



Introduction to Embodied AI

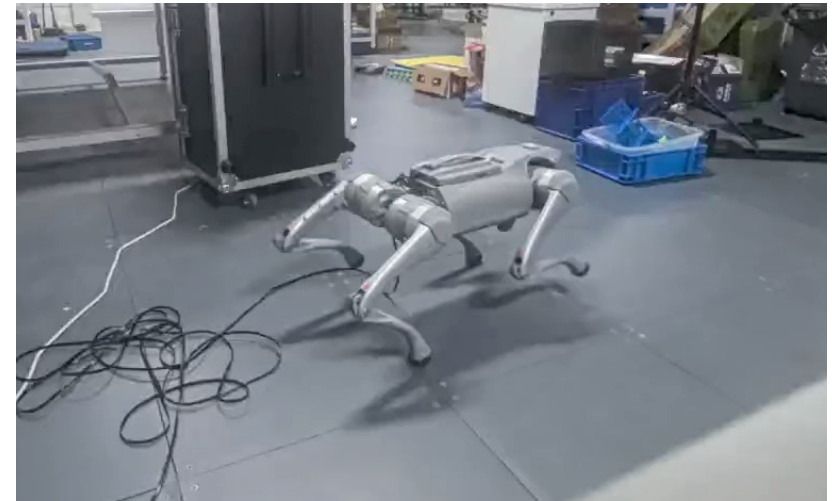
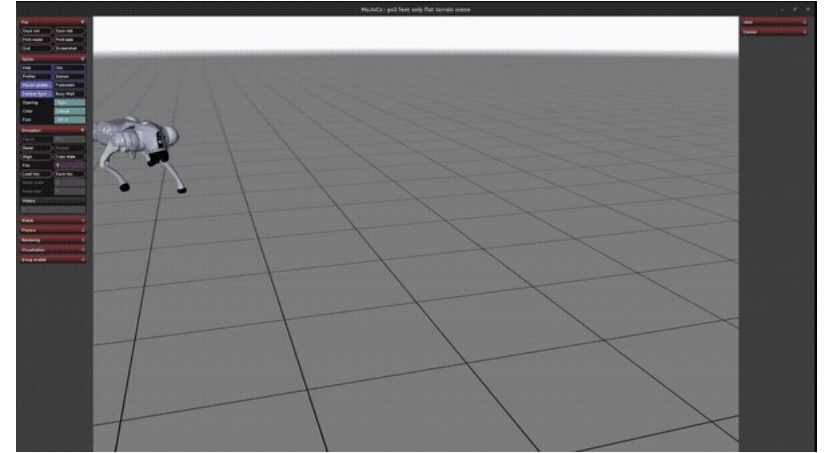
Lecture 12 - Robotic Task II

Prof. He Wang



Assignment 3: Quadrupedal Locomotion

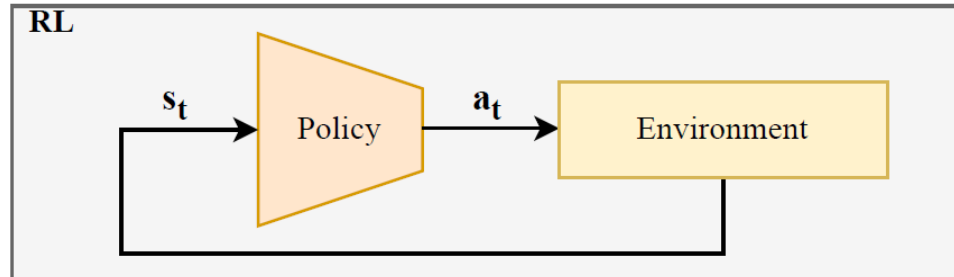
- Quadrupedal Locomotion using Unitree Go-2
 - Task 1: Implement your PPO
 - Task II: Quadrupedal Robots – Stand Up
 - Task III: Quadrupedal Robots – Move
- Each student needs to do this part independently, real experiments will be done in groups.
- Release date: 4/30 Wednesday evening
- Due date: 5/14 Wednesday 11:59PM



Locomotion

Frameworks – End-to-End Learning

- Why robotics researcher always use PPO?
 - In highly dynamic tasks, the Q-function requires in SAC / DDPG / ... exhibits sharp variations with respect to changes in action.
 - With parallel simulation, sample-efficiency of on-policy methods is no-longer a problem.



Frameworks – Curriculum Learning

Example 1: Terrain traverse

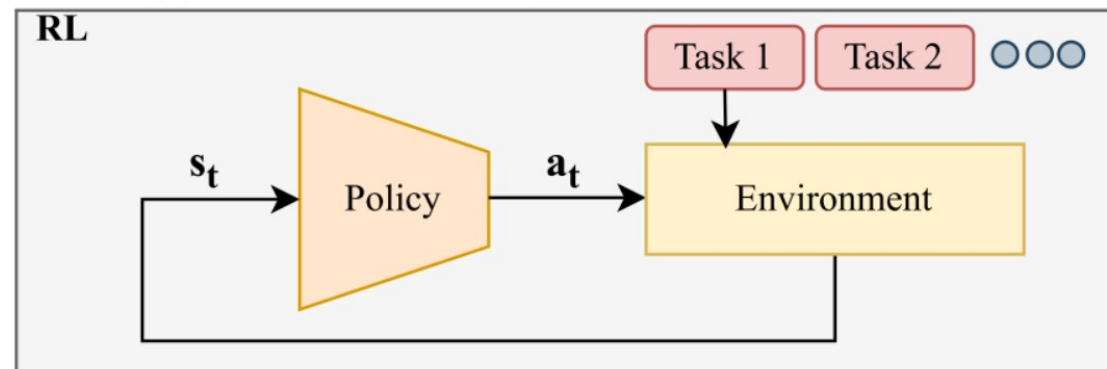
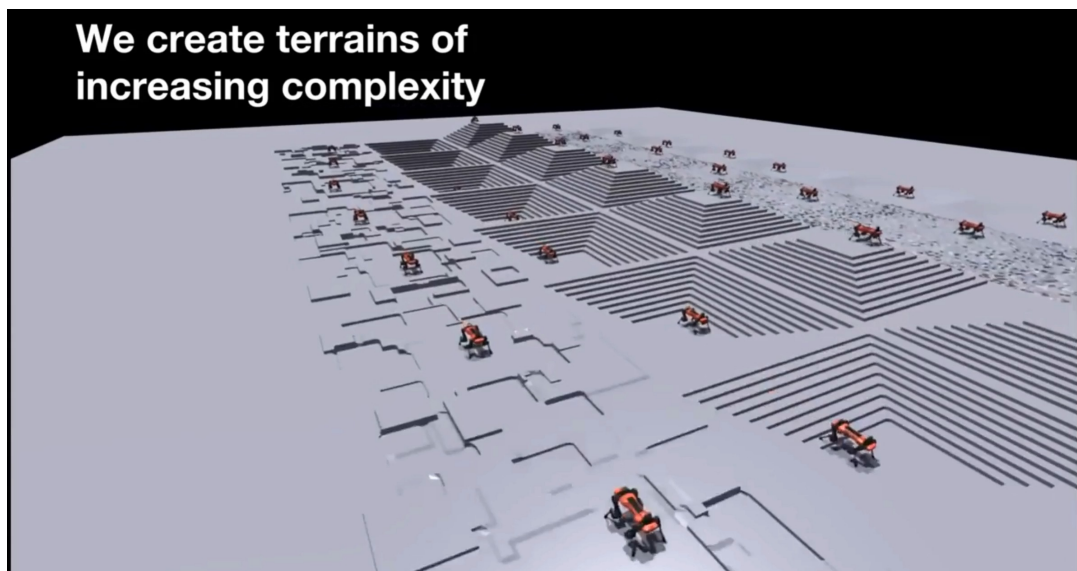
Learning to Walk in Minutes Using Massively Parallel Deep Reinforcement Learning

Nikita Rudin
ETH Zurich and NVIDIA
rudinn@ethz.ch

David Hoeller
ETH Zurich and NVIDIA
dhoeller@ethz.ch

Philipp Reist
NVIDIA
preist@nvidia.com

Marco Hutter
ETH Zurich
mahutter@ethz.com



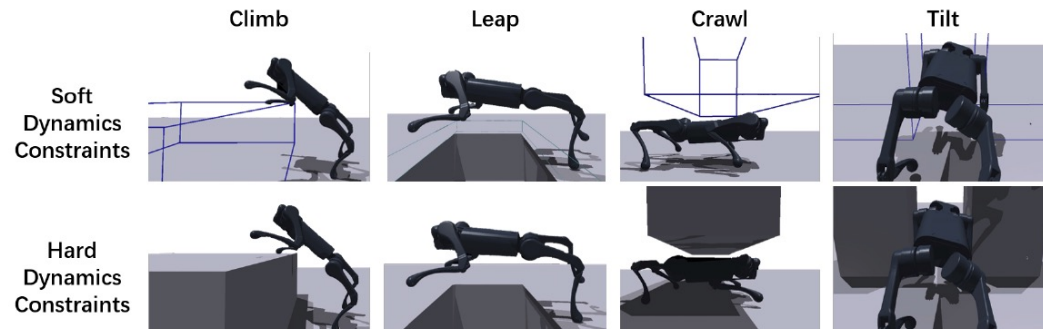
Frameworks – Curriculum Learning

Example 2: Obstacle avoidance

Robot Parkour Learning

Ziwen Zhuang^{*13} Zipeng Fu^{*2} Jianren Wang⁴ Christopher Atkeson⁴ Sören Schwertfeger³
Chelsea Finn² Hang Zhao¹⁵

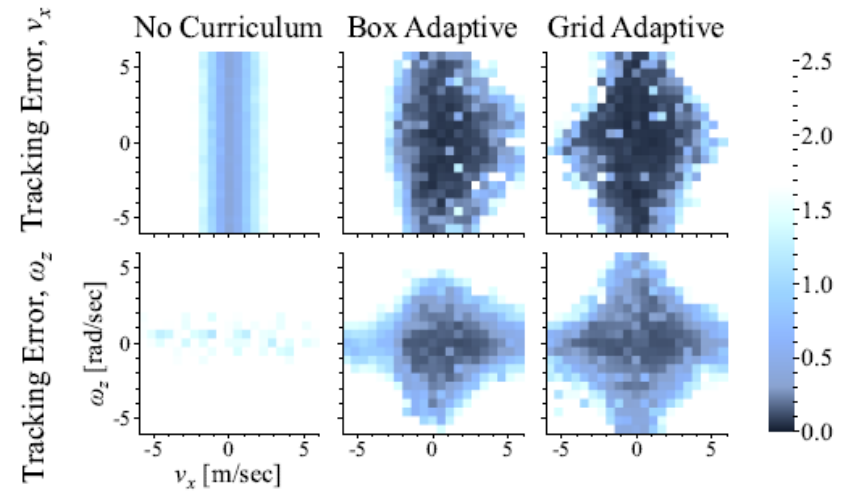
¹Shanghai Qi Zhi, ²Stanford, ³ShanghaiTech, ⁴CMU, ⁵Tsinghua, ^{*}project co-leads



Example 3: Fast moving

Rapid Locomotion via Reinforcement Learning

Gabriel B. Margolis^{*1}, Ge Yang^{*1 2}, Kartik Paigwar¹, Tao Chen¹, and Pulkit Agrawal^{1 2}
¹MIT Improbable AI Lab ²NSF AI Institute for Artificial Intelligence and Fundamental Interactions
Massachusetts Institute of Technology, Cambridge, MA 02139



Learning Frameworks – Hierarchical Learning

Example 2: Terrain traverse (vision-based)

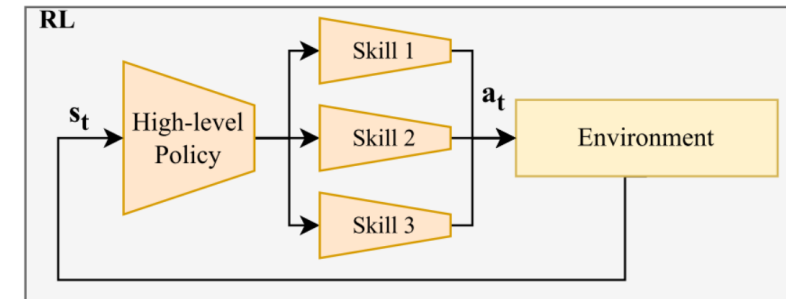
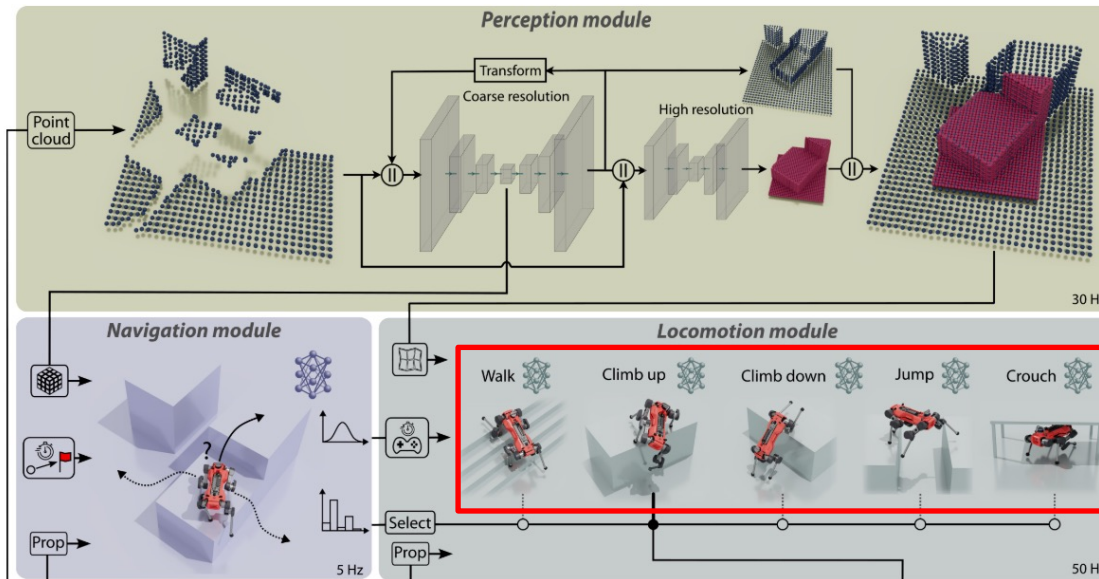
ANYmal Parkour: Learning Agile Navigation for Quadrupedal Robots

David Hoeller
ETH Zurich
NVIDIA
dhoeller@ethz.ch
Equal contribution

Nikita Rudin
ETH Zurich
NVIDIA
rudinn@ethz.ch
Equal contribution

Dhionis Sako
ETH Zurich
dsako@ethz.ch

Marco Hutter
ETH Zurich
mahutter@ethz.ch



Skill Library

Learning Frameworks – Privileged Learning

- Why some privilege information (linear velocity, base height) so important?
 - Imaging a dog jumping over a gap
 - Actually, the dog does not know where it is with only proprioception

- Some analysis

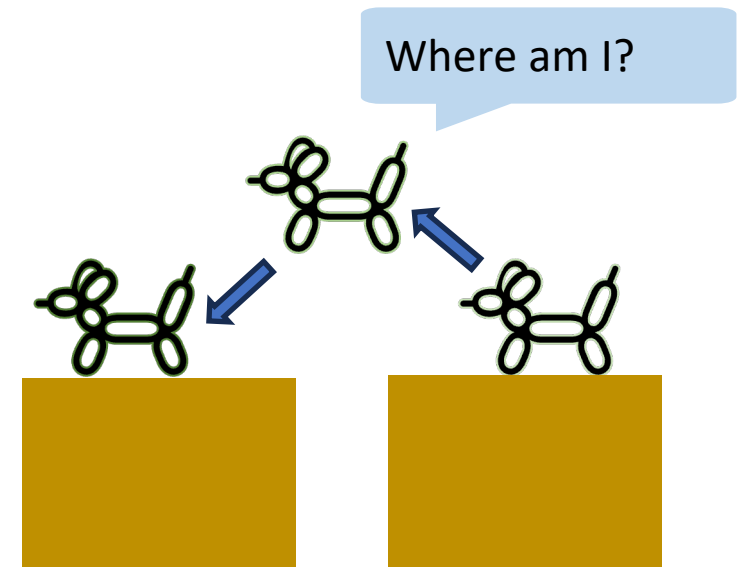
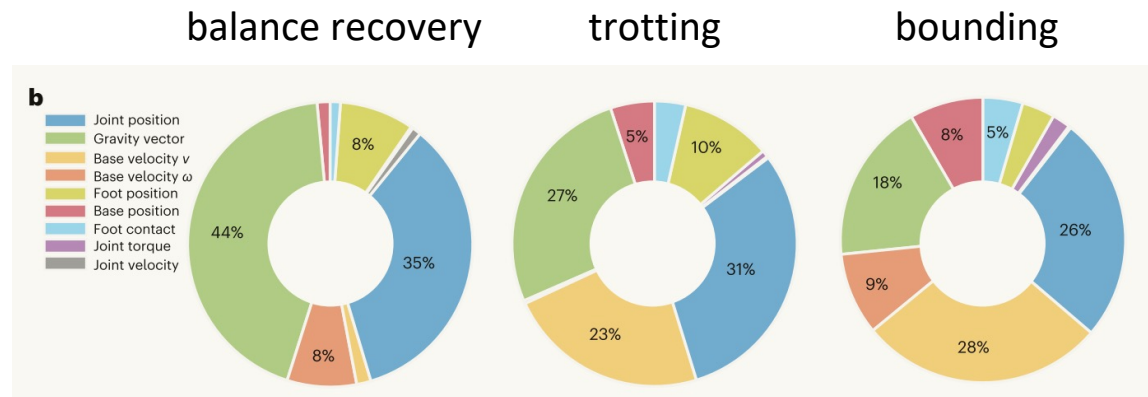
Identifying important sensory feedback for learning locomotion skills

Received: 3 December 2022

Accepted: 30 June 2023

Published online: 21 August 2023

Wanming Yu^{1,5}, Chuanyu Yang^{1,2,5}, Christopher McCreavy¹,
Eleftherios Triantafyllidis¹, Guillaume Bellegarda³, Milad Shafiee³,
Auke Jan Ijspeert³ & Zhibin Li⁴✉

