Toward Generalizable Robot Manipulation: Cross-Embodiment Learning and Real-World Evaluation

INTELLIGENT ROBOTICS AND VISION LAB
UT DALLAS

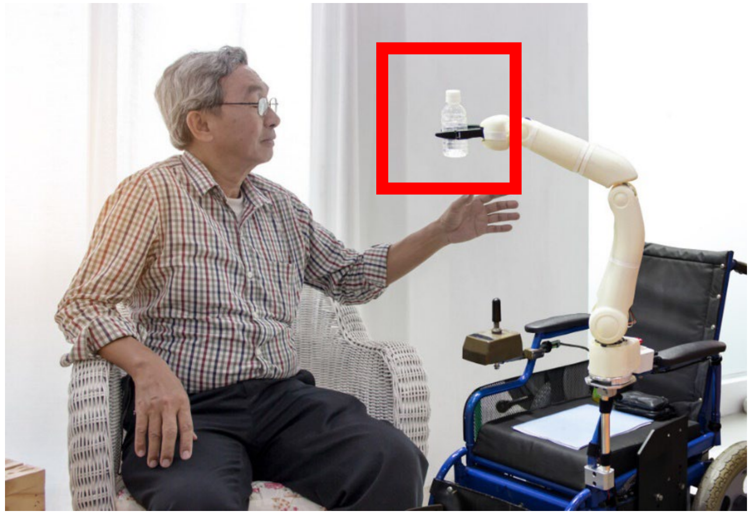
Yu Xiang
Assistant Professor
Intelligent Robotics and Vision Lab
University of Texas at Dallas

