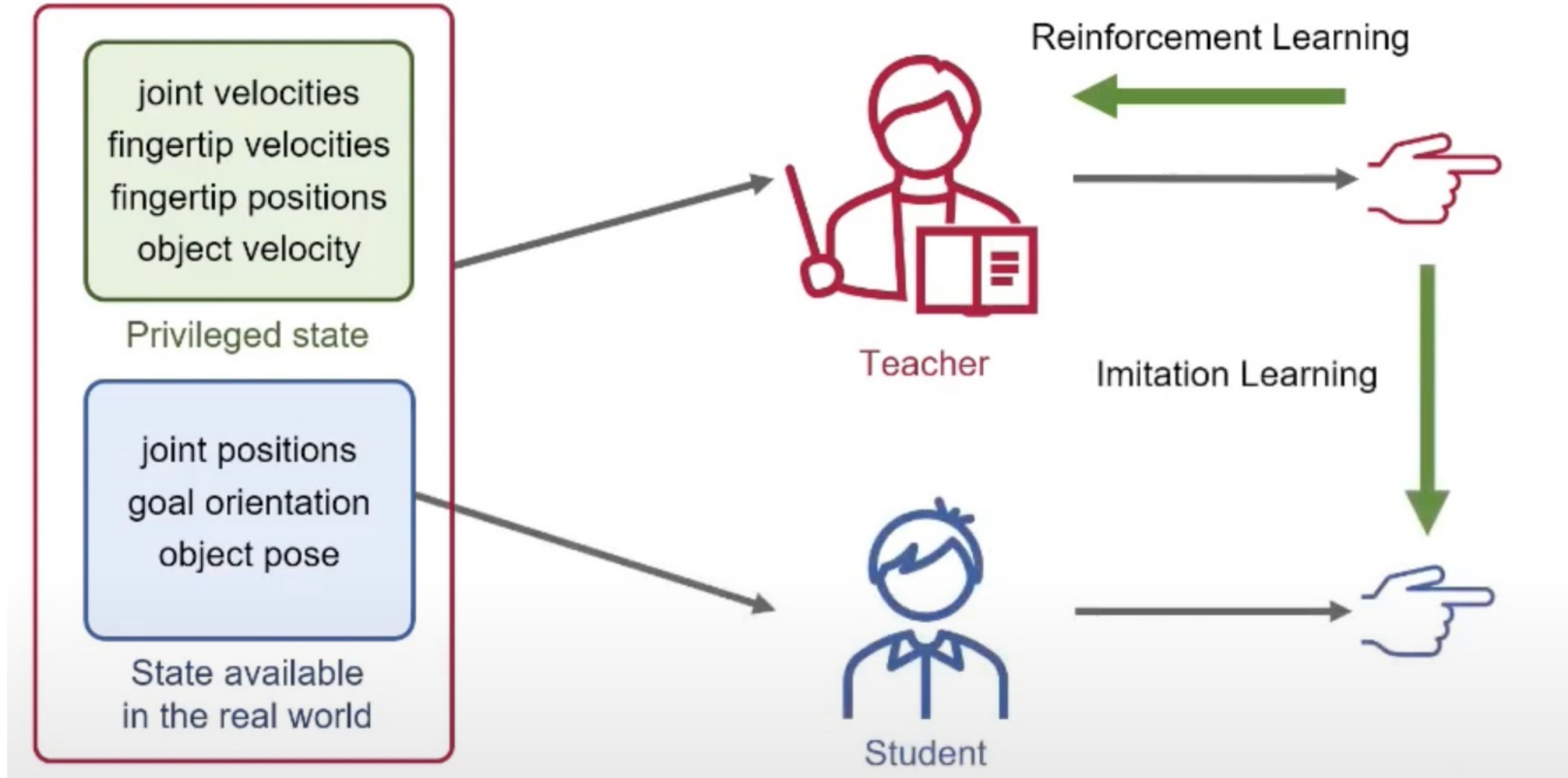
Hand faces upward and downward



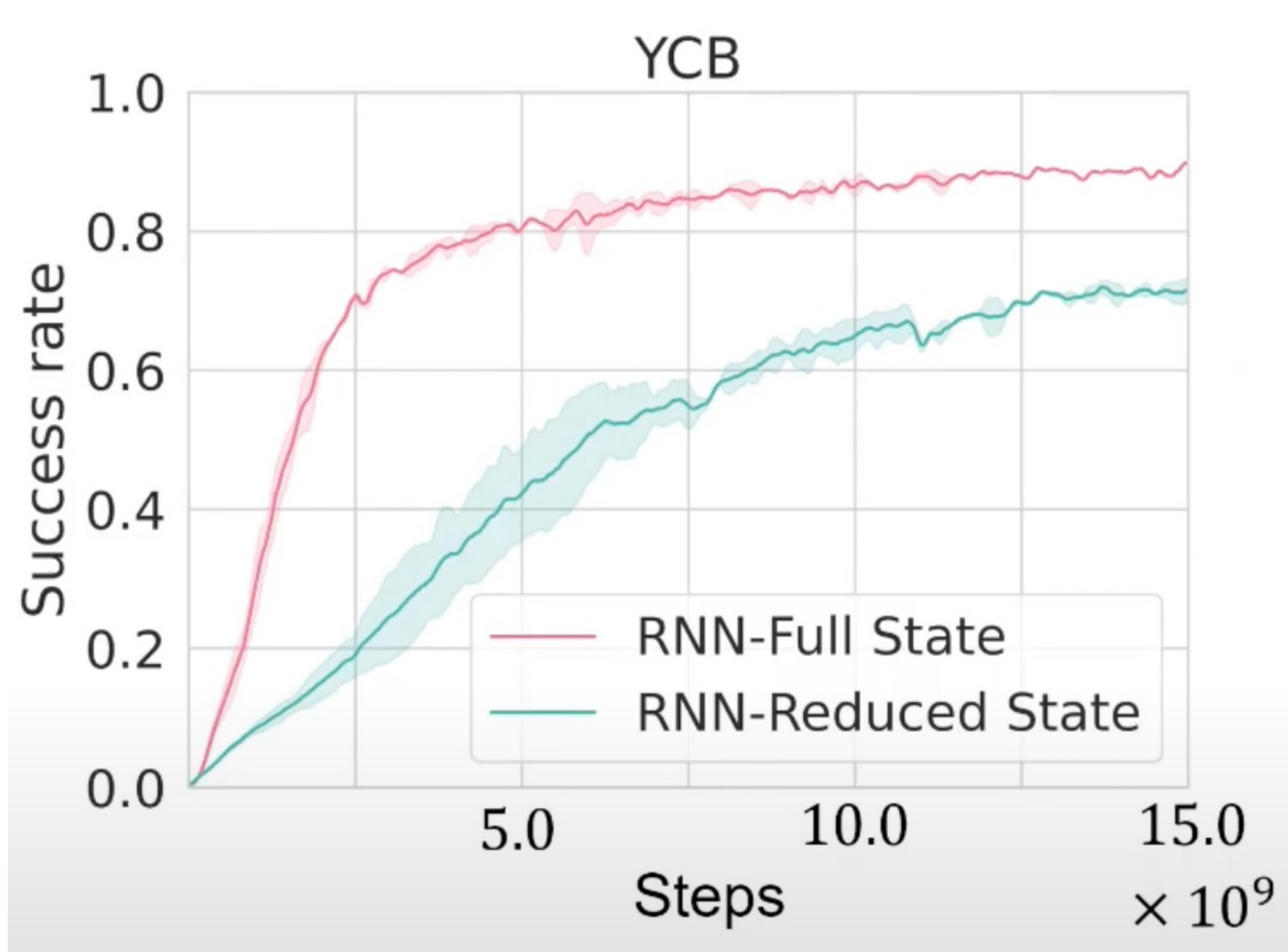
State-based RL