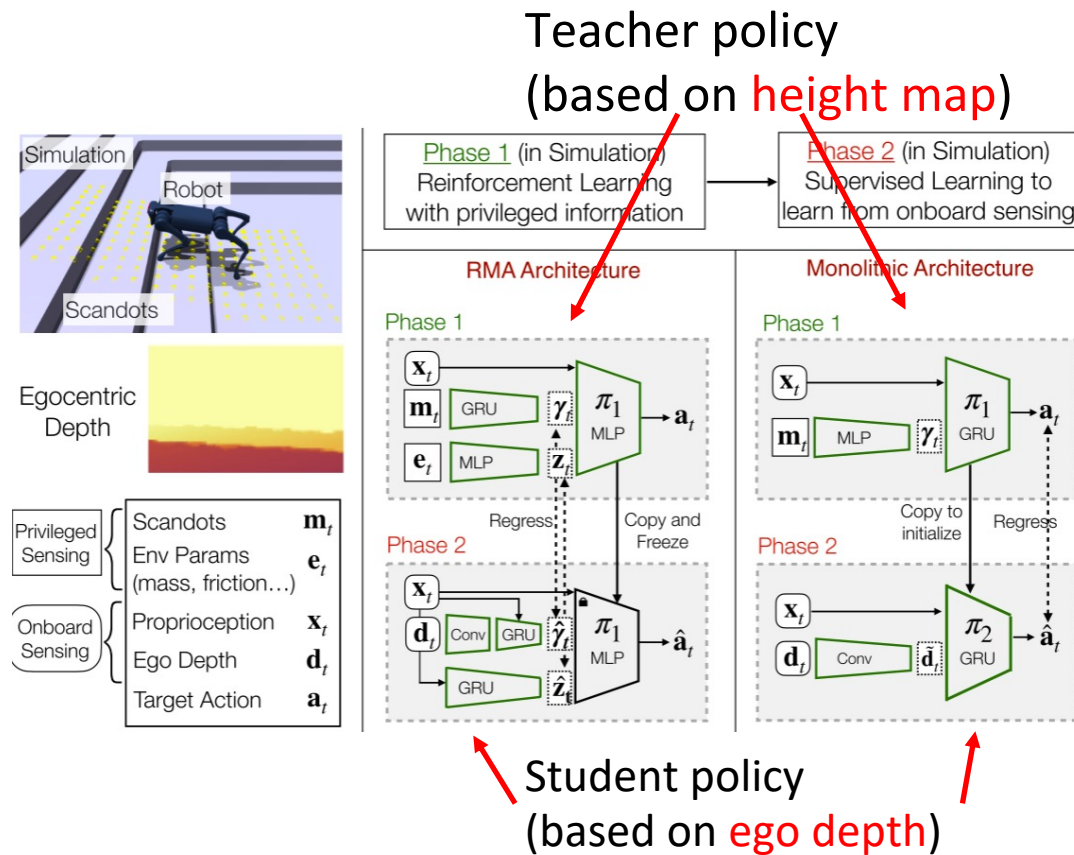


Learning Frameworks – Privileged Learning

Example 1: Policy Distill

Legged Locomotion in Challenging Terrains using Egocentric Vision

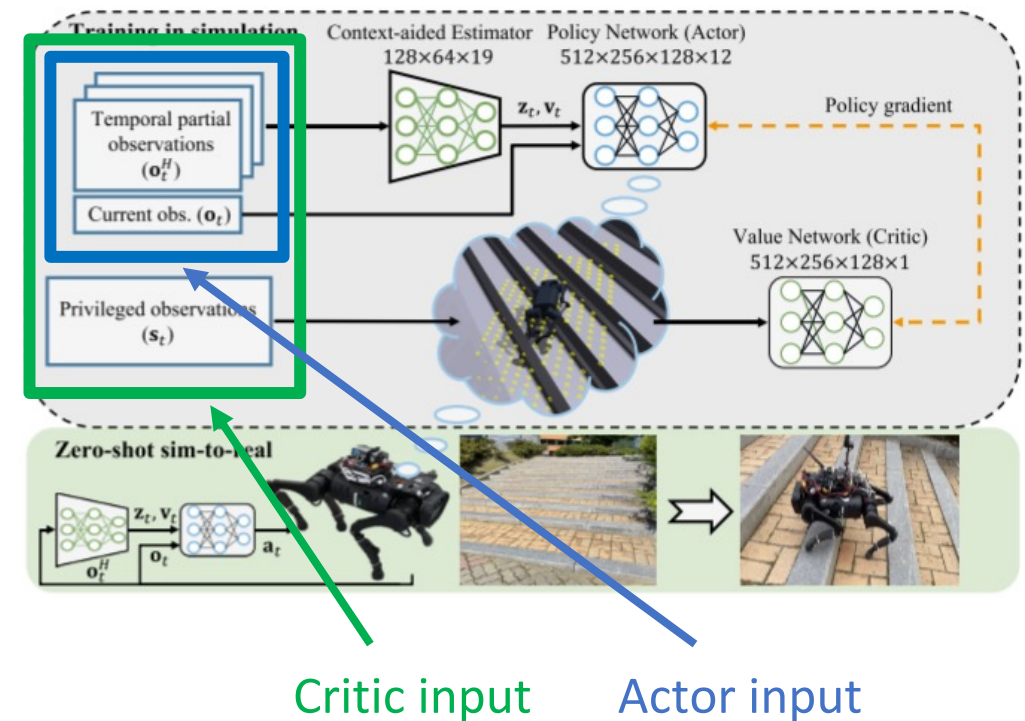
Ananye Agarwal^{*1}, Ashish Kumar^{*2}, Jitendra Malik^{†2}, Deepak Pathak^{†1}
¹Carnegie Mellon University, ²UC Berkeley



Example 2: Asymmetric Actor-Critic

DreamWaQ: Learning Robust Quadrupedal Locomotion With Implicit Terrain Imagination via Deep Reinforcement Learning

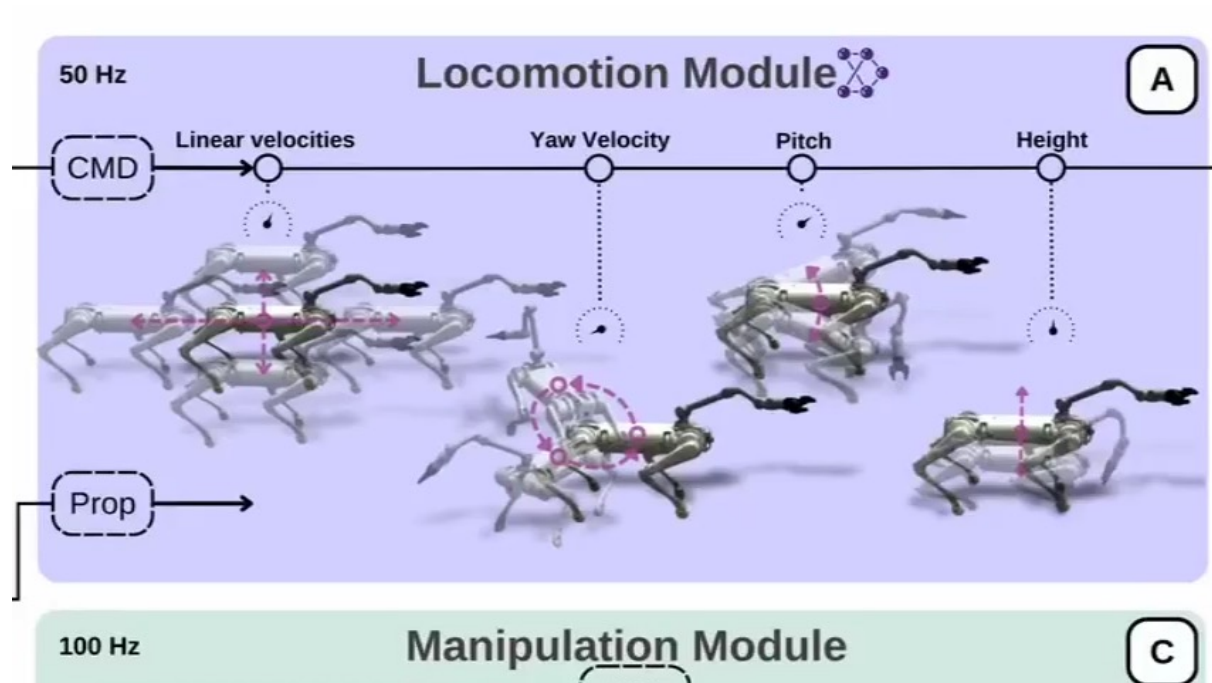
I Made Aswin Nahrendra¹, Byeongho Yu¹, and Hyun Myung^{1*}, *Senior Member, IEEE*



Example of Locomanipulation: QuadWBG



QuadWBG - Locomotion



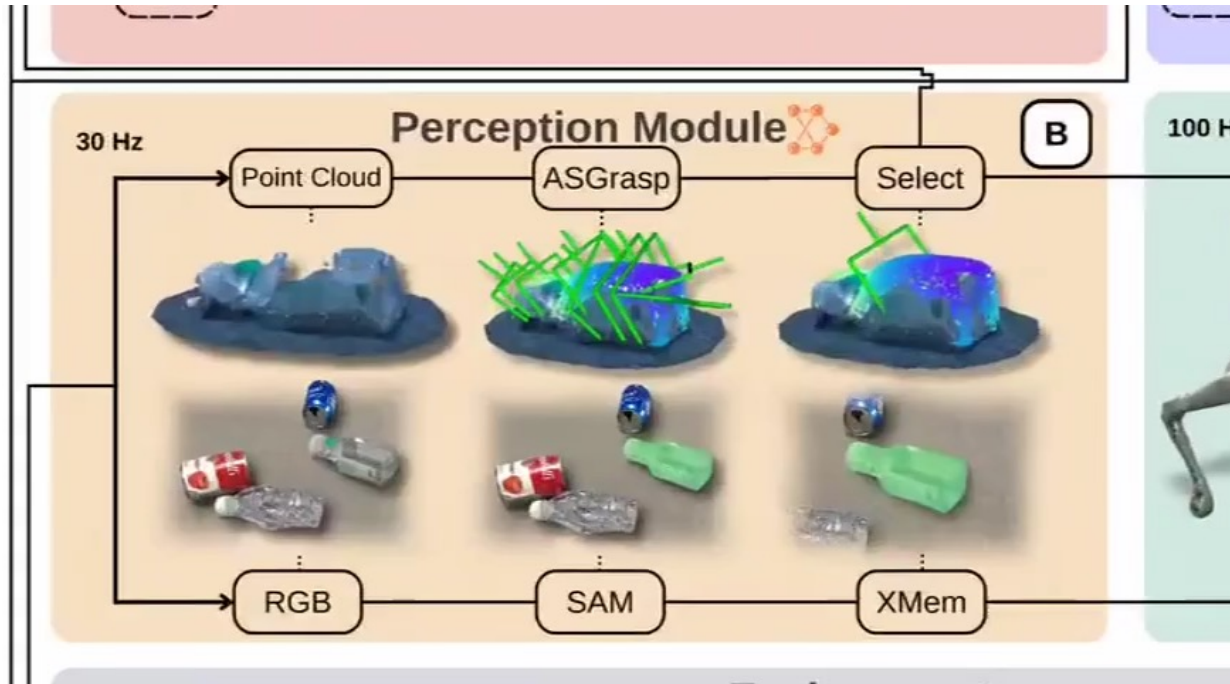
5D Command Locomotion

- linear velocity (x, y)
- yaw velocity
- height position (z)
- pitch position

Whole-body awareness

- randomize ee pose while training

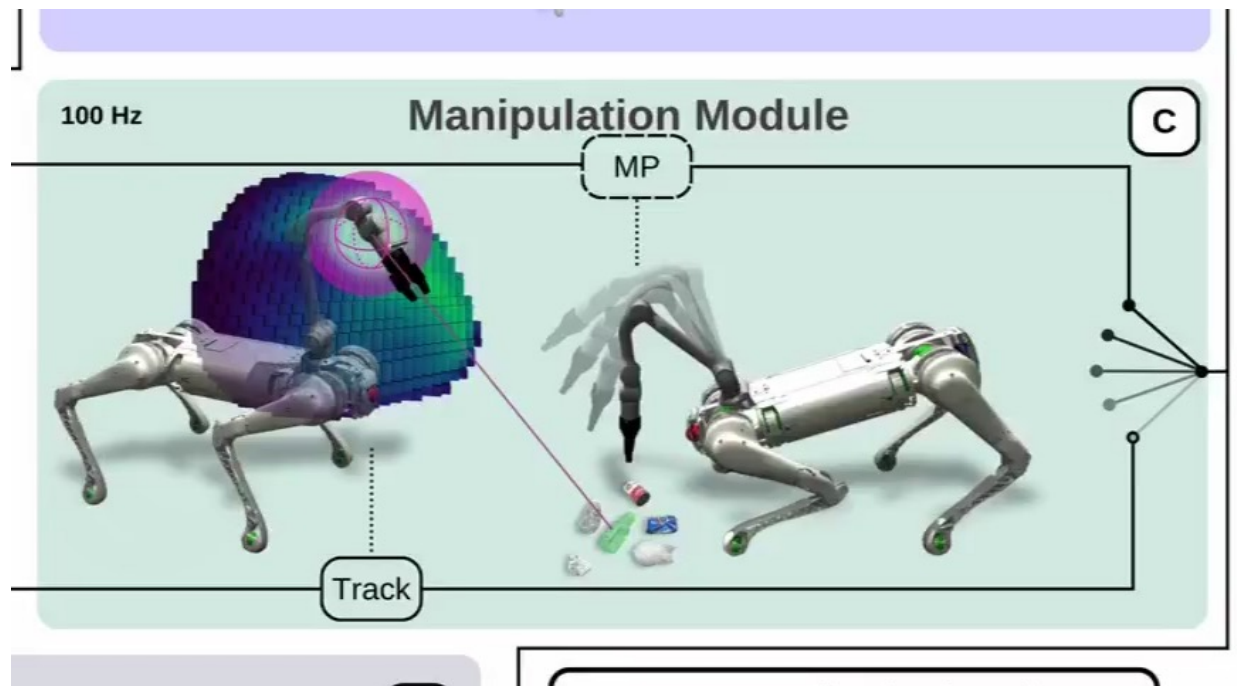
QuadWBG - Perception



Off-the-shelf Module

- Xmen for tracking
- ASGrasp for grasp detection

QuadWBG - Manipulation



Differential IK and online planning

- Continuous and smooth motion
- Adaptive to unexpected movements

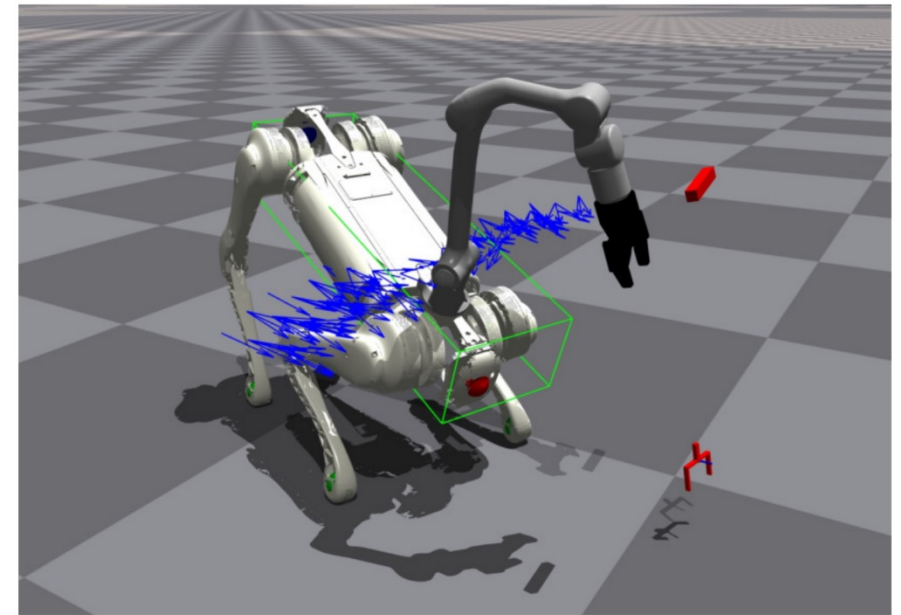
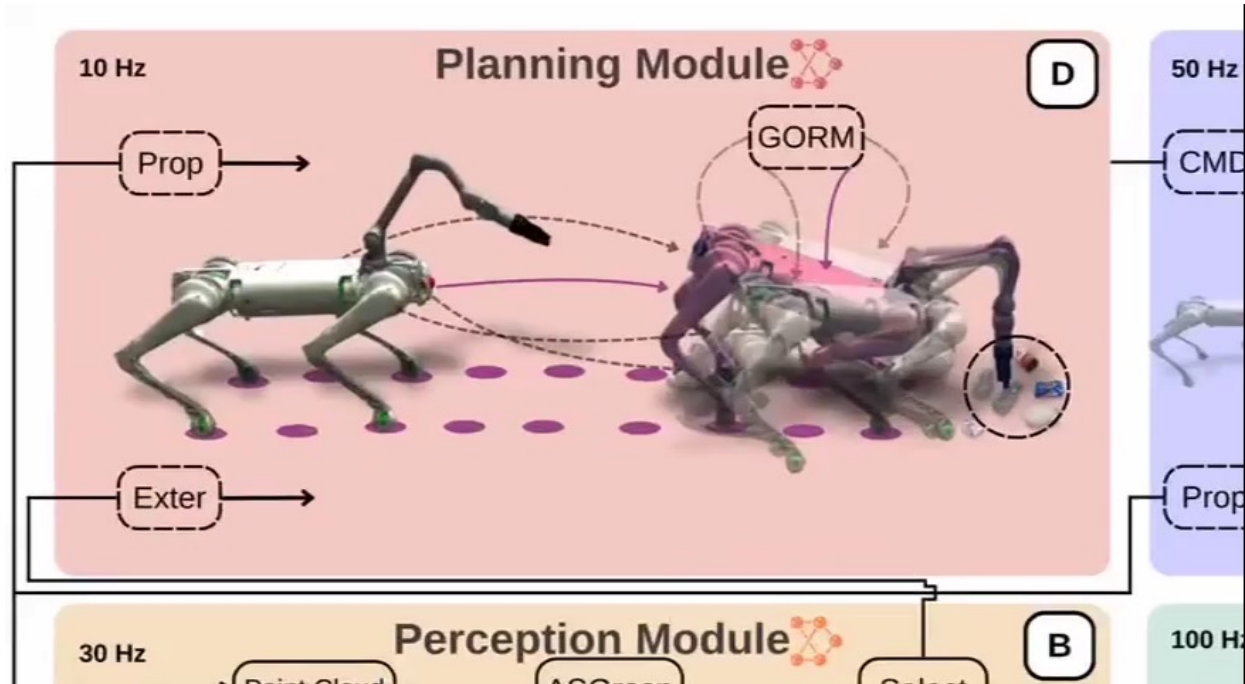
Parameterized tracking sphere

- Minimize motion blur on image
- Reduce loss of tracking

Reachability-based mode switching

- Statistics-based, accurate and stable

QuadWBG – High-level Planner



QuadWBG – High-level Planner

State

Observation	Dimension
goal pose	12
gravity vector	3
joint position	18
last action	5

Action

Action	Dimension
base height	1
base pitch	1
base linear velocity (x, y)	2
base yaw velocity	1