9/28/2026

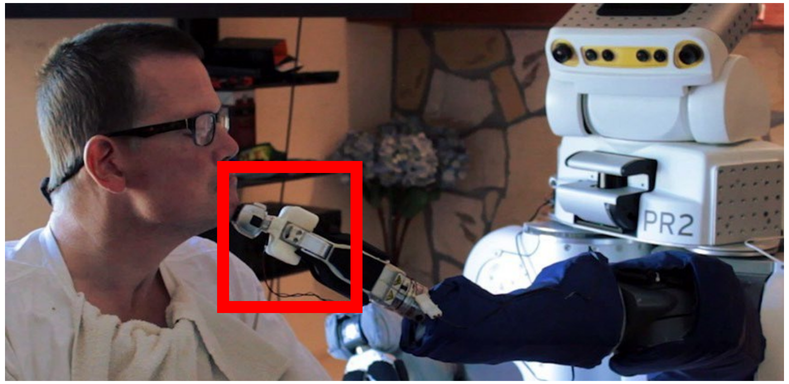
DENSO Pittsburgh Lab

Future Intelligent Robots in Human Environments

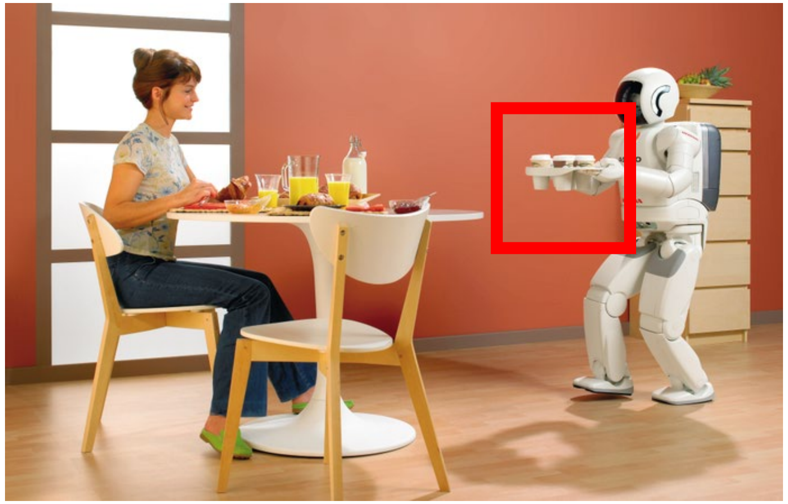
Manipulation



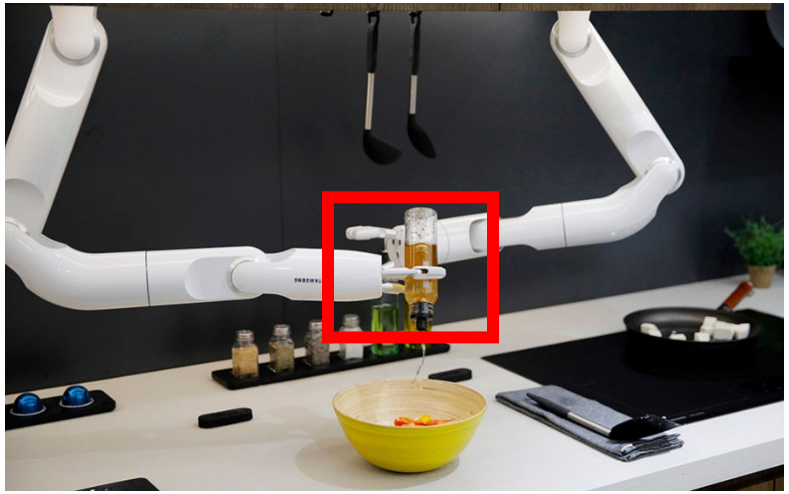
Senior Care



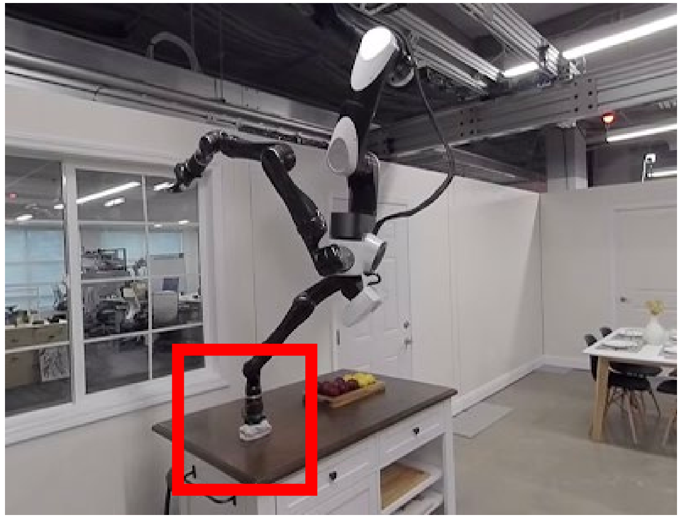
Assisting



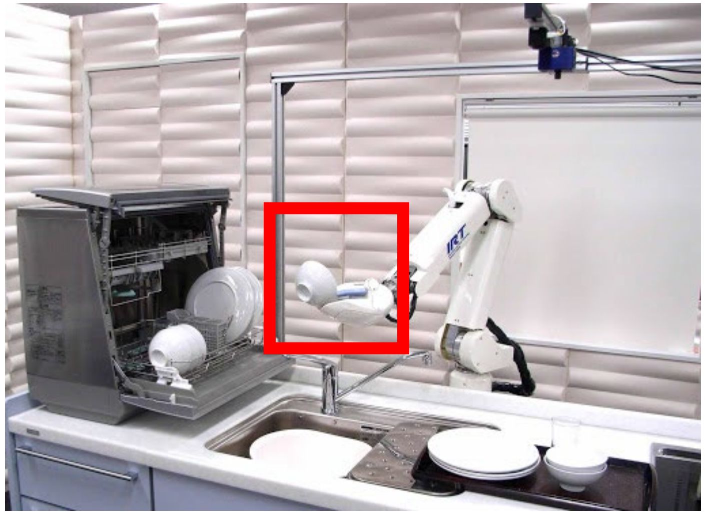
Serving



Cooking



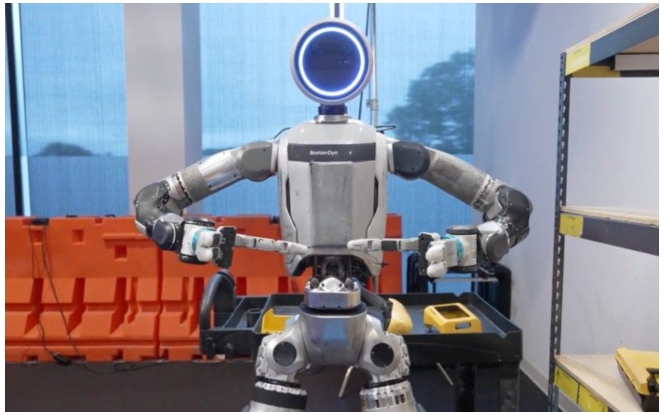
Cleaning



Dish washing

We will have many different robots

Humanoid



Boston Dynamics Atlas

Unitree G1

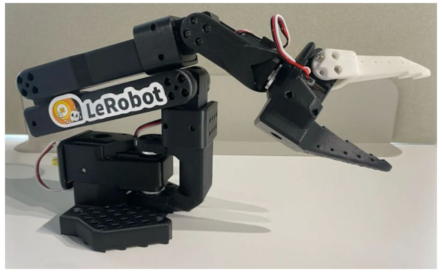
Figure

XPeng Iron

1X

Manipulator

Franka Emika



SO101 arm

Kuka arm

Mobile Manipulator

Fetch

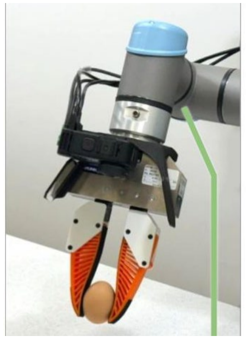
Mobile AI

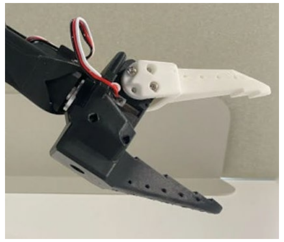
We will have many different grippers/hands