State-based Teacher-Student

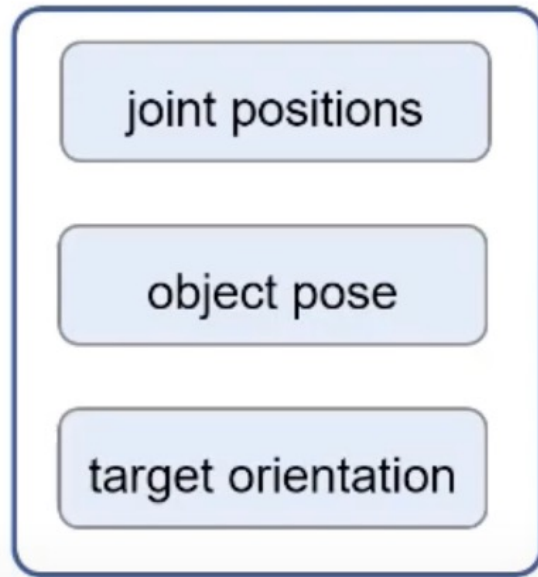


Experiment

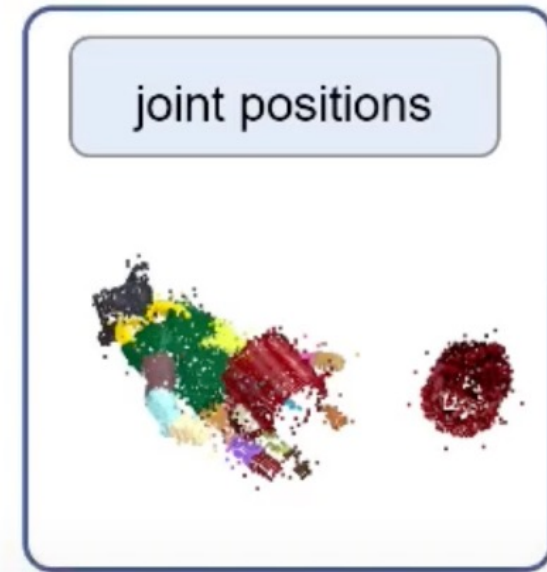