Reward

$$r_{\text{GORM}} = \exp(-(\min(\mathbf{d}_{\text{GORM}}))^2)$$

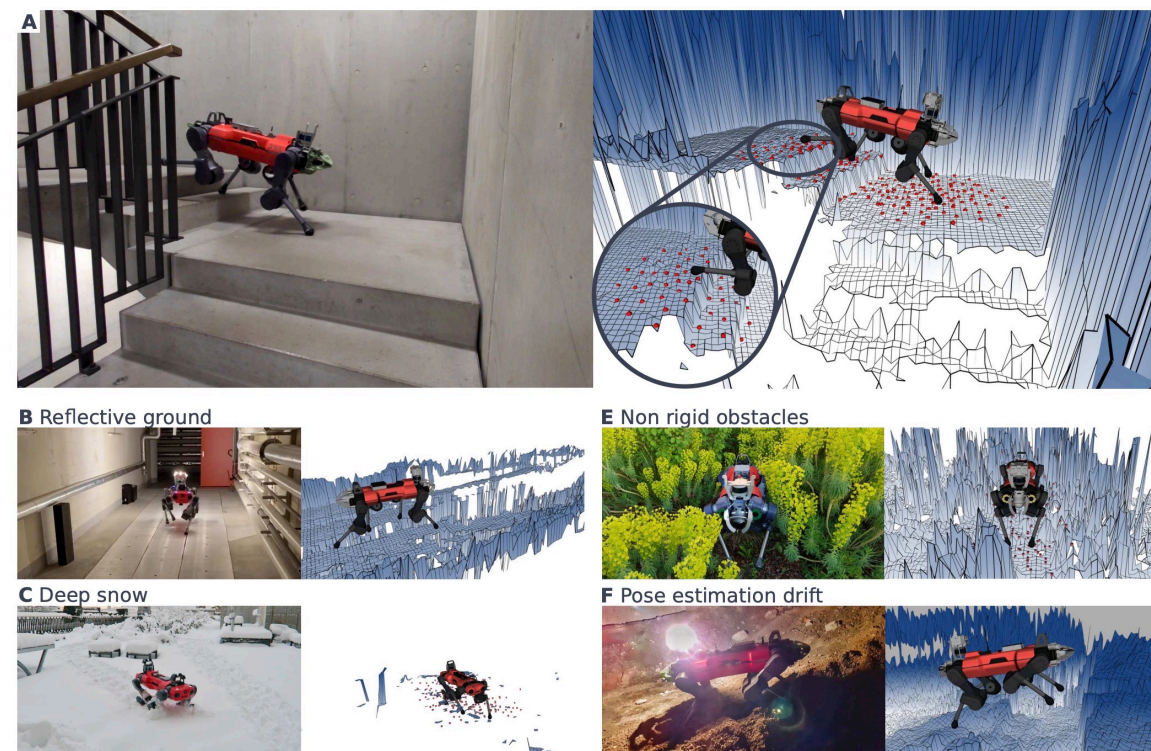
$\mathbf{d}_{\text{GORM}}$: filtered base 6D pose distribution in TCP frame

Other Exemplar Works



Robust to terrain
(only proprioception)

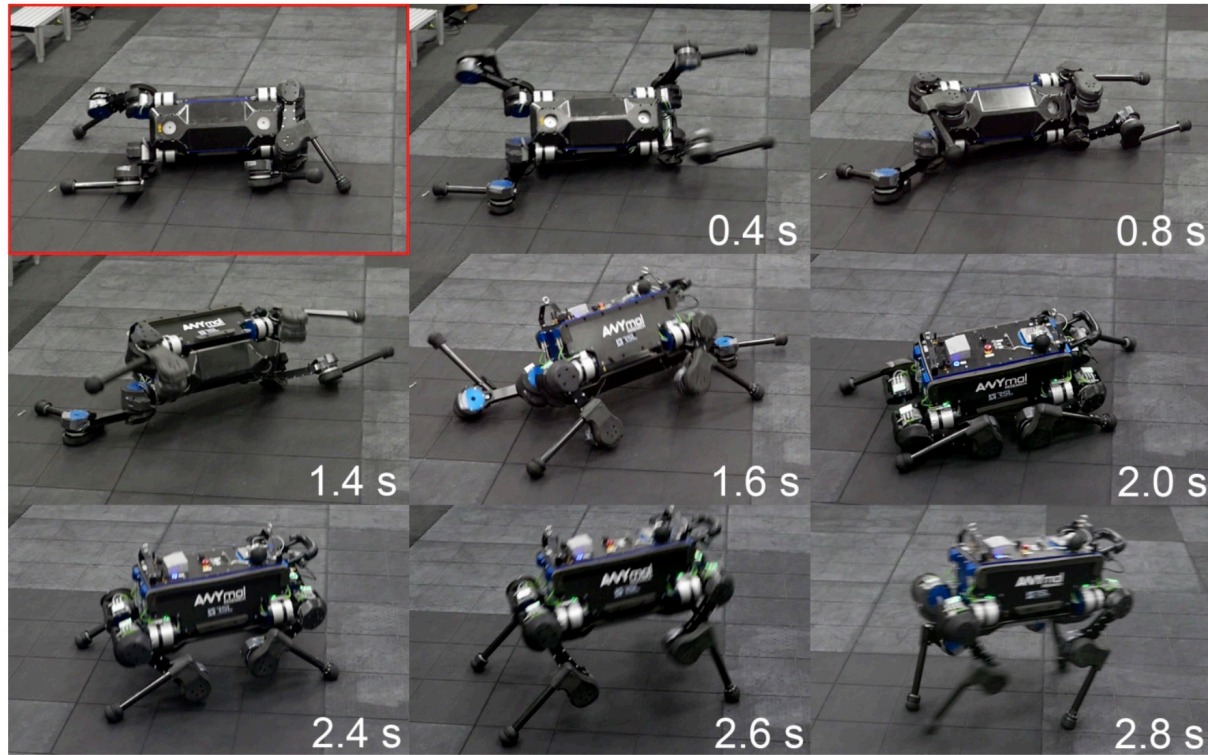
[Lee, Joonho, et al. 2020]



Robust to terrain
(perceive env by vision, radar)

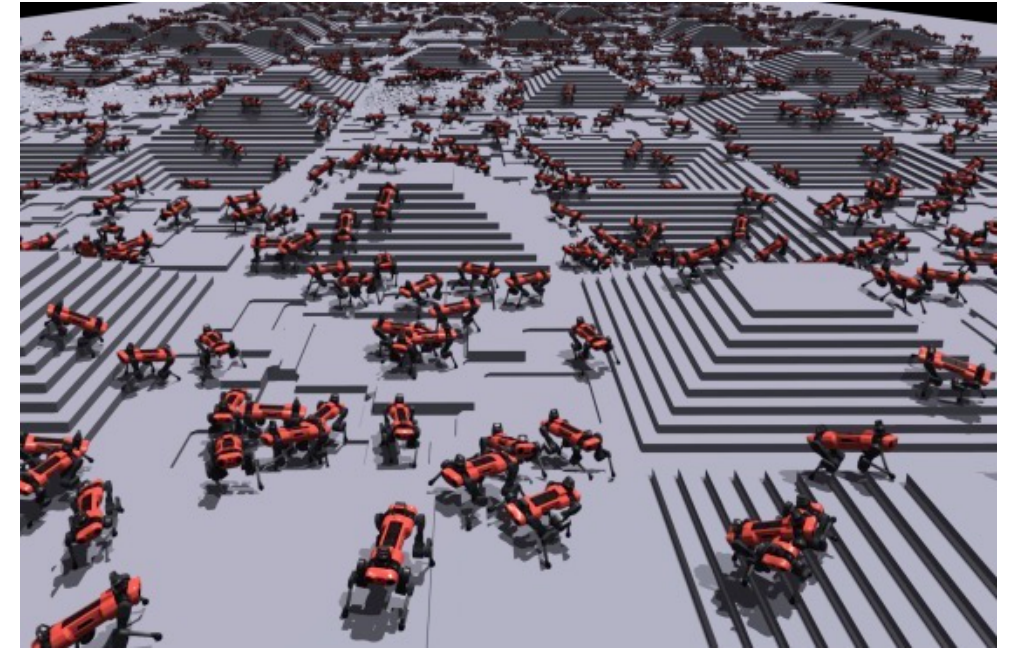
[TAKAHIRO MIKI et al. 2022]

Other Exemplar Works



Robust to a fall

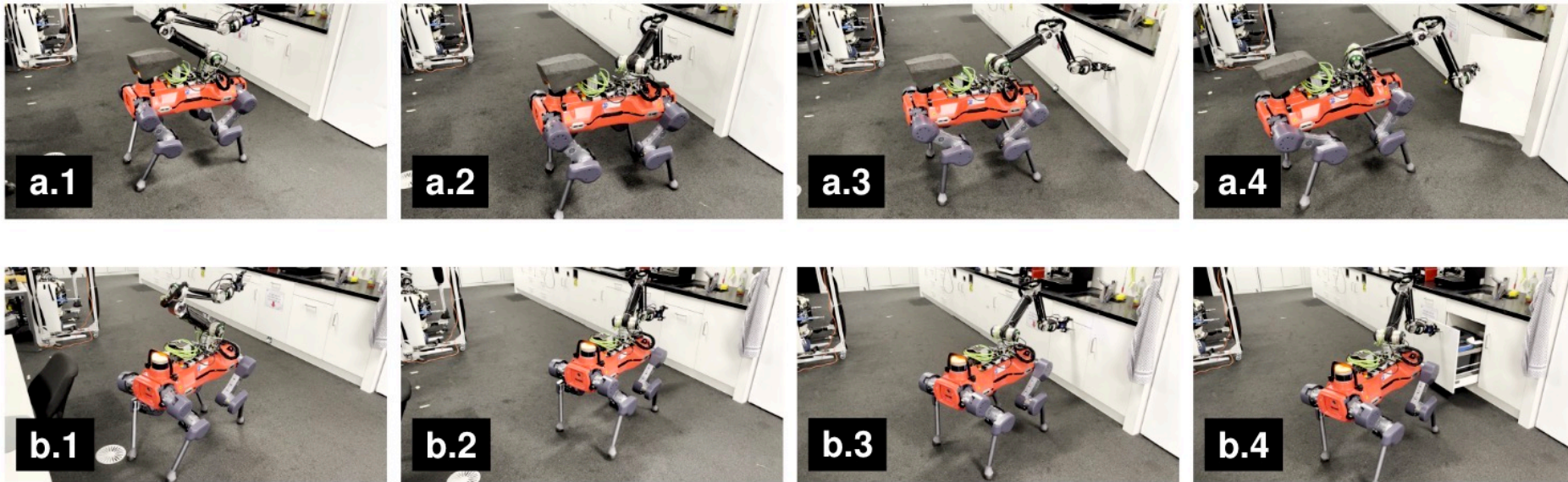
[Lee, Joonho, et al. 2019]



Learn fast

[Nikita Rudin et al. 2020]

Other Exemplar Works

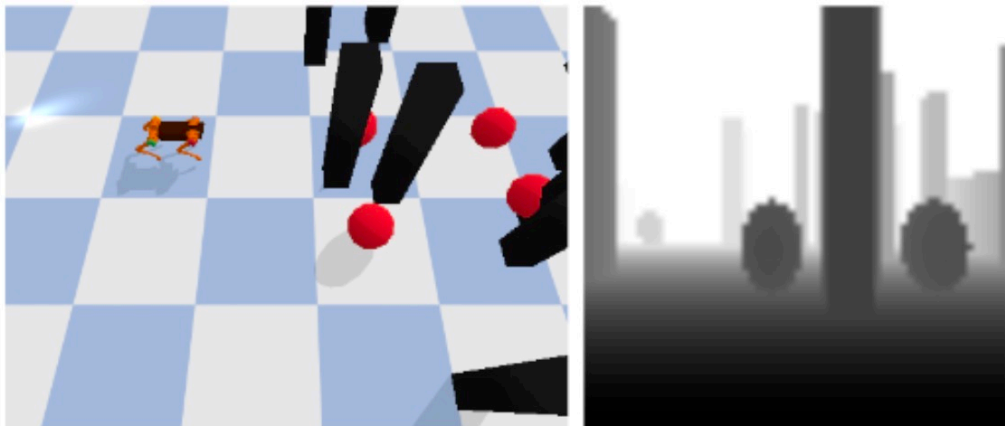


Downstream topic: mobile manipulation [Mayank Mittal et al. 2022]



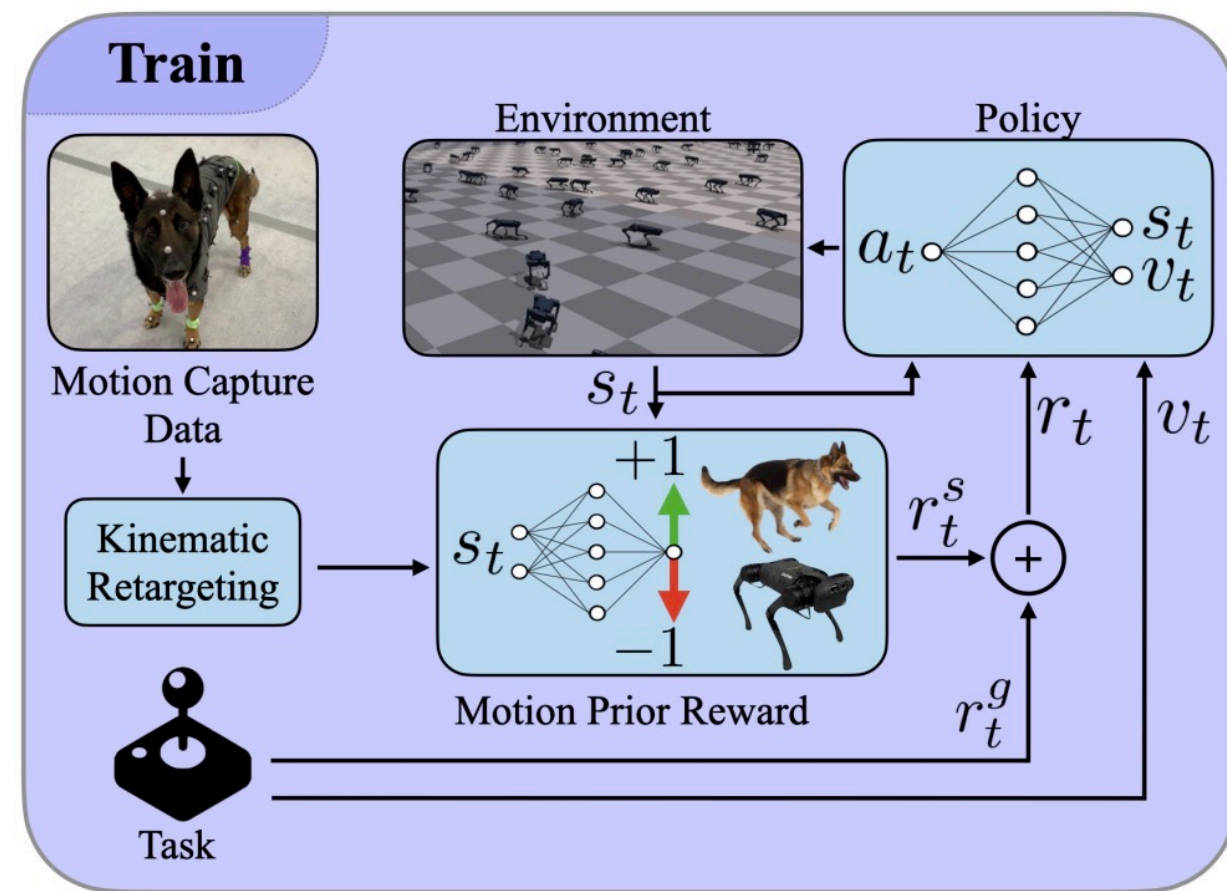
Downstream application: goalkeeper [Huang, Xiaoyu, et al. 2022]

Other Exemplar Works



Avoid obstacles

[Ruihan Yang et al. 2022]



Move naturally

[Alejandro Escontrela et al. 2022]

Challenge and Recent Works

Learning Vision-Guided Quadrupedal Locomotion End-to-End with Cross-Modal Transformers

Ruihan Yang^{*1}, Minghao Zhang^{*2}, Nicklas Hansen¹, Huazhe Xu³, Xiaolong Wang¹,

¹UC San Diego, ²Tsinghua University, ³UC Berkeley

ICLR 2022



Paper



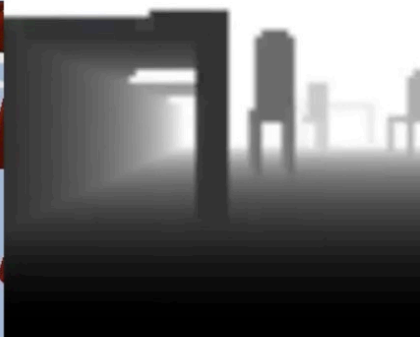
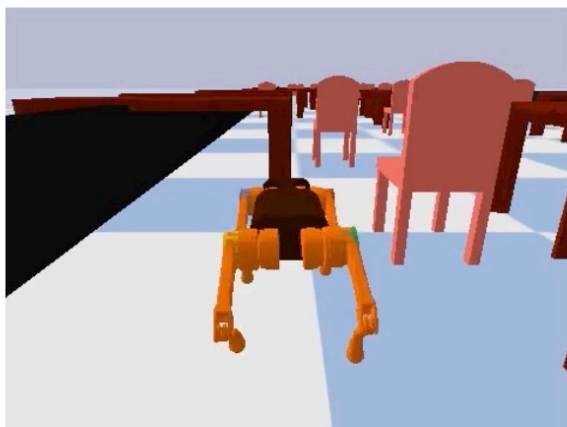
arXiv



Video



Code



Method

- End-to-end RL (PPO)
- Input
 - proprioceptive states
 - Temporal depth images
- Output
 - Action distribution or value