Panda


Fetch

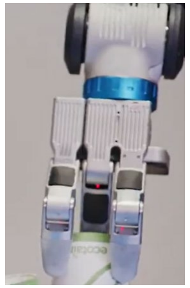

Sawyer


UMI


SO101


Barrett

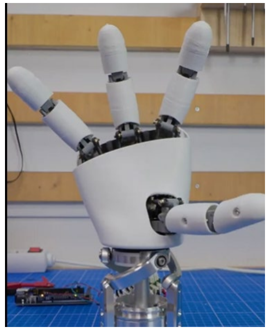

Robotiq

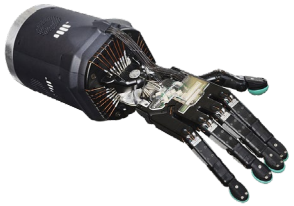

Atlas

Jaco


Allegro


Leap


LeRobot


Shadow


INSIPRE


PAL Hey5 hand


Sharpa

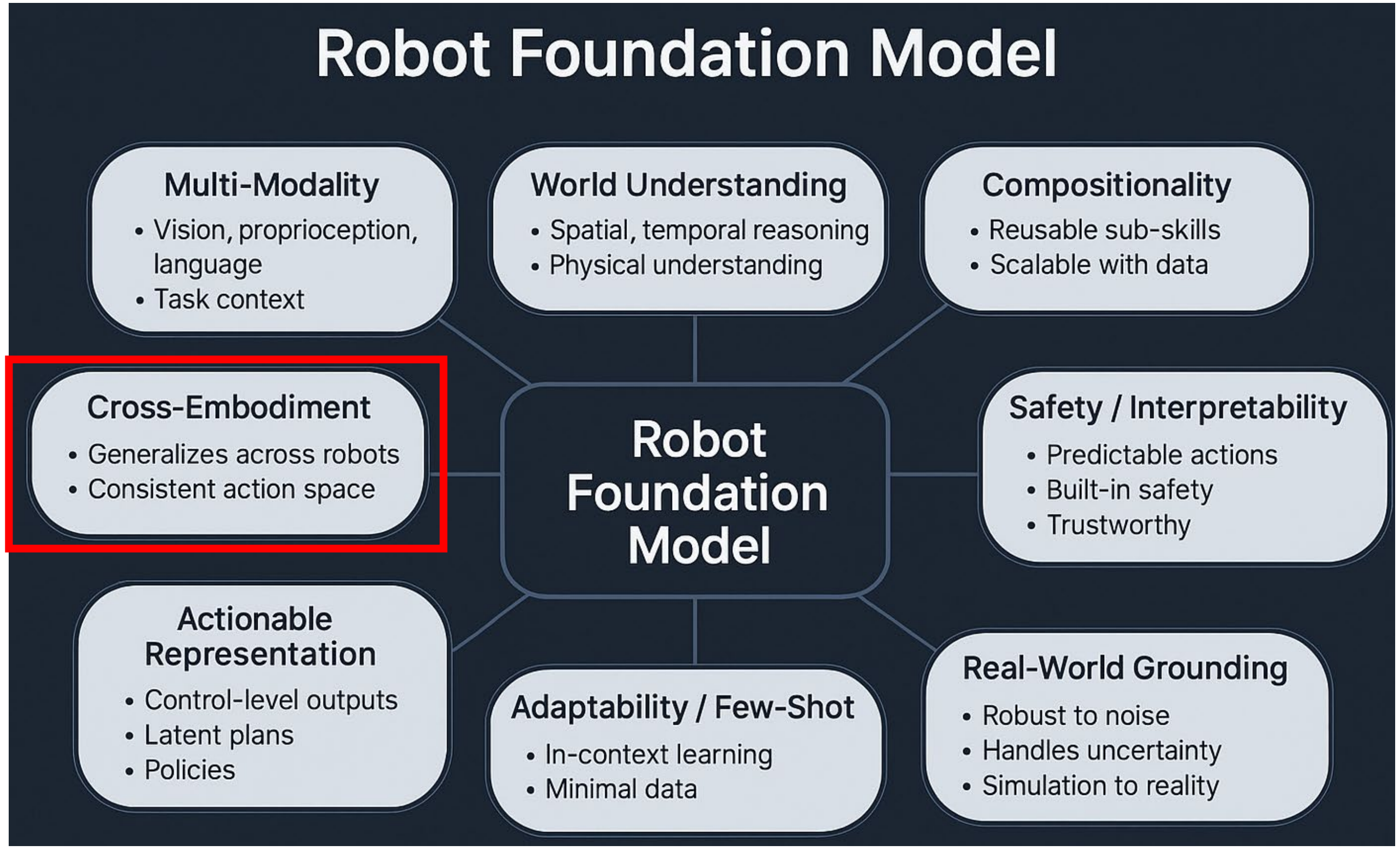

Aero Hand Open

4

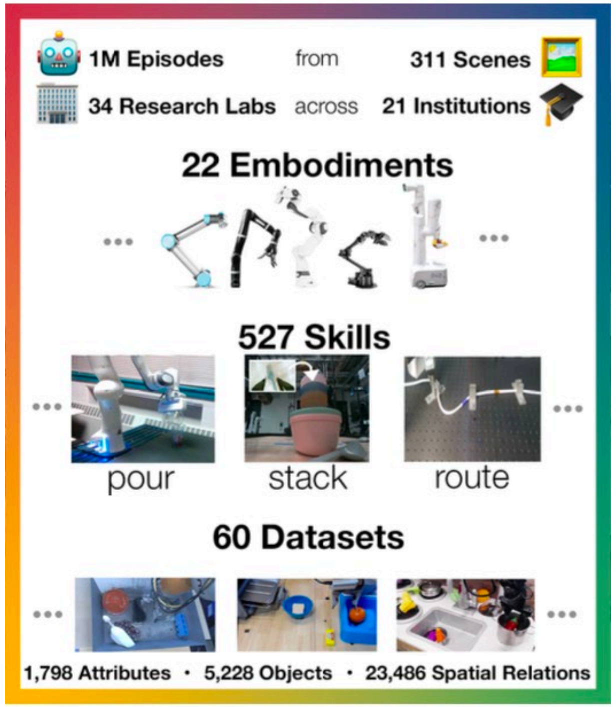
How to make all these robots work?

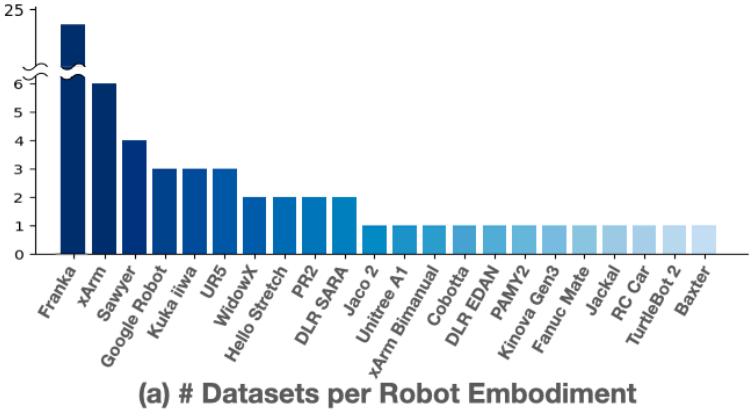
- Robot Foundation Model
  - A **foundation model** in AI refers to a **large, pre-trained model** that serves as a *base* (or "foundation") for building many downstream applications and specialized models.