Vision-based Student

Overcome the object pose assumption



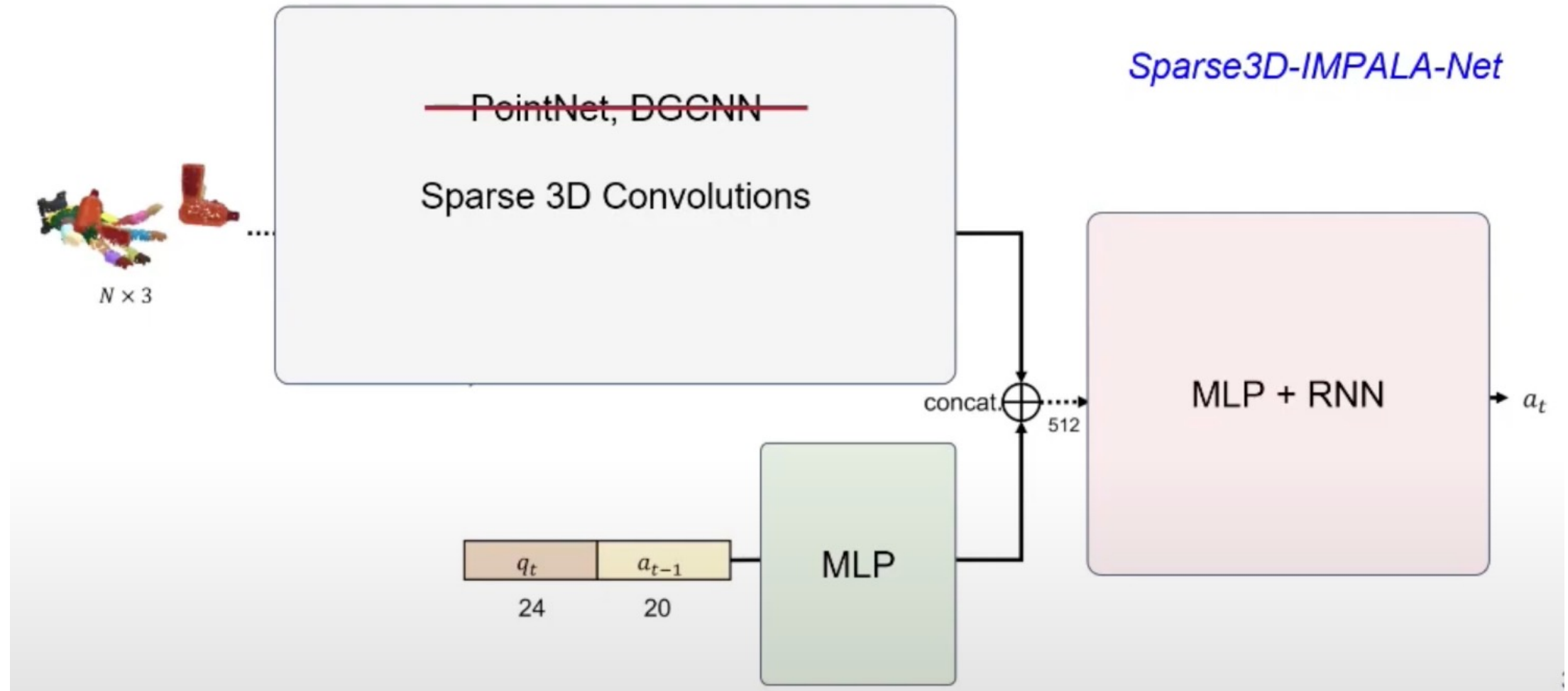
Student Observations
(Low-Dimensional)