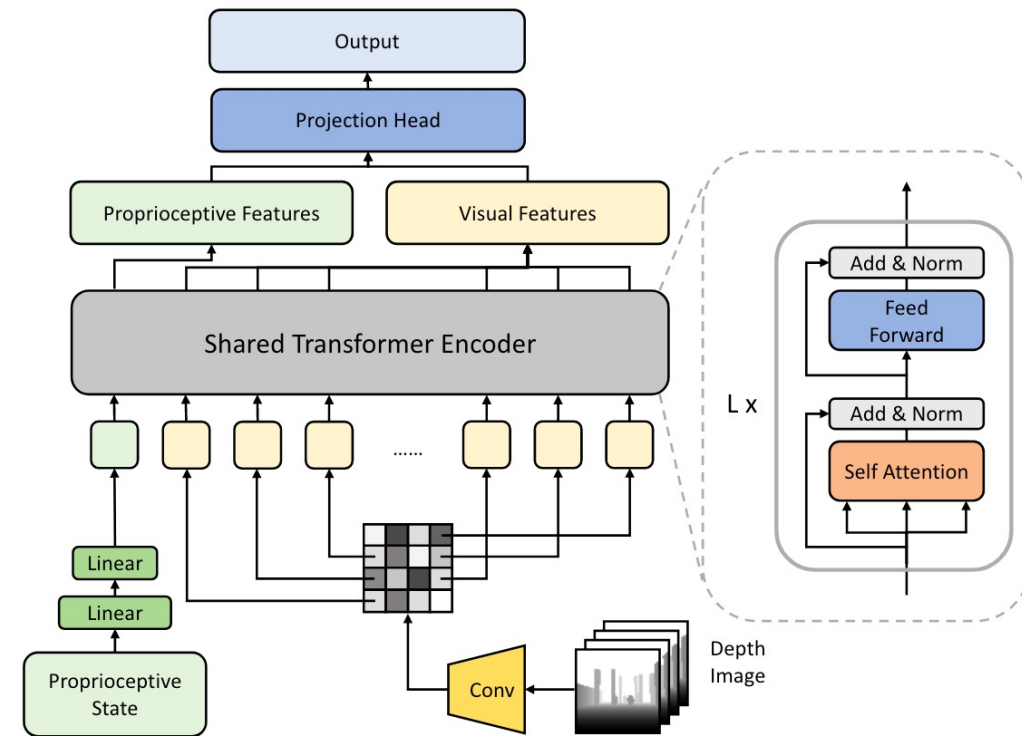
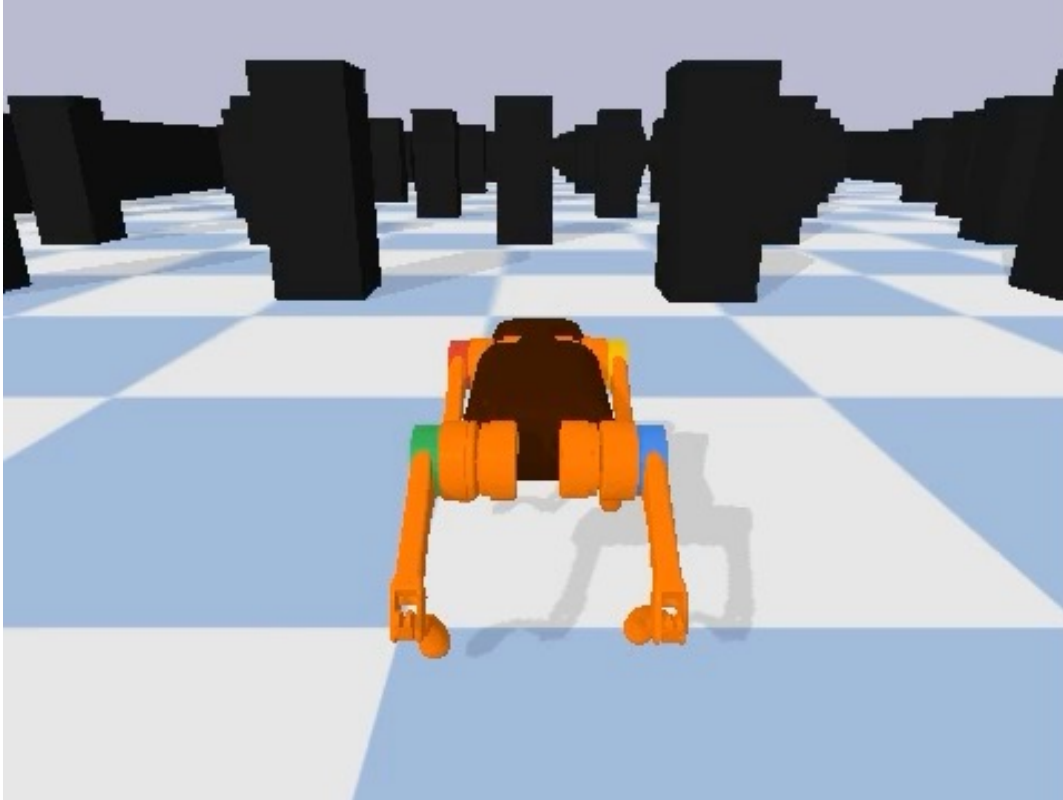
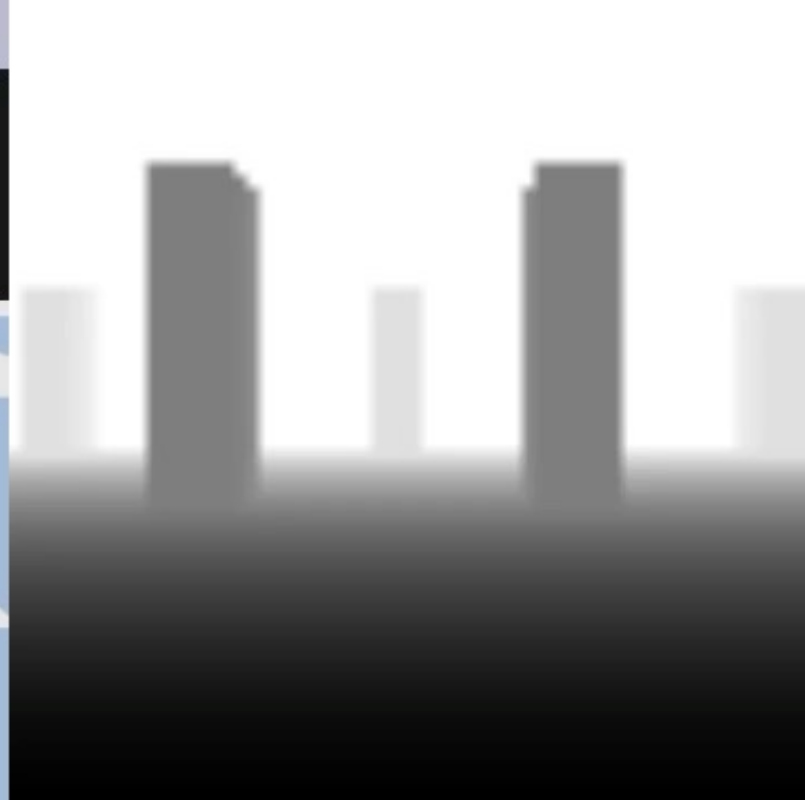


Figure 2: **Network Architecture.** We process proprioceptive states with a MLP and depth images with a ConvNet. We take proprioceptive embedding as a single token, split the spatial visual feature representation into $N \times N$ tokens and feed all tokens into the Transformer encoder. The output tokens are further processed by the projection head to predict value or action distribution.

Results in Simulation



Visualization



Depth observation

Results in Real World



Indoor



Outdoor

From Quadrupedal to Bipedal

- Key differences

Aspect	Quadrupedal	Bipedal
Structure	Low center-of-mass Small DOF	High center-of-mass Large DOF Complex contact modelling of feet
Learning	Traditional MPC / RL methods	Imitate mocap data (reference) Focus more on balancing
Task	Rough terrain traversal Load carrying	Human-like tasks: Loco-manipulation Dance

Bipedal – Use Reference Motion

Methods	Characteristic	Advantages and Disadvantages	Implementation Flow Chart
Residual learning	Add residential term to the known motor positions at the current time step.	A: <u>Fast convergence speed</u> D: <u>Require high-quality predefined reference</u>, limit to specific motions, and <u>lack robustness</u> to complicated terrains.	<pre> graph LR UC[User Commands] --> P[Policy π] Ref[Reference] --> P P --> R[Robots] R --> Sum((⊕)) Sum --> Ref </pre>
Guided learning	Mimic the predefined reference and directly specifies joint-level command.	A: Accelerate the learning process and <u>robust to the terrains</u>. D: <u>Limited to the predefined motions</u> and <u>lack adaptability</u> to unforeseen changes in environment.	<pre> graph LR UC[User Commands] --> P[Policy π] Ref[Reference] -.-> P P --> R[Robots] </pre>
Reference-free learning	Learn locomotion skills from scratch without any prior knowledge.	A: High potential for gait exploration, <u>high robust</u> to the complicated terrain. D: <u>Requires intensive reward shaping</u> for gait patterns and relatively expensive computational training resources.	<pre> graph LR UC[User Commands] --> P[Policy π] P --> R[Robots] </pre>

Bipedal – Use Reference Motion

ExBody²: Advanced Expressive Humanoid Whole-Body Control

Mazeyu Ji^{*1}

Xuanbin Peng^{*1}

Fangchen Liu²

Jialong Li¹

Ge Yang³

Xuxin Cheng^{1,†}

Xiaolong Wang^{1,4,†}

¹UC San Diego

²UC Berkeley

³MIT

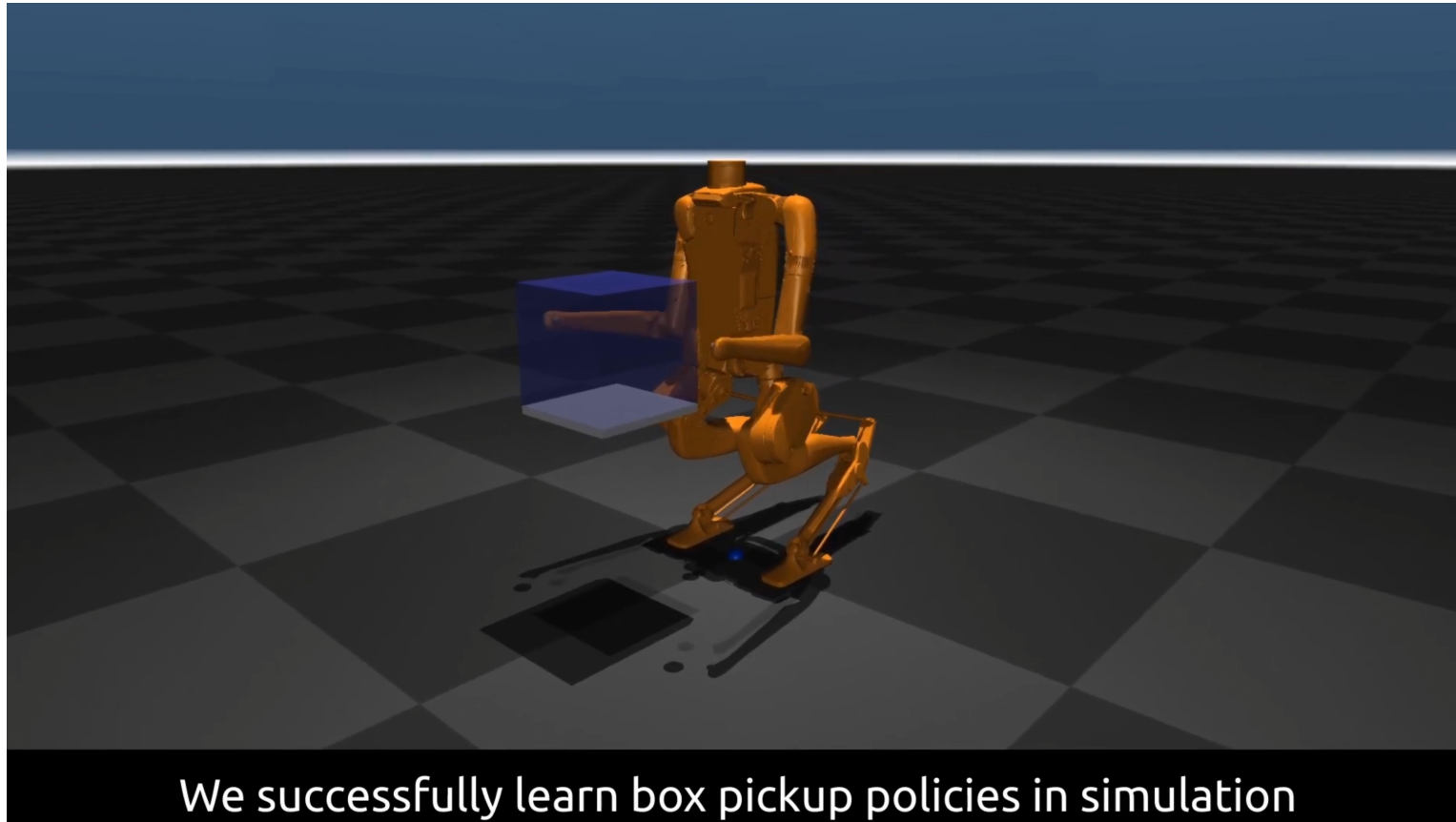
⁴NVIDIA



Bipedal – Loco-Manipulation

Sim-to-Real Learning for Humanoid Box Loco-Manipulation

Jeremy Dao, Helei Duan, Alan Fern



We successfully learn box pickup policies in simulation

Bipedal – Contact Handling

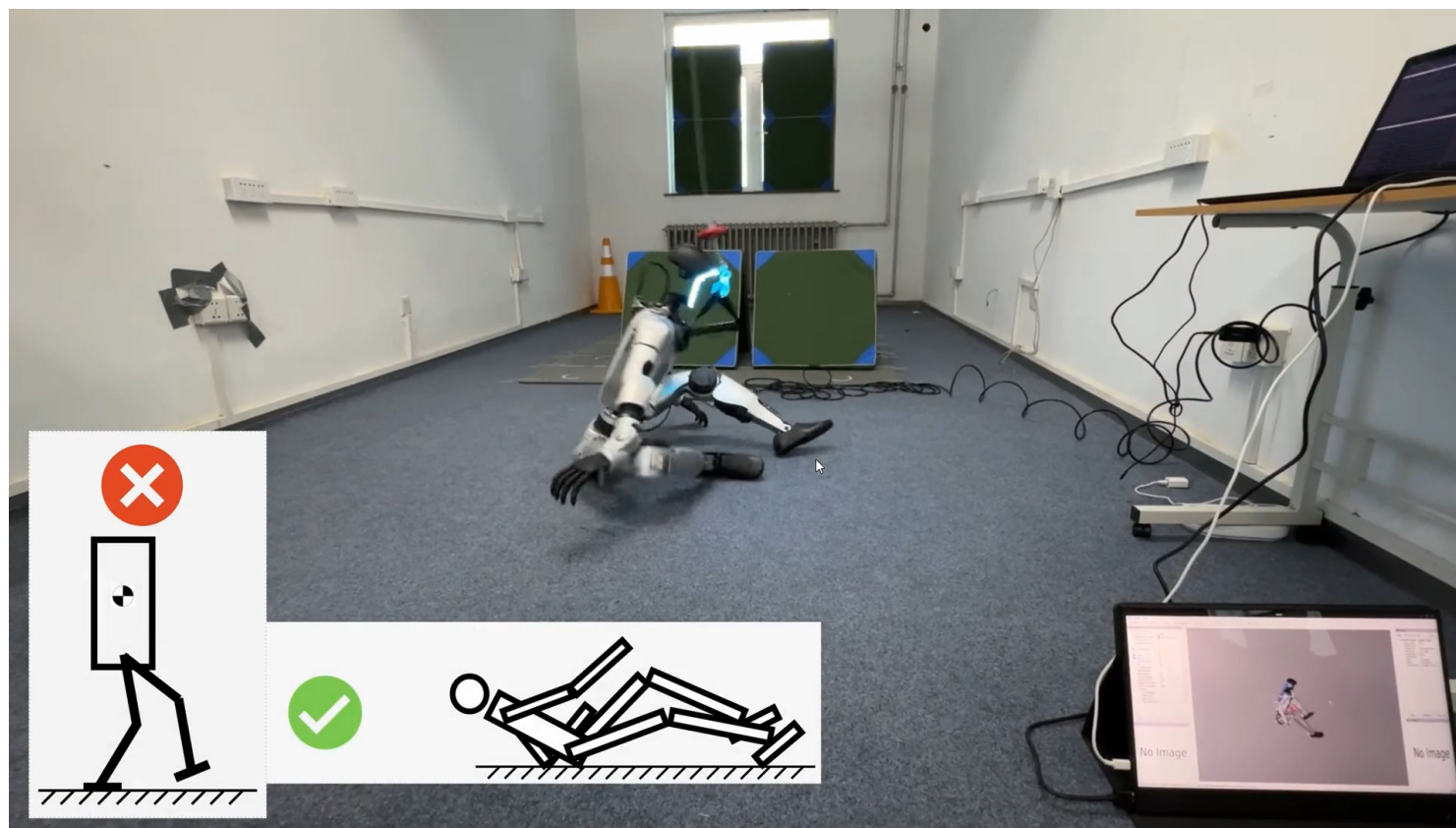
Embrace Collisions: Humanoid Shadowing for Deployable Contact-Agnostic Motions

Ziwen Zhuang

IIS, Tsinghua University
Shanghai Qi Zhi Institute

Hang Zhao

IIS, Tsinghua University
Shanghai Qi Zhi Institute



Sim-to-real

- Ways to mitigate sim-to-real gap
 - Reward design
 - Action smoothness, action rate
 - Domain Randomization
 - Dynamics: center of mass, mass, friction
 - Other: latency, image augmentation
 - Domain Adaptation
 - System Identification
 - Model actuator dynamics with real data
 - Better contact model in simulation

Difficulty

- Open-source is a problem
 - Many companies don't open algorithm
 - Hard to reimplement even if open-sourced algorithm
- Both MPC and RL needs careful tuning
 - Each gait needs to tune a model (especially for MPC)
- Robustness issue and sim-to-real gap

Future Directions

- Combining control and learning
 - Using trajectory planner to guide the low-level RL-based controller
 - Using RL to calculate reference key points (e.g. foothold location) for MPC
 - ...
- Better Training Strategy
 - Make training robust to the coefficients of reward terms
- Safety
 - Safe RL
- Mobile-manipulation
 - Make upper-body better coordinate with lower-body

Additional Resources

- Codebase
 - Unitree RL GYM: https://github.com/unitreerobotics/unitree_rl_gym
 - Legged GYM: https://github.com/leggedrobotics/legged_gym
 - MuJoCo Playground: https://github.com/google-deepmind/mujoco_playground
- Papers
 - Quadrupedal Locomotion
 - Learning-based legged locomotion: state of the art and future perspectives
 - Learning to Walk in Minutes Using Massively Parallel Deep Reinforcement Learning
 - DreamWaQ: Learning Robust Quadrupedal Locomotion With Implicit Terrain Imagination via Deep Reinforcement Learning
 - Legged Locomotion in Challenging Terrains using Egocentric Vision
 - ANYmal Parkour: Learning Agile Navigation for Quadrupedal Robots
 - Walk These Ways: Tuning Robot Control for Generalization with Multiplicity of Behavior

Additional Resources

- Papers

- Quadrupedal Locomotion

- Learning quadrupedal locomotion on deformable terrain
 - Data Efficient Reinforcement Learning for Legged Robots

- Quadrupedal Loco-Manipulation

- Visual Whole-Body Control for Legged Loco-Manipulation

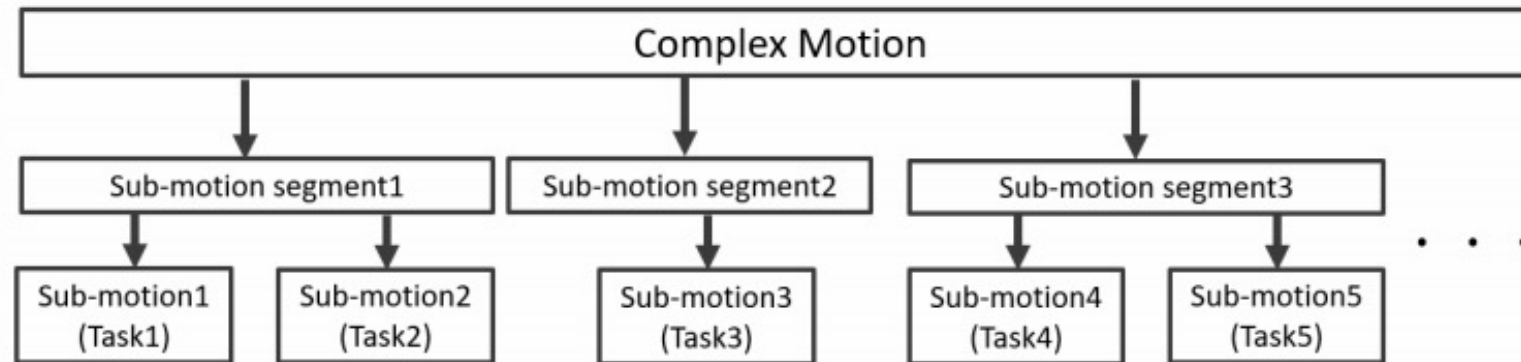
- Bipedal Locomotion

- Deep Reinforcement Learning for Bipedal Locomotion: A Brief Survey
 - ExBody2: Advanced Expressive Humanoid Whole-Body Control
 - Reinforcement Learning for Versatile, Dynamic, and Robust Bipedal Locomotion Control
 - Toward Understanding Key Estimation in Learning Robust Humanoid Locomotion
 - Berkeley Humanoid: A Research Platform for Learning-based Control