Our recent focus



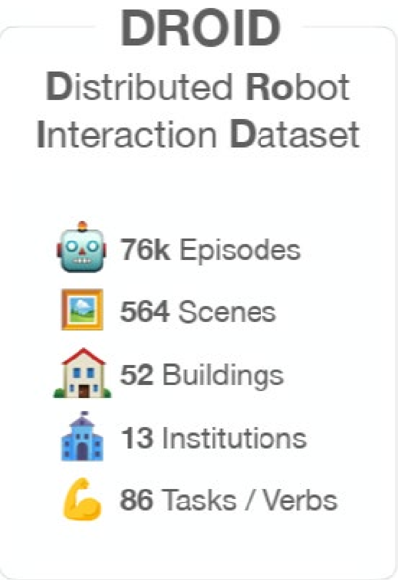
Current Robot Data

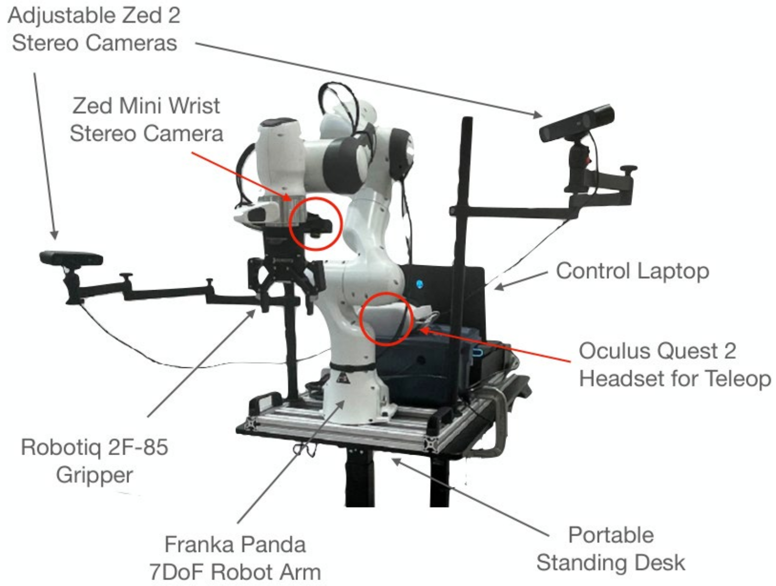




Open X-Embodiment

- Franka
- xArm
- Sawyer
- Google Robot
- Kuka iiwa
- UR5
- WidowX
- Hello Stretch
- PR2
- DLR SARA
- Jaco 2
- Unitree A1
- xArm Bimanual
- Cobotta
- DRL EDAN (5-finger)
- PAMY2
- Kinova Gen3
- Fanuc Mate
- Jackal
- RC Car
- TurtleBot 2
- Baxter