Student Observations
(Point Cloud)

Vision-based Student

Student policy with vision input



Experiment

Train and test with noise on point cloud

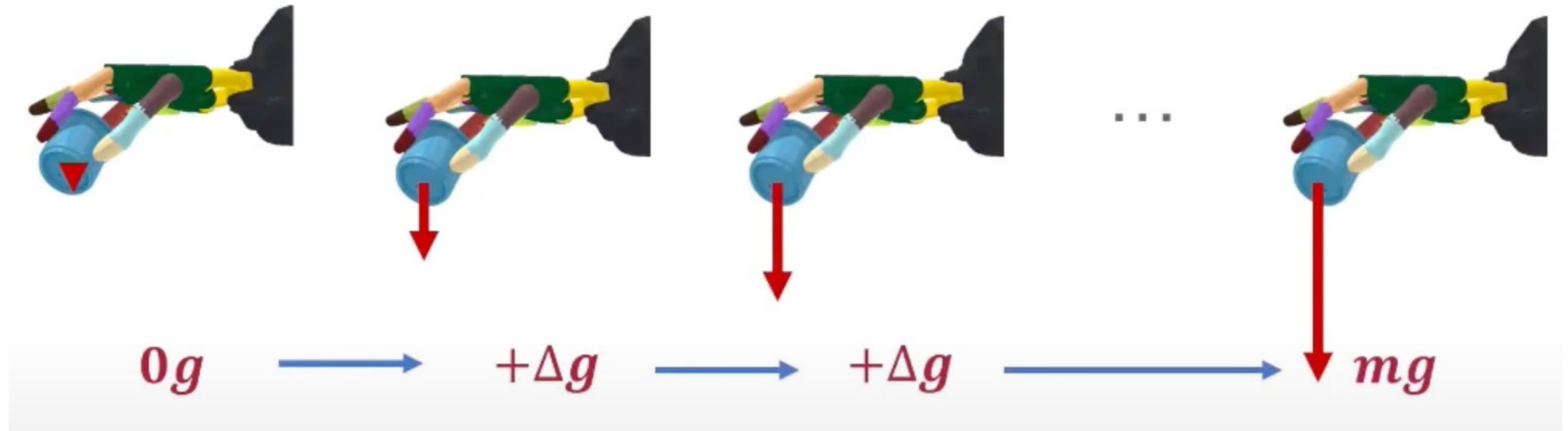
	Object	Success rate (%)
	025_mug	89.67 ± 1.2
	065-d_cups	68.32 ± 1.9
	072-b_toy_airplane	84.52 ± 1.4
	073-a_lego_duplo	58.16 ± 3.1
	073-c_lego_duplo	50.21 ± 3.7
	073-e_lego_duplo	66.57 ± 3.1

One vision policy for one object

Hand faces upward

Hand faces upward -> Hand faces downward

Gravity Curriculum



Experiment

49% relative performance improvement
with gravity curriculum

Success rate (%)	Student	Student + Gravity Curriculum
YCB	45.22 $\pm$ 2.1	67.33 $\pm$ 1.9



Robot Vision and Learning

— From a Perspective of Embodied AI

Next week: Lecture 14, Robotic Tasks IV