Gripper Manipulation

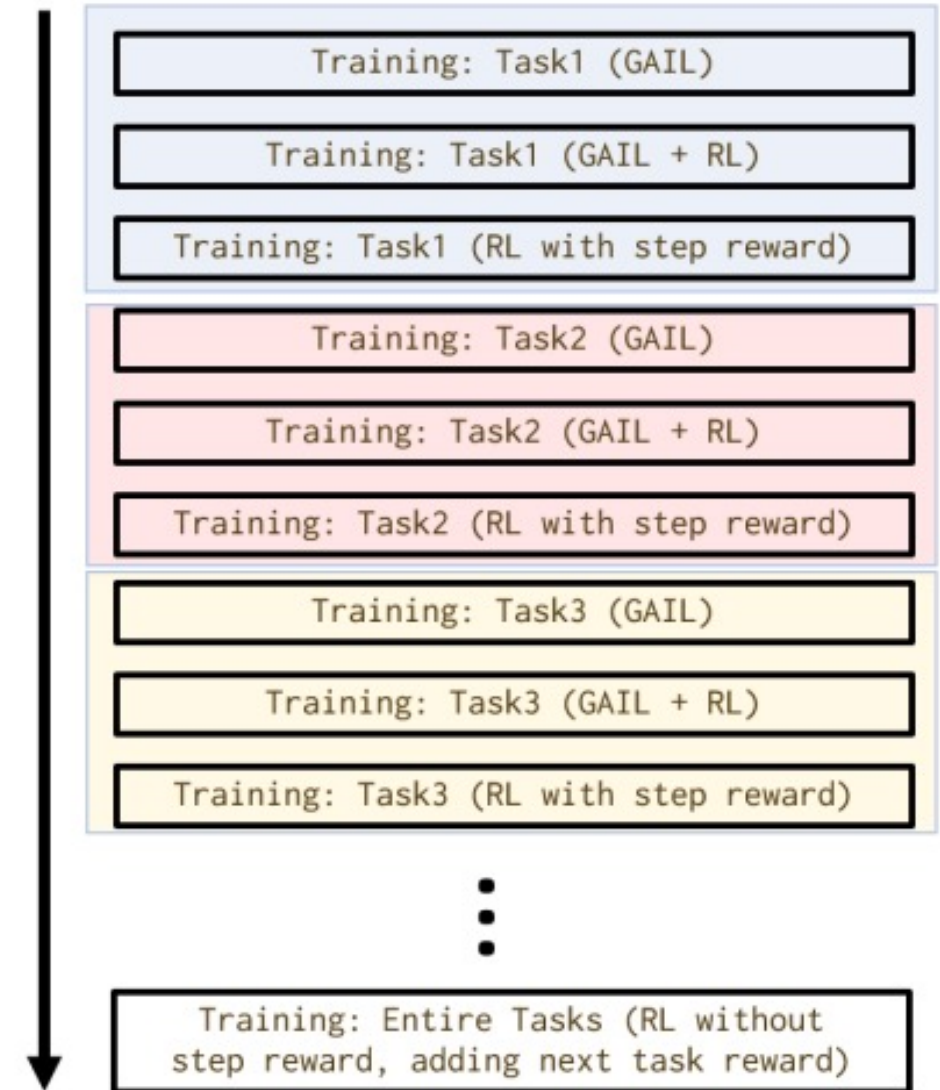
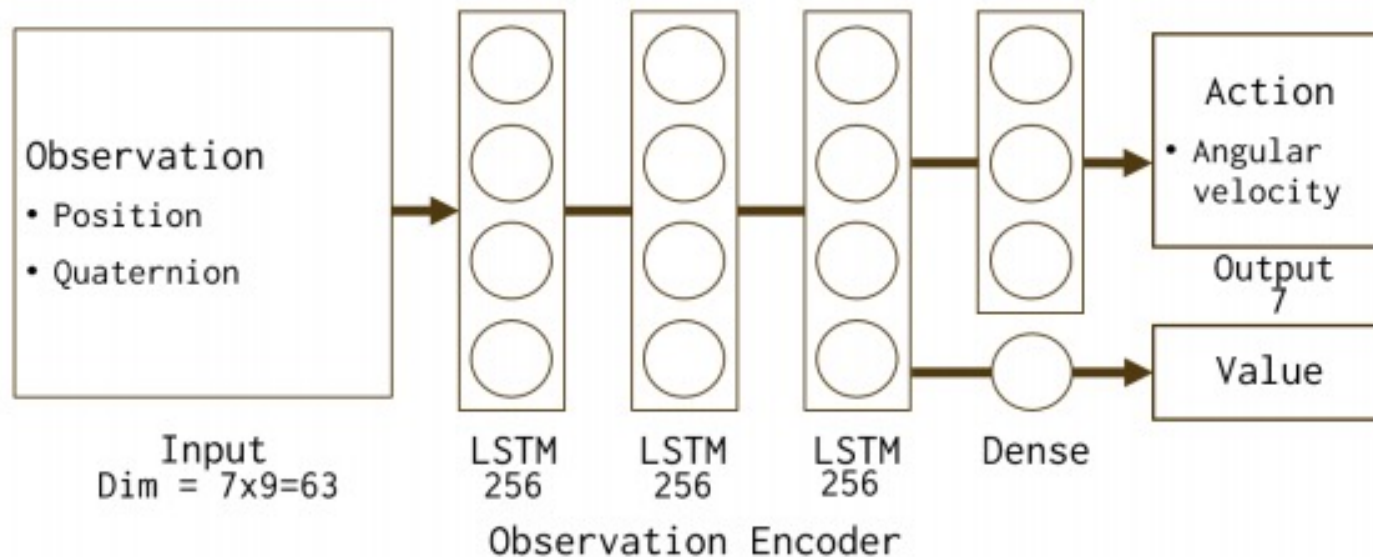
Learning 6DoF Grasping Using Reward-Consistent Demonstration

- 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:



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

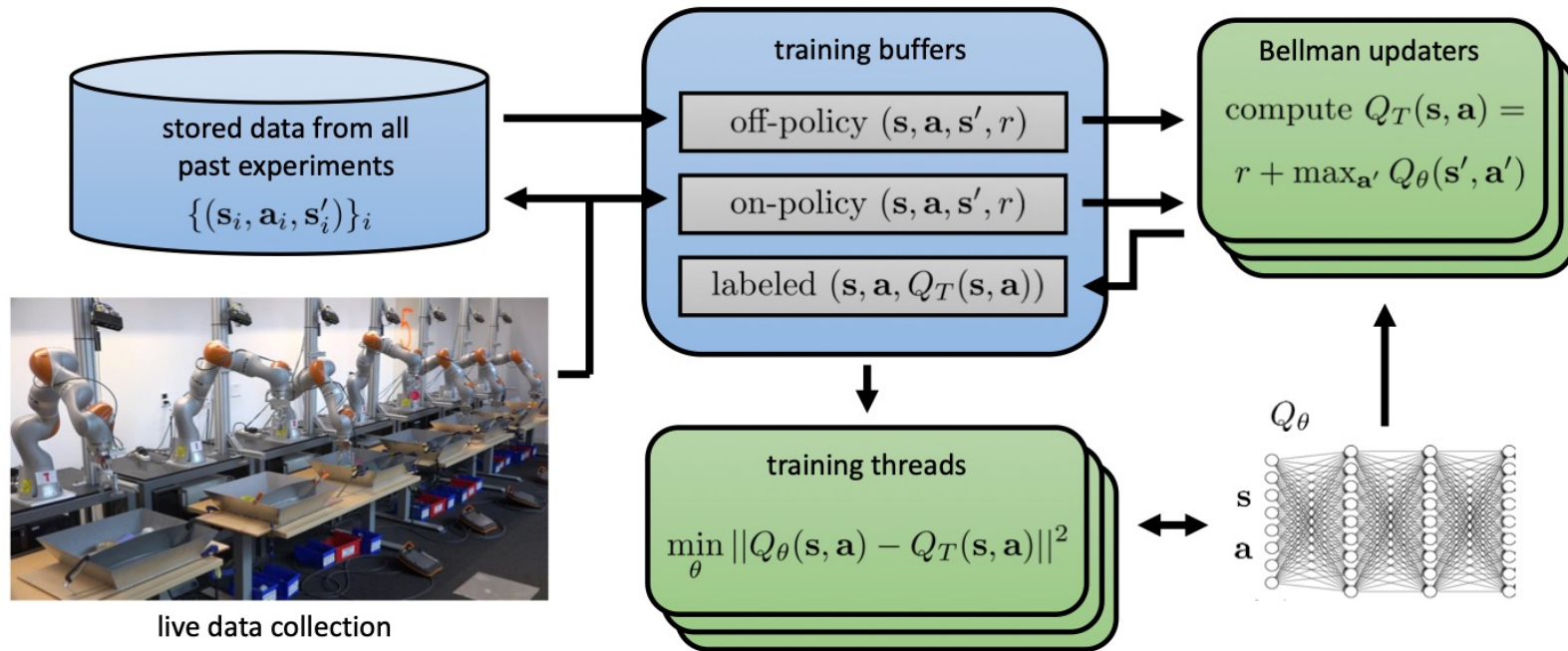


Seven robots collecting grasp data

QT-Opt

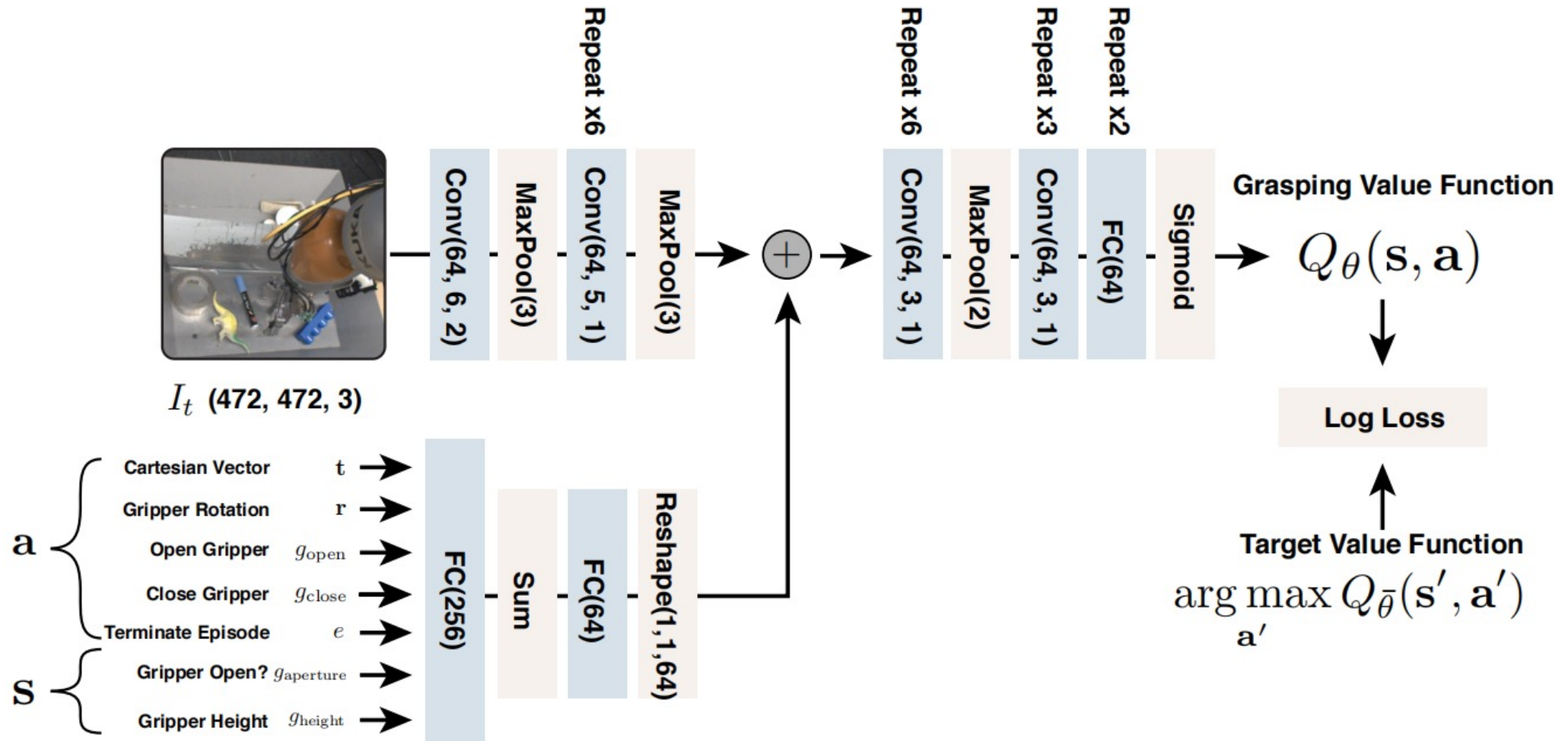
Distributed RL infrastructure for QT-Opt

- 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

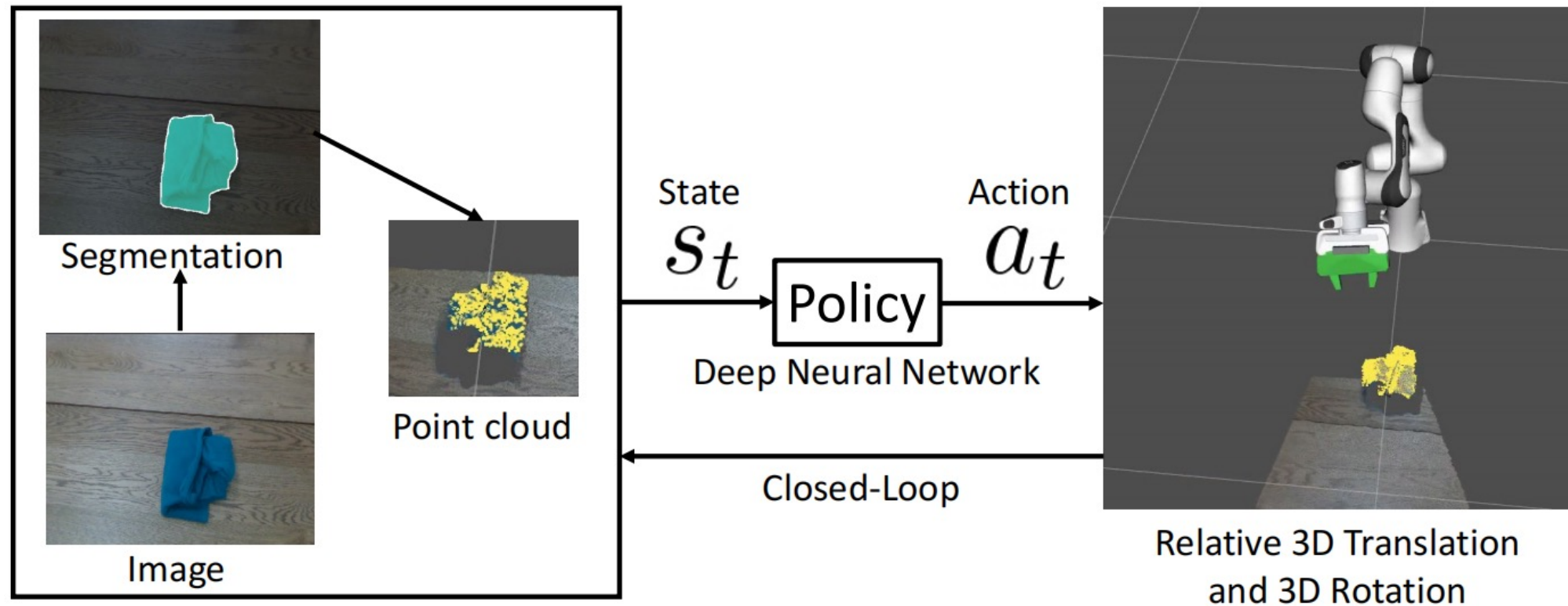
The architecture of grasping Q-Function



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

Learning a closed-loop policy for 6D robotic grasping

- Off-policy



GA-DDPG

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}

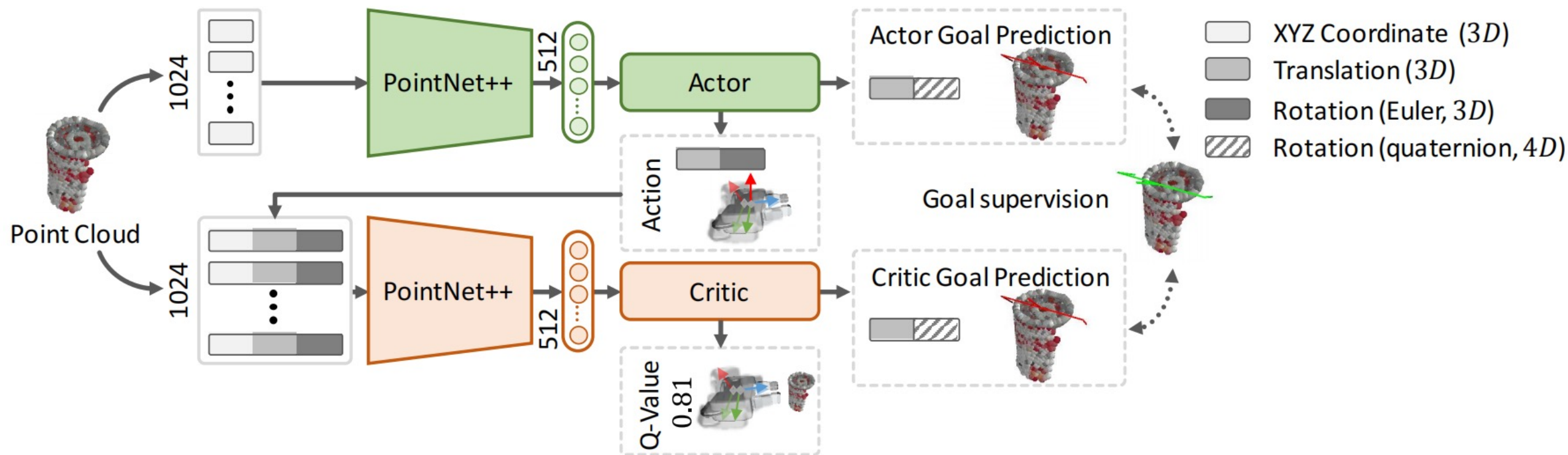
GA-DDPG

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



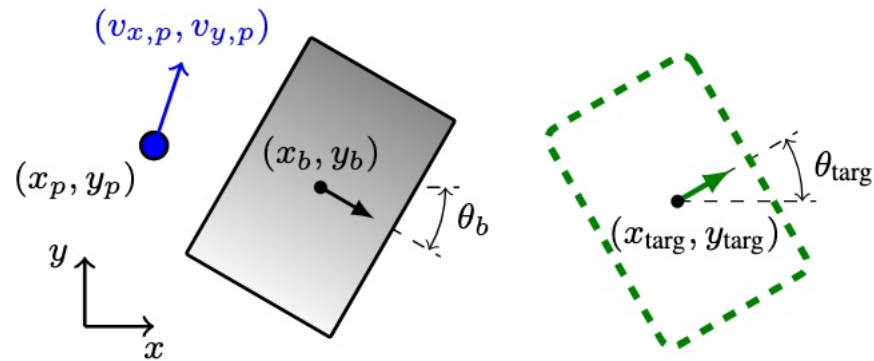
$$L_{\theta} = \lambda L_{BC}(a^*, a_{\theta}) + (1 - \lambda) L_{DDPG}(s, a_{\theta}) + L_{AUX}(g, g_{\theta})$$