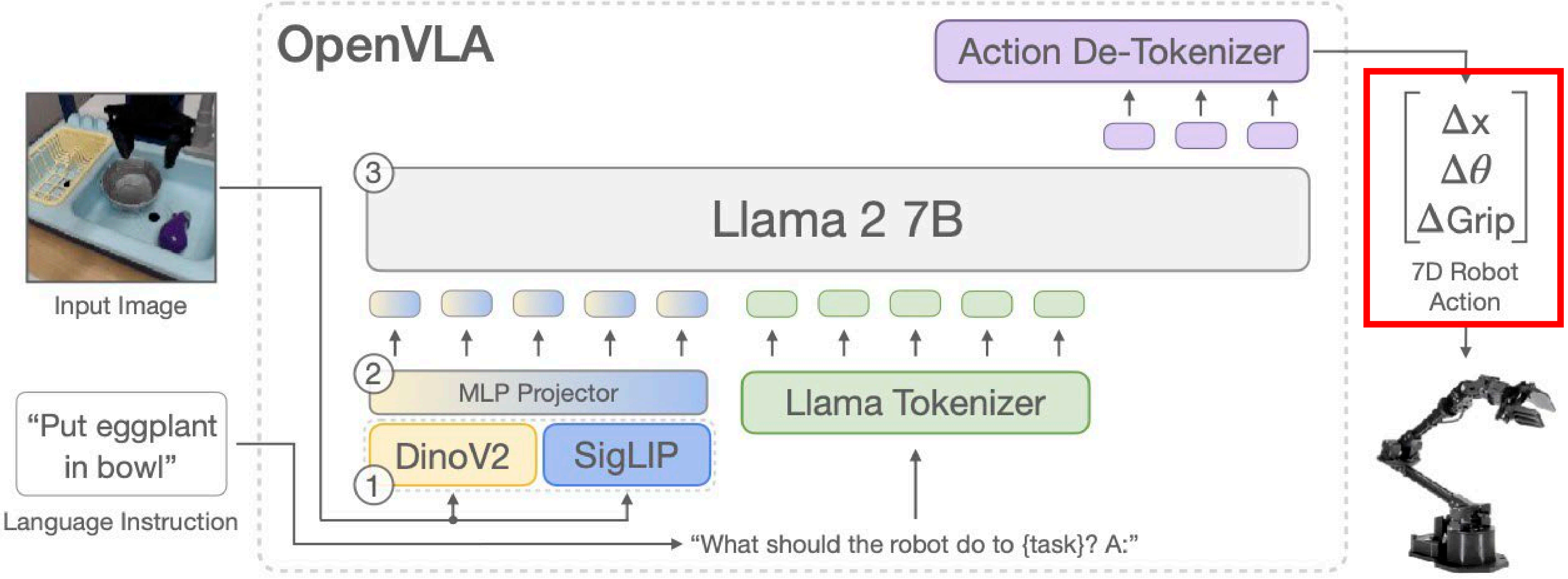
The DROID dataset

Very biased to 2-finger grippers

Current Model Architecture

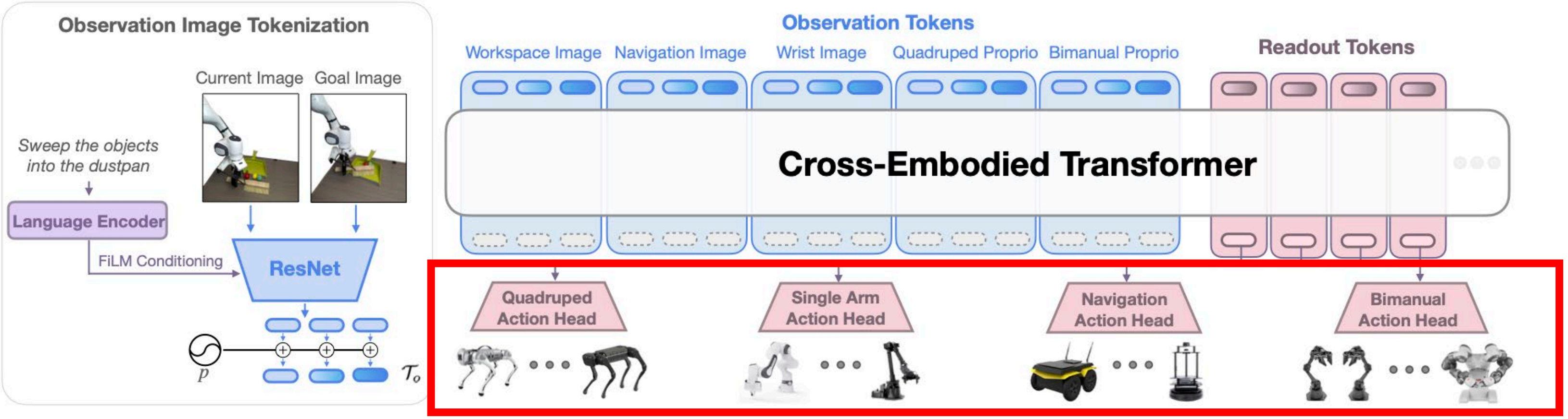
Action

- Gripper pose for two-finger grippers
- Cannot be used for multi-finger hands (no hand joints)

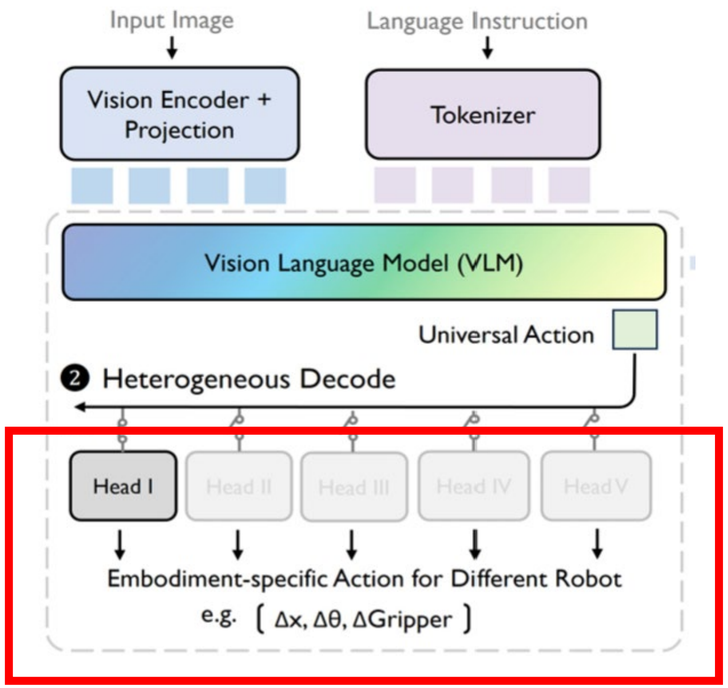


OpenVLA: An Open-Source Vision-Language-Action Model. Kim et al., 2024.

Cross-Embodiment Model Architecture



CrossFormer: Scaling Cross-Embodied Learning for Manipulation, Navigation, Locomotion, and Aviation. Doshi et al., CoRL, 2024.

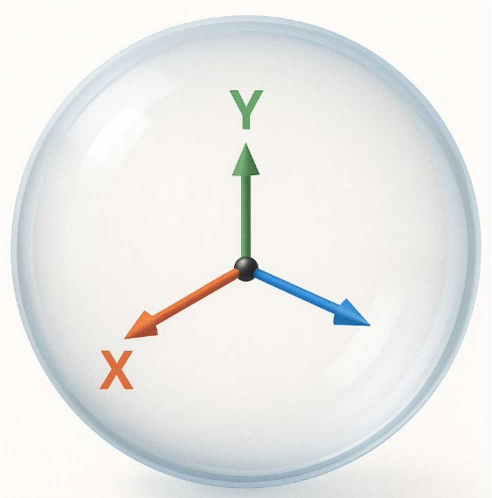


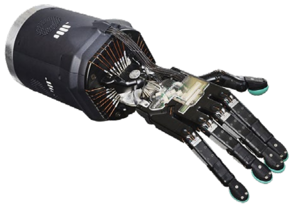
One Action head for each robot type

- Given a new robot, one new head needs to be trained

Universal Actions for Enhanced Embodied Foundation Models. Zheng et al., CVPR, 2025.