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

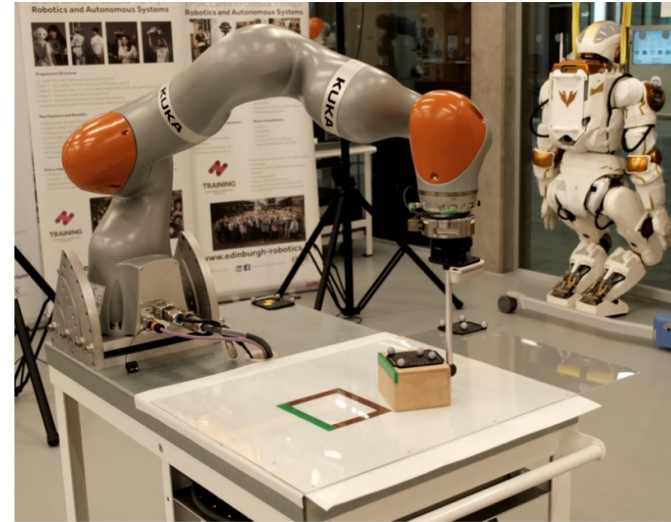
Non-prehensile Manipulation

Planar Pushing

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

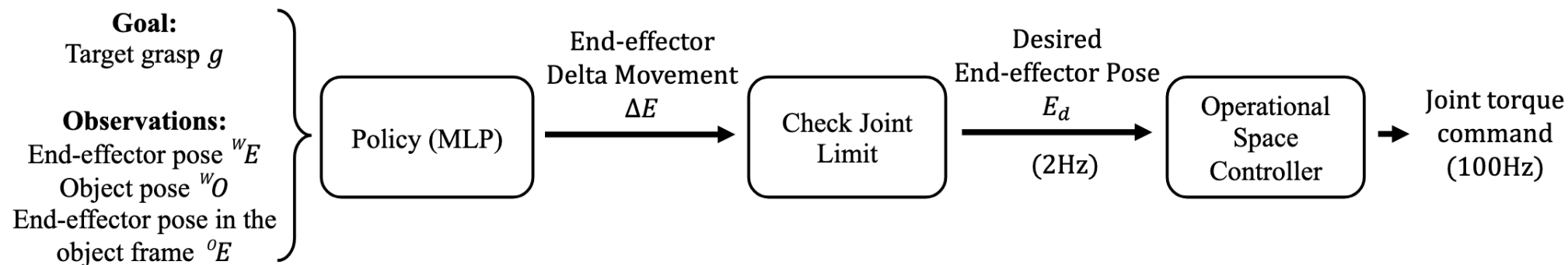


Observation and action



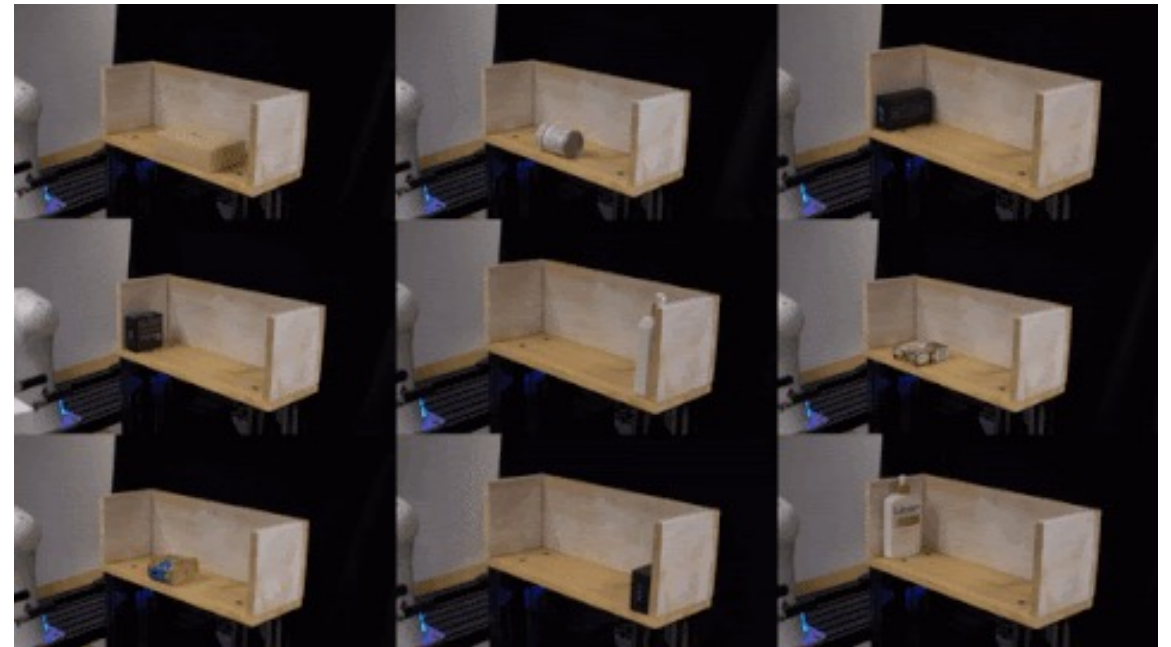
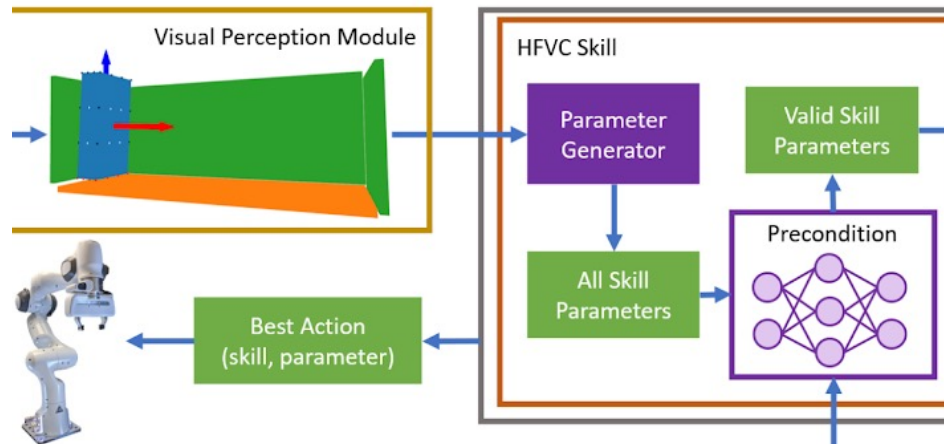
Extrinsic Dexterity

- 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



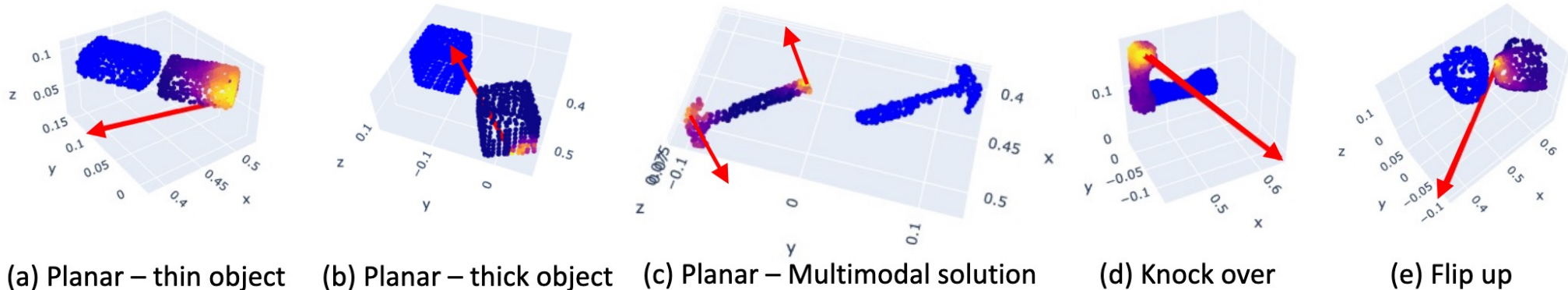
Generalizable 6D Non-prehensile Manipulation

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



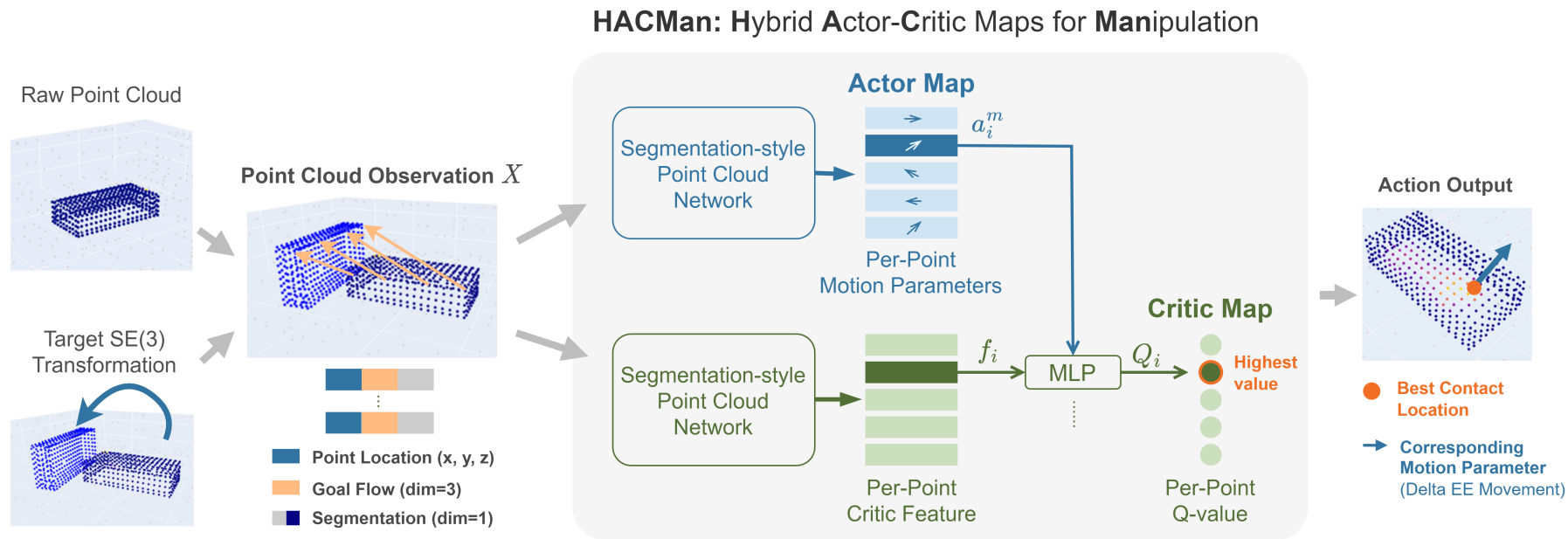
HACMan

- 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

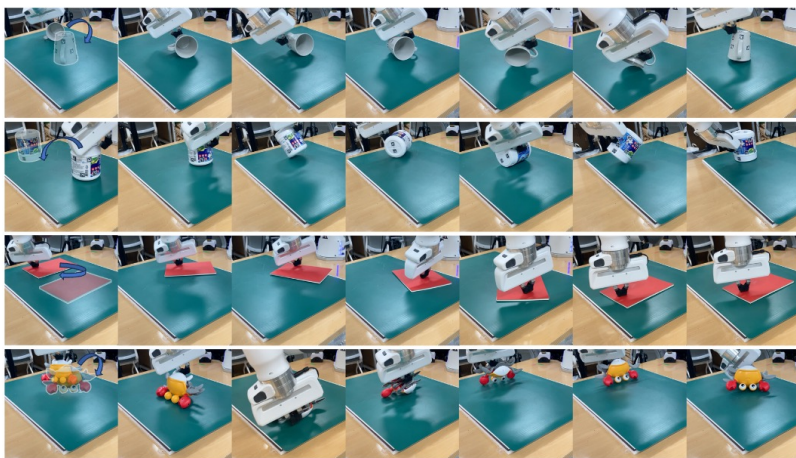
Closed-loop Non-prehensile Manipulation Policy

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



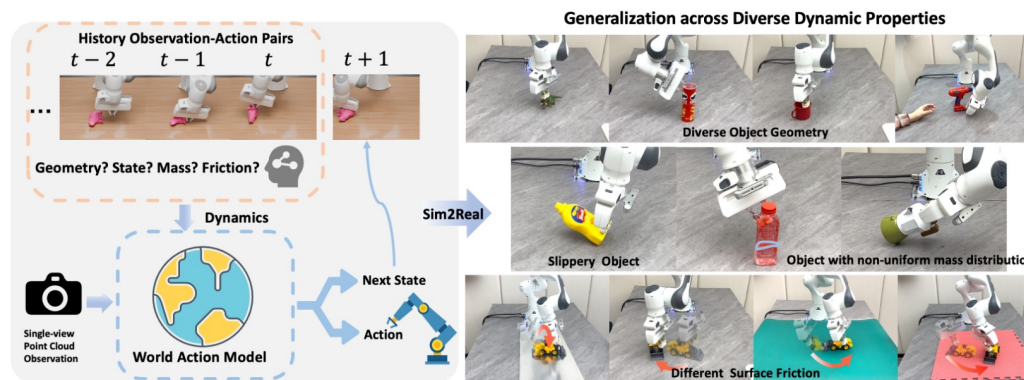
ICLR 2024

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

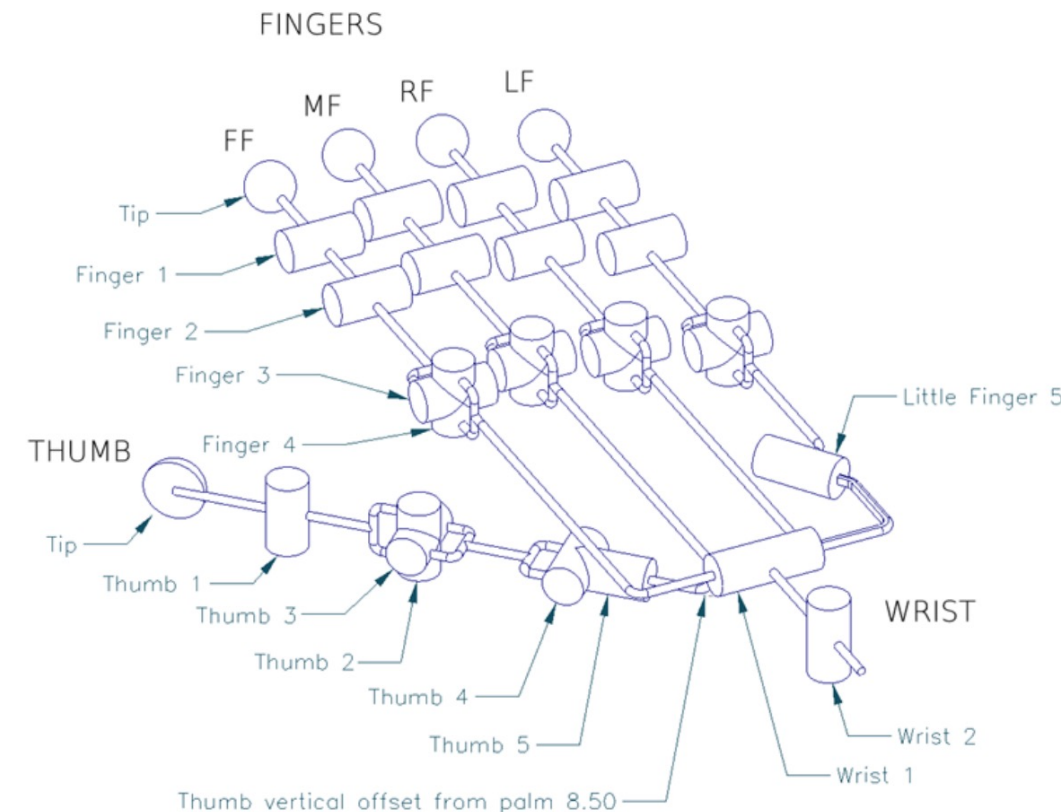
2

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)

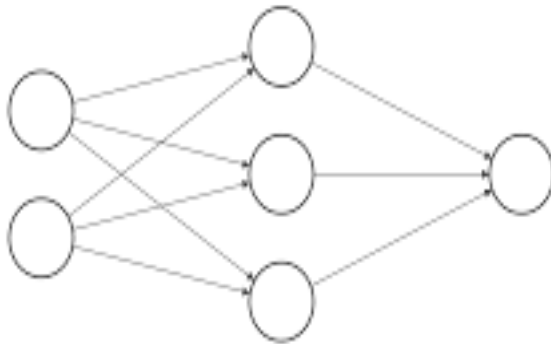
RL for continuous Control



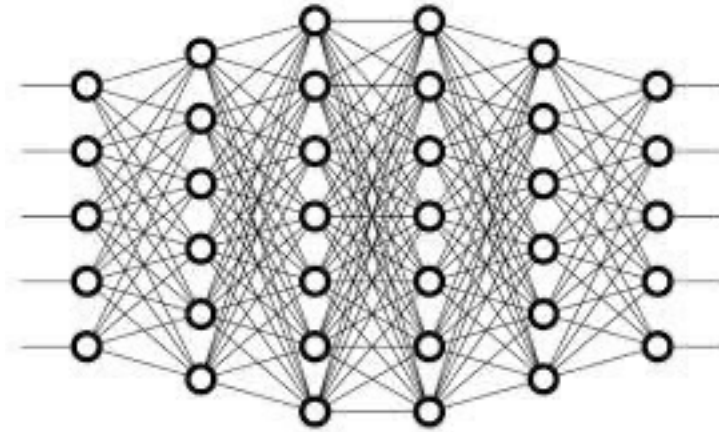
✗ Prohibitively large number of samples and safety concerns

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

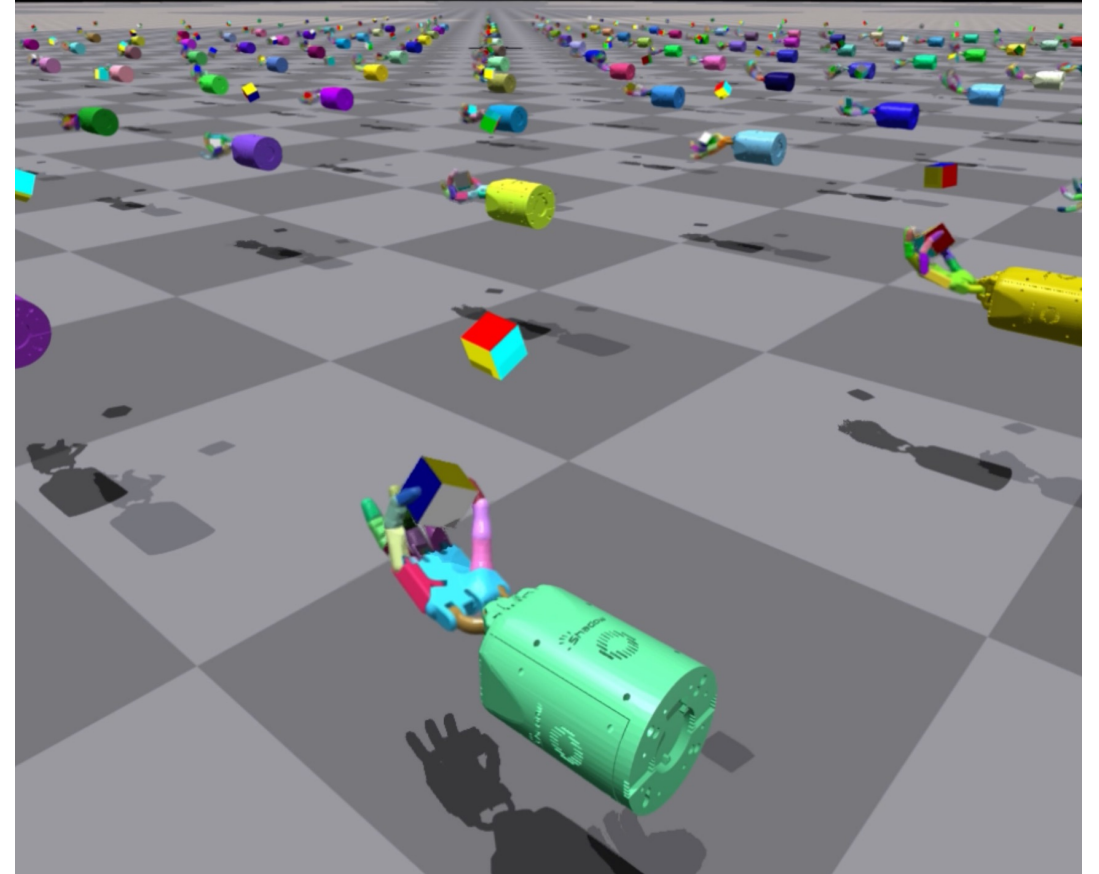
Challenge 3: Sim-to-Real Gap

- 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

Dexterous Manipulation

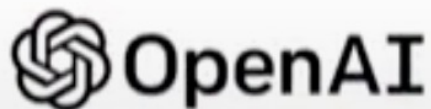
- Challenge 1: sample inefficient
 - RL + Imitation learning
 - ...
- Challenge 2: visual observation
 - Teacher-student distillation
 - ...
- Challenge 3: sim-to-real gap
 - Domain randomization
 - Model-based RL
 - ...