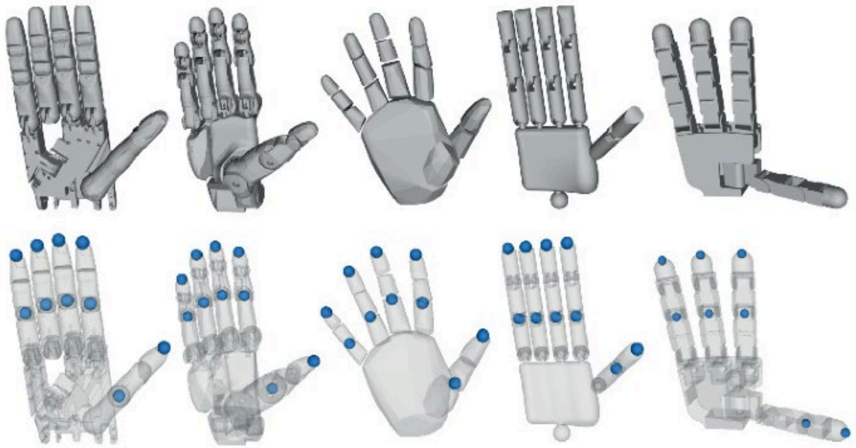
Can we find a unified action space for different robot hands?



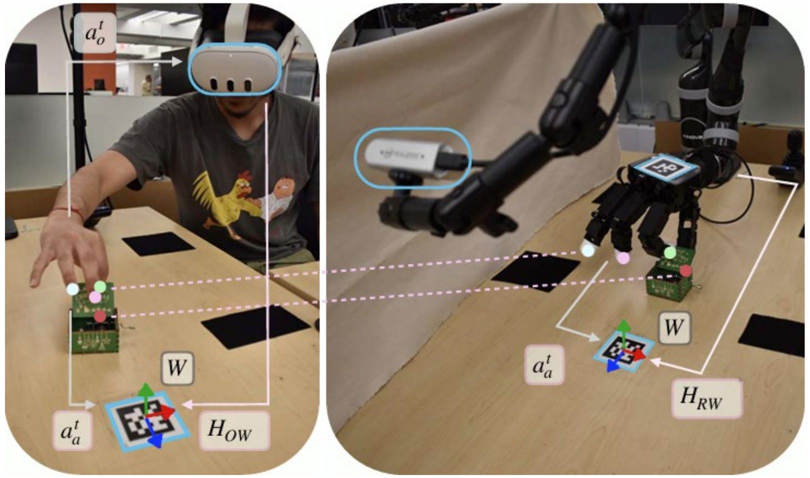


Human Hand
(human data)

Previous work: manual alignment of hands (retargeting)



Learning Cross-hand Policies for High-DOF Reaching and Grasping (She et al., 2024)



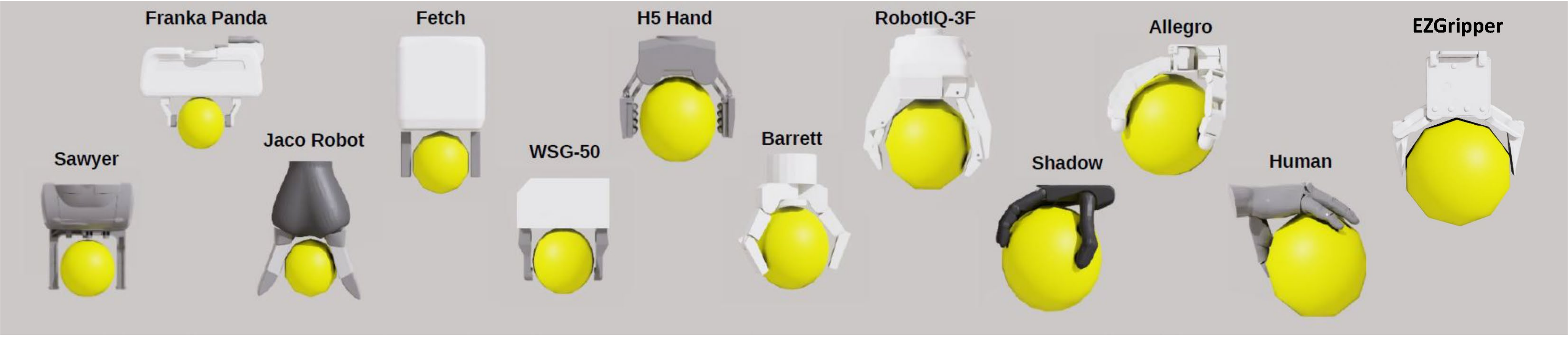
HuDOR, Guzey et al. NYU 2025

https://object-rewards.github.io/

- Manual mapping or hand-designed correspondence
- Hard to deal with different number of fingers
- Cannot handle unseen hands