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

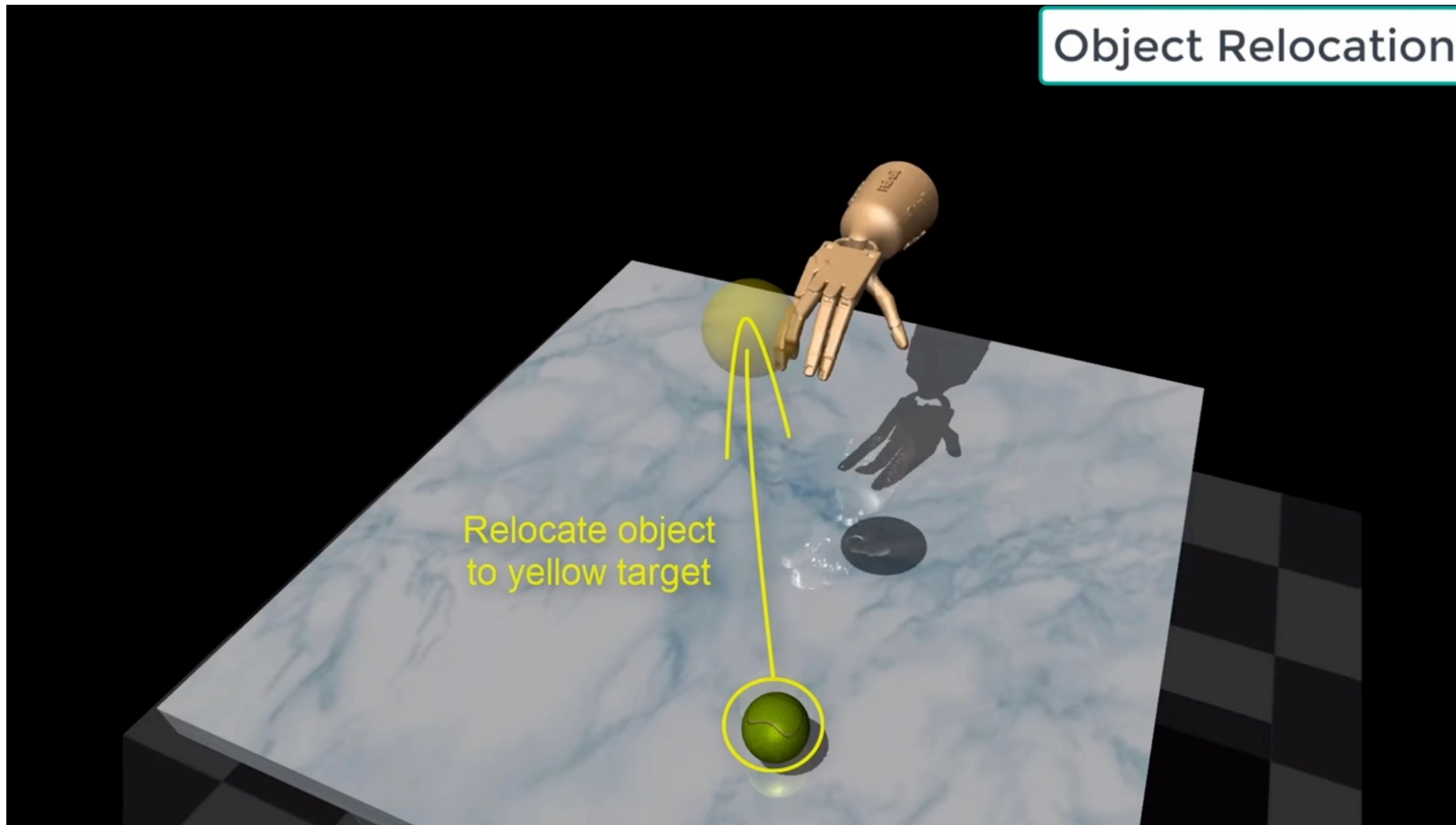
(* equal contributions)



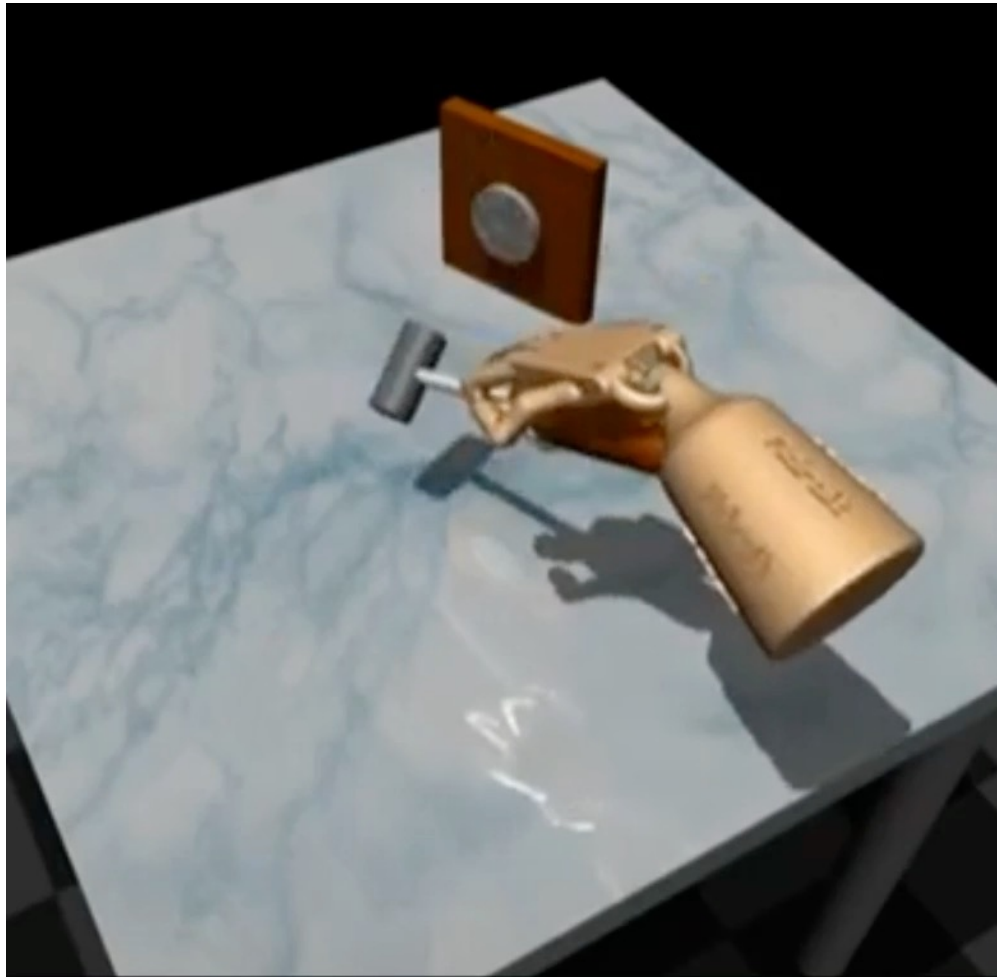
PAUL G. ALLEN SCHOOL
OF COMPUTER SCIENCE & ENGINEERING



Results in simulation



Only RL or Only IL



Model-free policy gradient



Behavior cloning

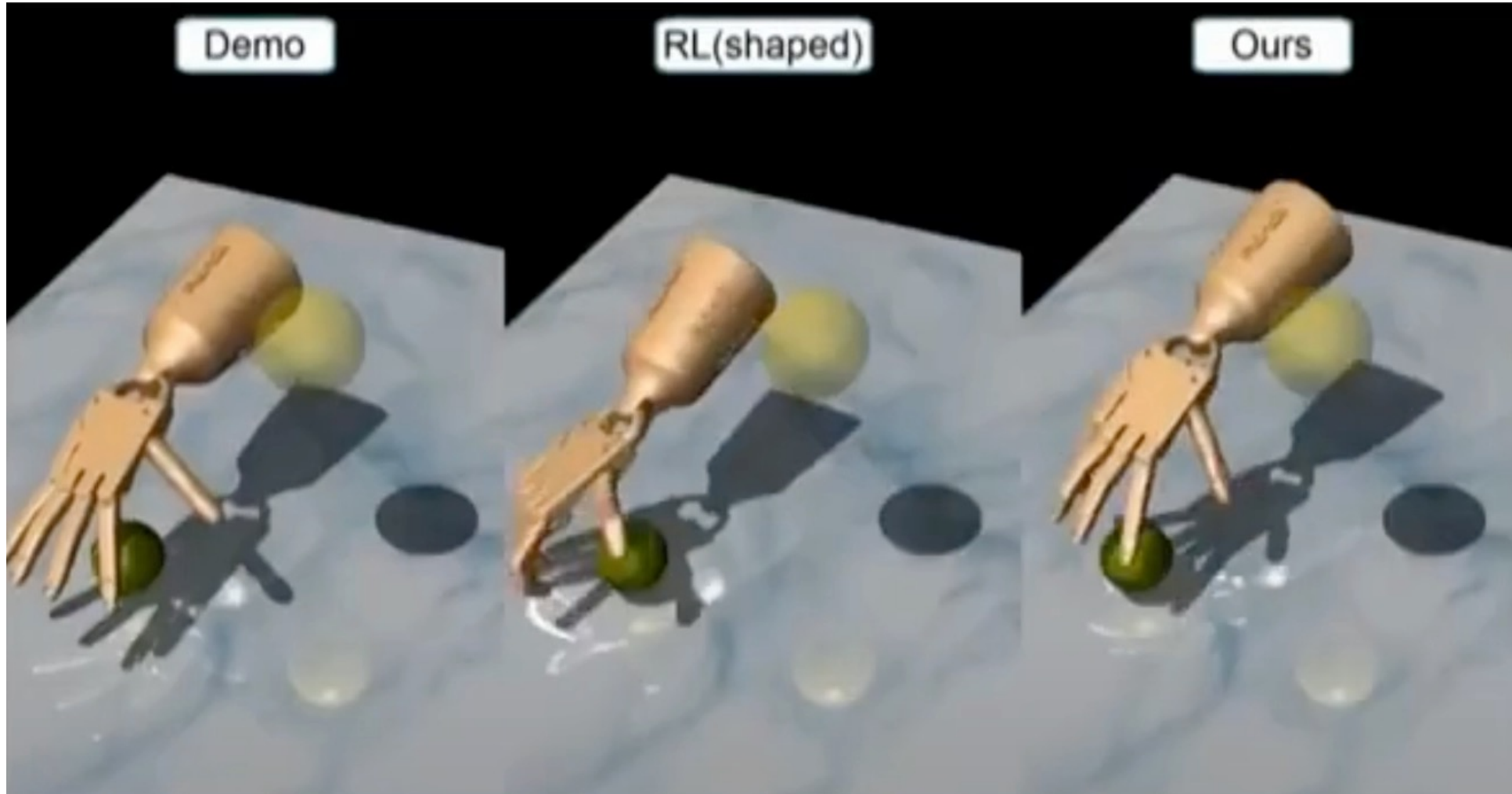
DAPG: Policy Gradient + Imitation

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

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

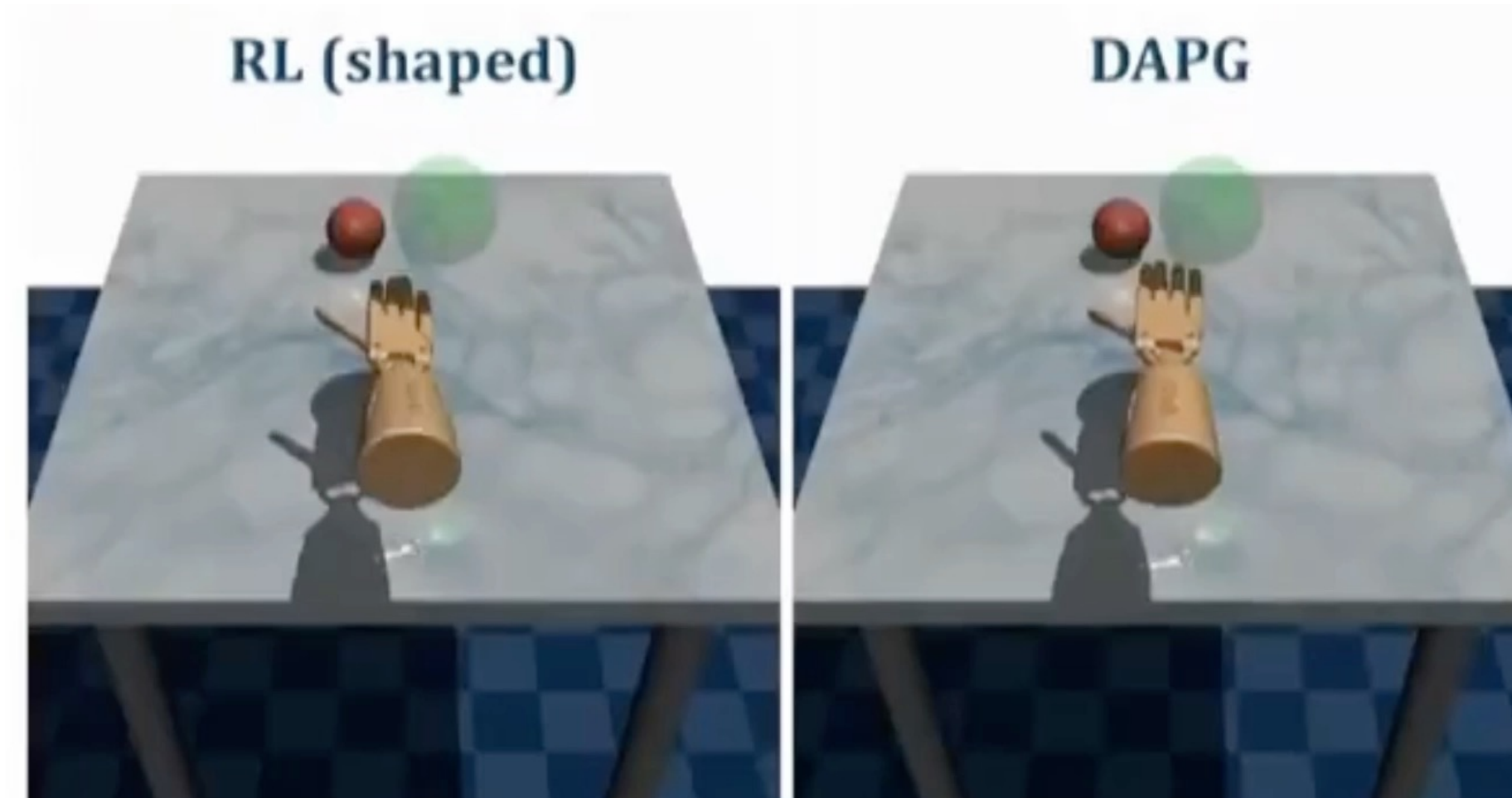
DAPG Benefits

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

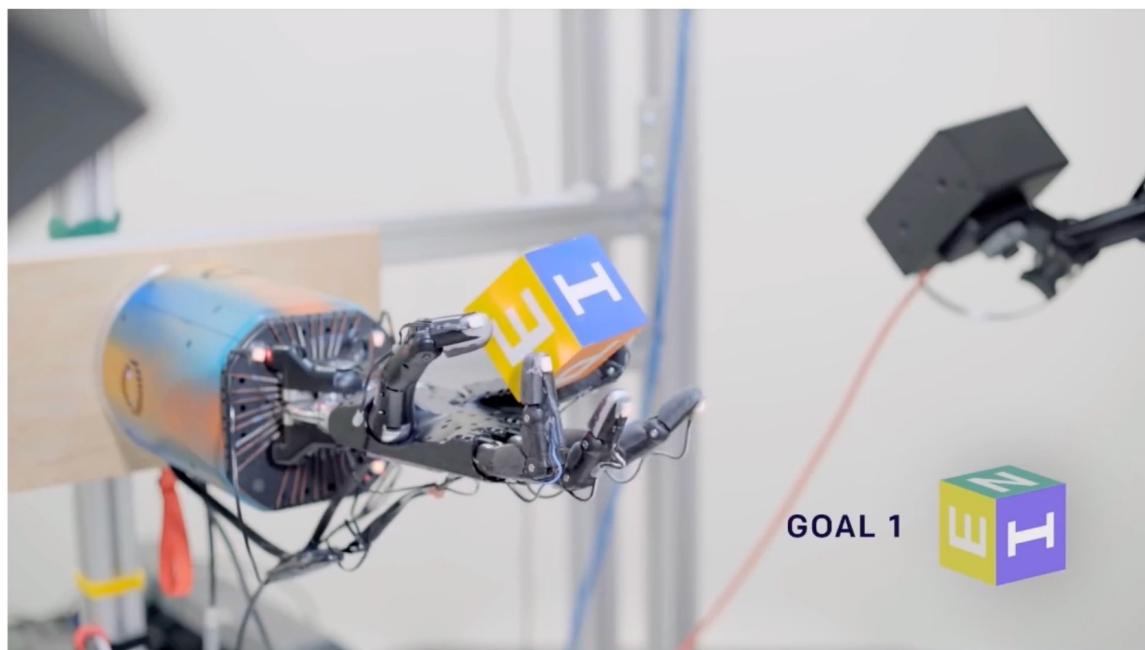


Comparison with Prior Work

Prior Work

Reorient one object, limited generalization to different objects

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

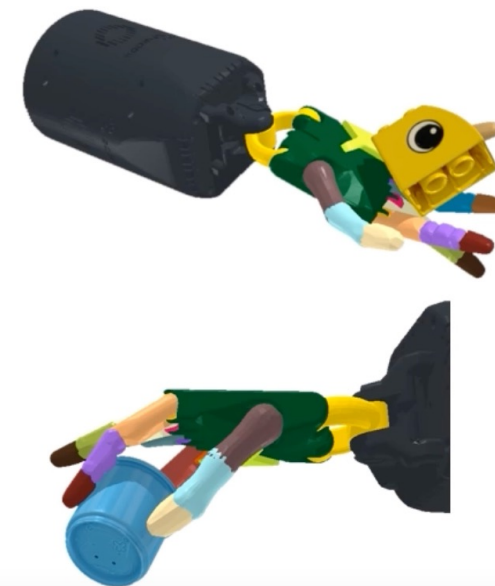


Their work

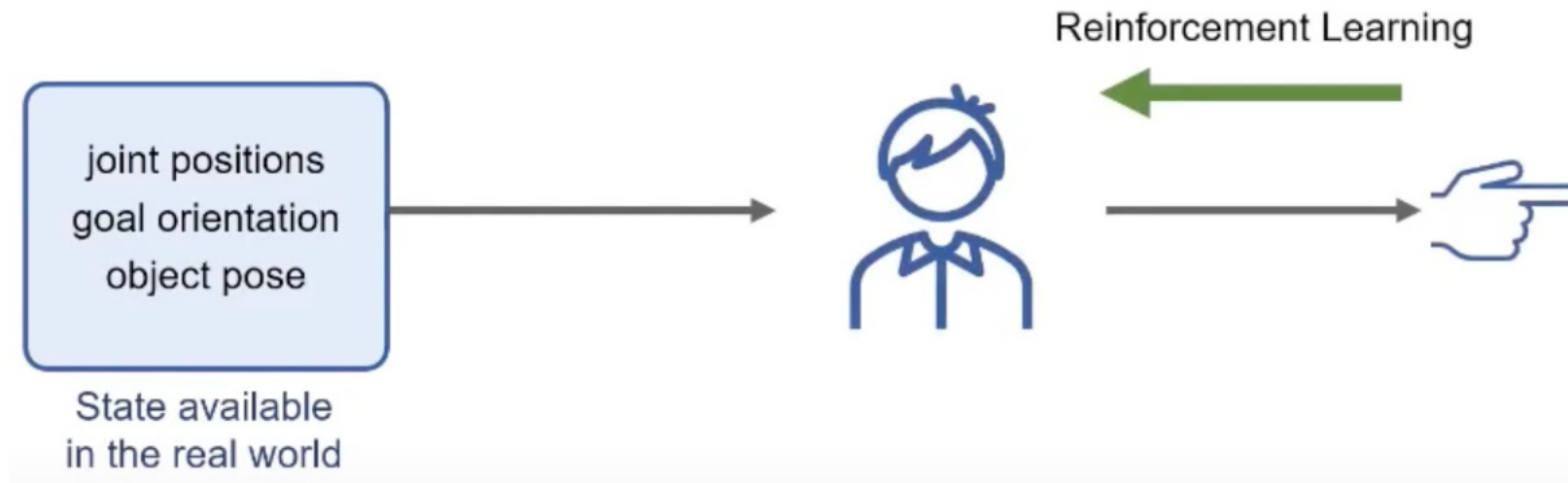
Reorient more than 2000
geometrically different objects



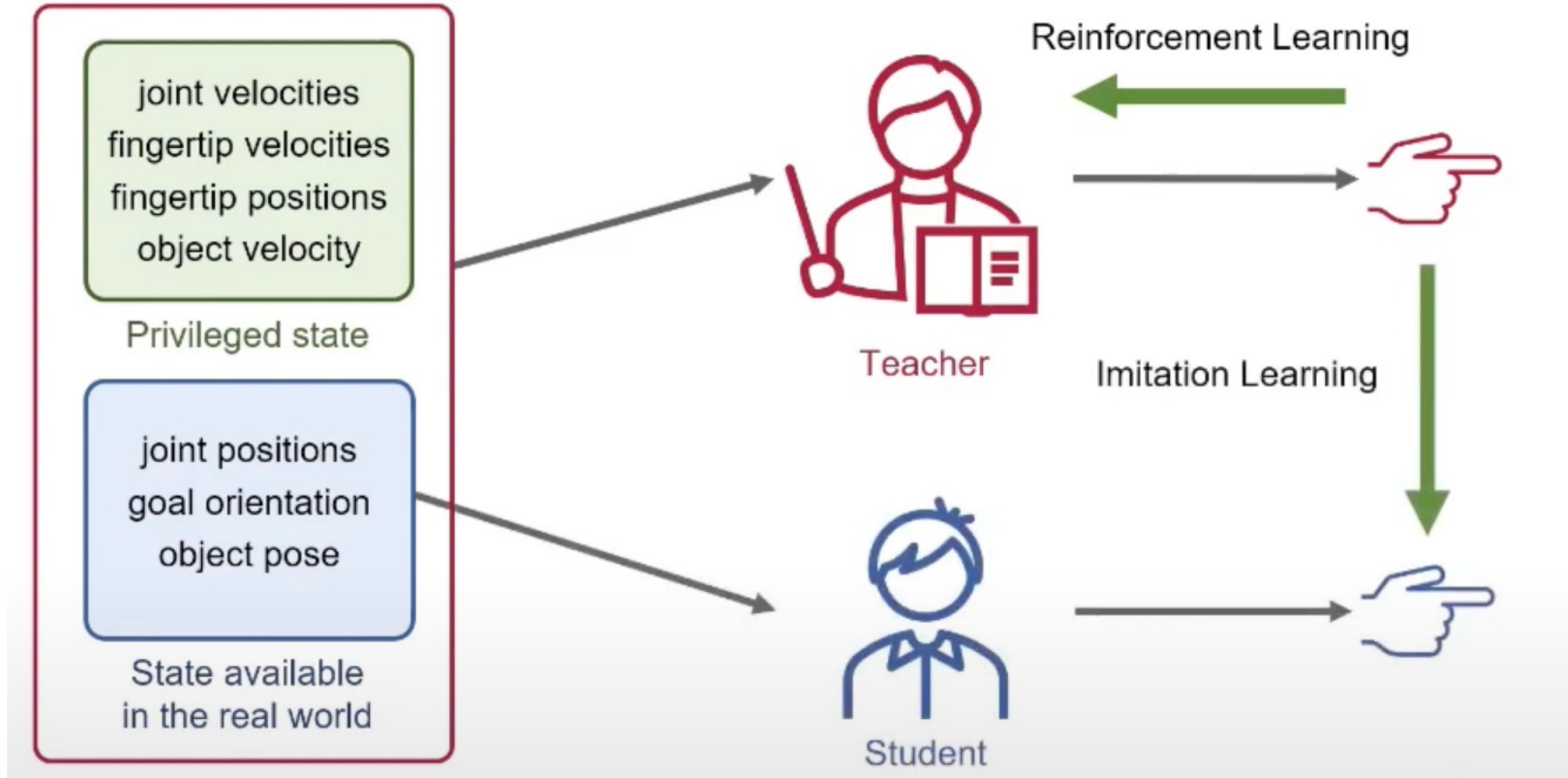
Hand faces upward and downward



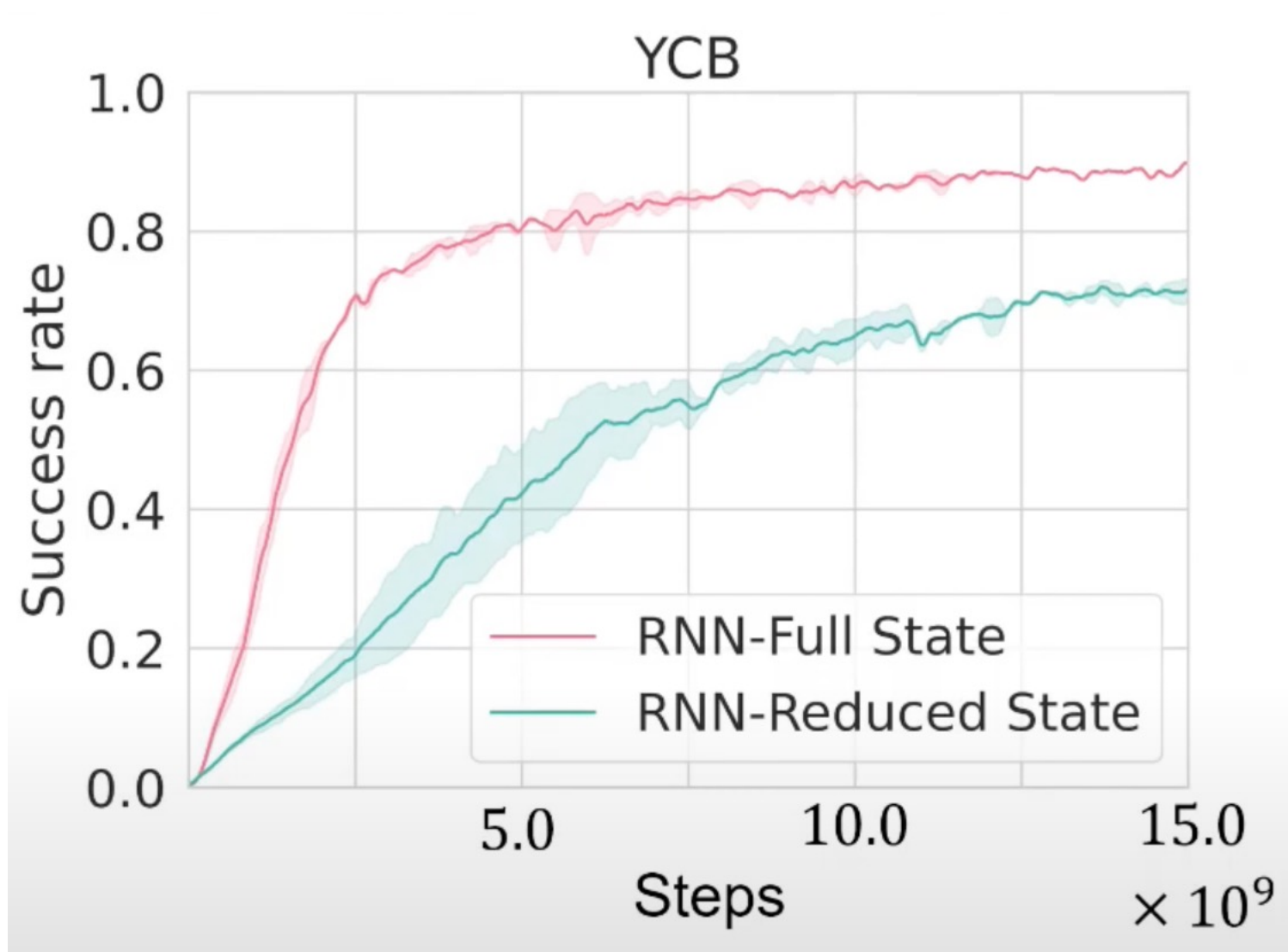
State-based RL



State-based Teacher-Student

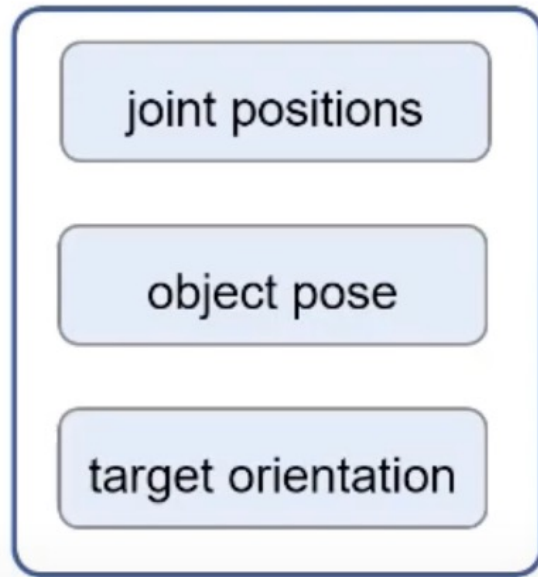


Experiment

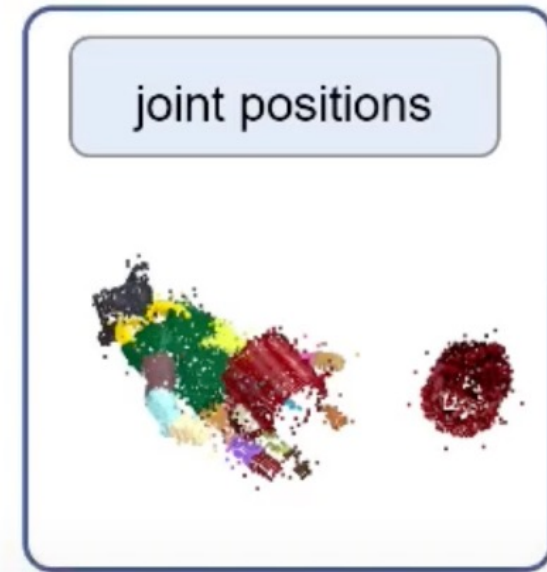


Vision-based Student

Overcome the object pose assumption



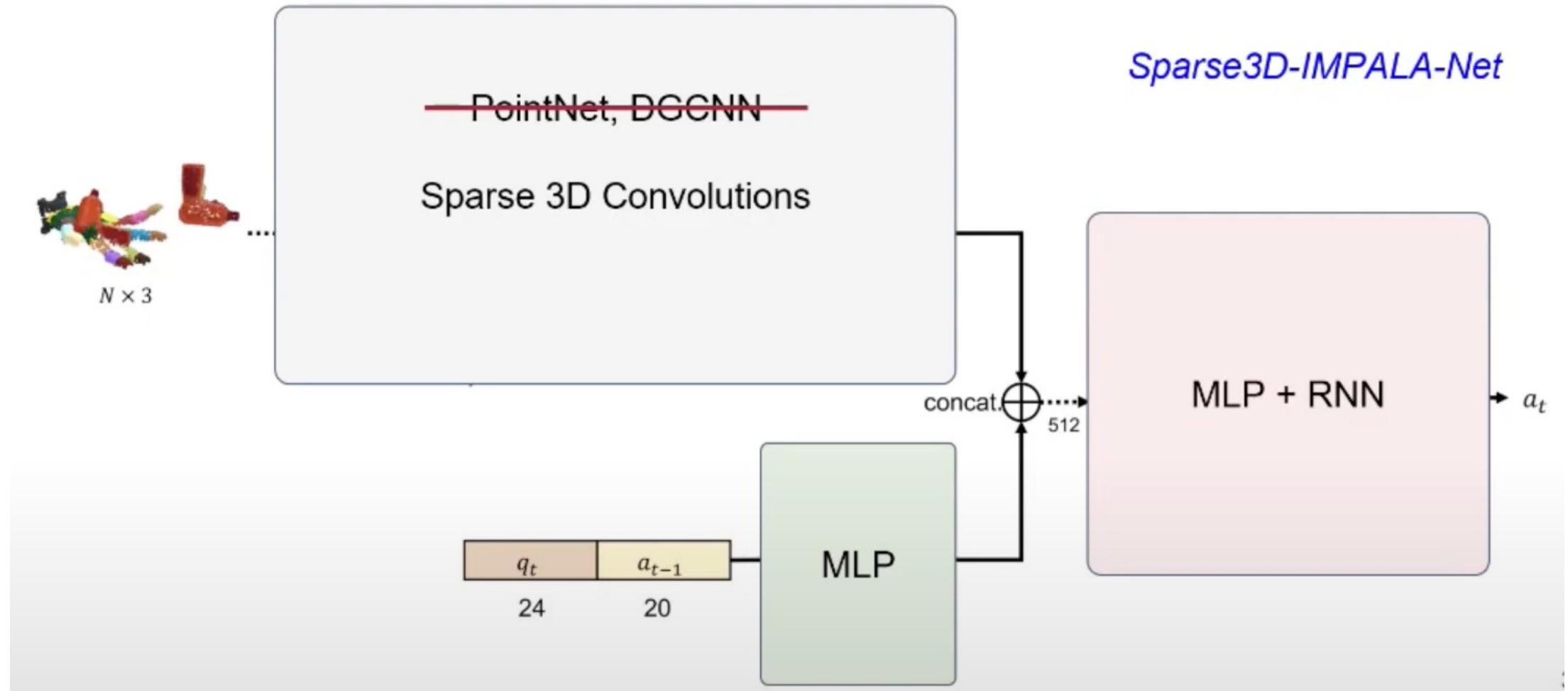
Student Observations
(Low-Dimensional)



Student Observations
(Point Cloud)

Vision-based Student

Student policy with vision input



Experiment

Train and test with noise on point cloud

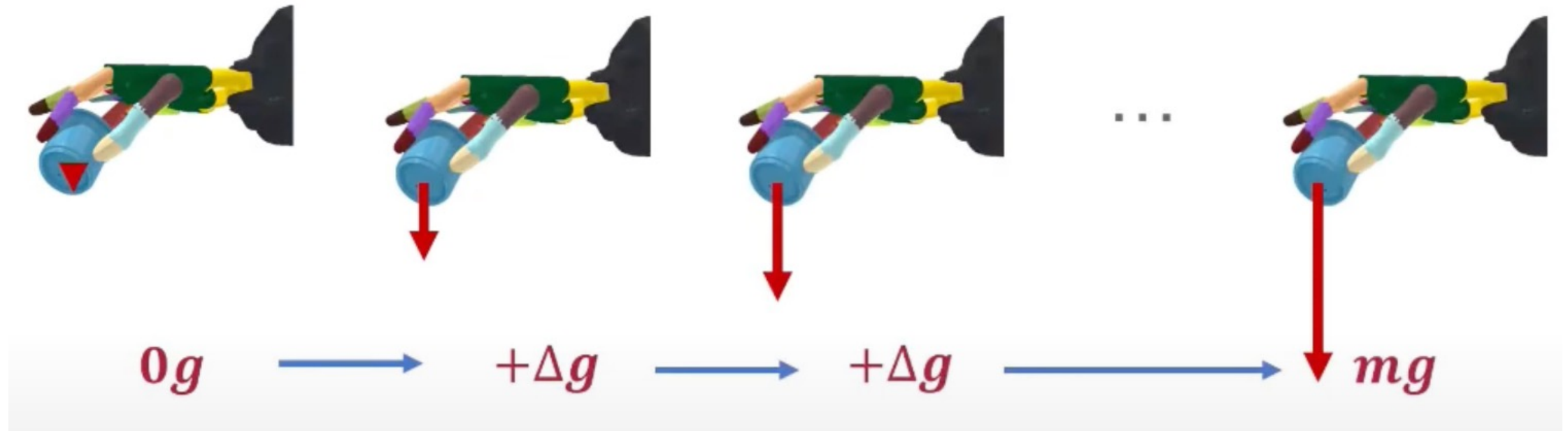
	Object	Success rate (%)
	025_mug	89.67 ± 1.2
	065-d_cups	68.32 ± 1.9
	072-b_toy_airplane	84.52 ± 1.4
	073-a_lego_duplo	58.16 ± 3.1
	073-c_lego_duplo	50.21 ± 3.7
	073-e_lego_duplo	66.57 ± 3.1

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 13, Robotic Tasks III