Our idea: Let’s use a sphere to align hands

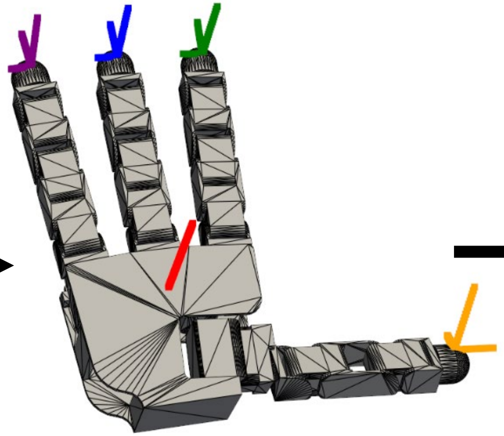
- Because any hand can grasp a sphere! (otherwise, it might not be that useful for manipulation)
- Spheres have some good properties for control (you will see)



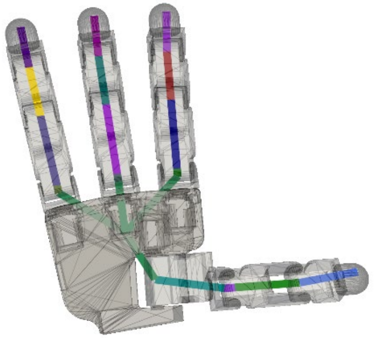
A unified action space for different robot hands

- Sphere creation

Hand URDF

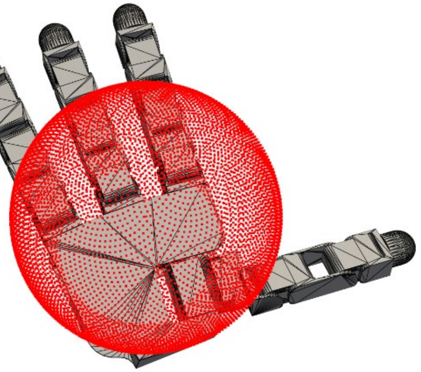


Hand URDF
- Frames for palm center and fingertips

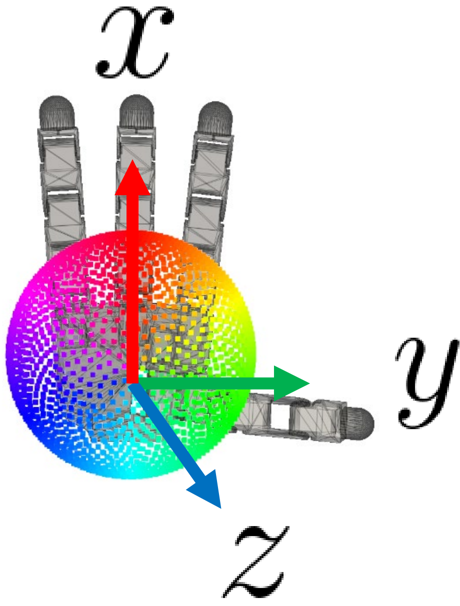


$$l = \frac{l_1 + l_2 + l_3 + l_4}{4}$$

Radius $r = l\frac{2}{\pi}$



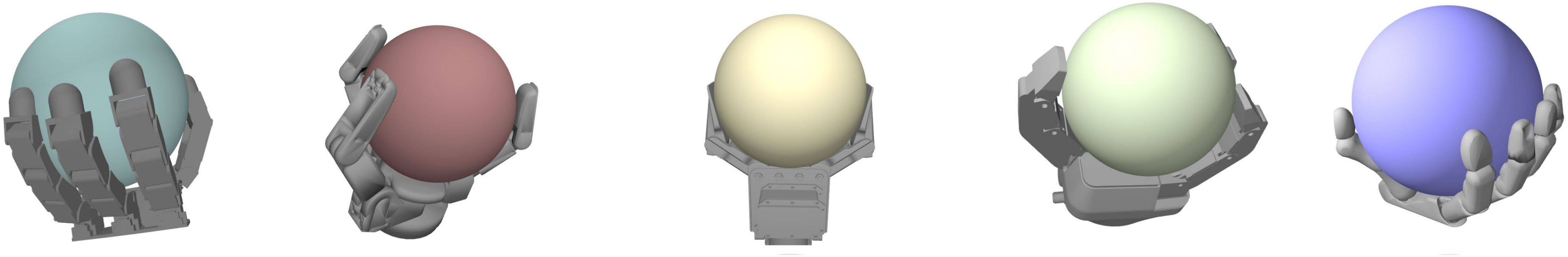
Sphere center above the palm center by r



x-axis towards the middle finger

A unified action space for different robot hands

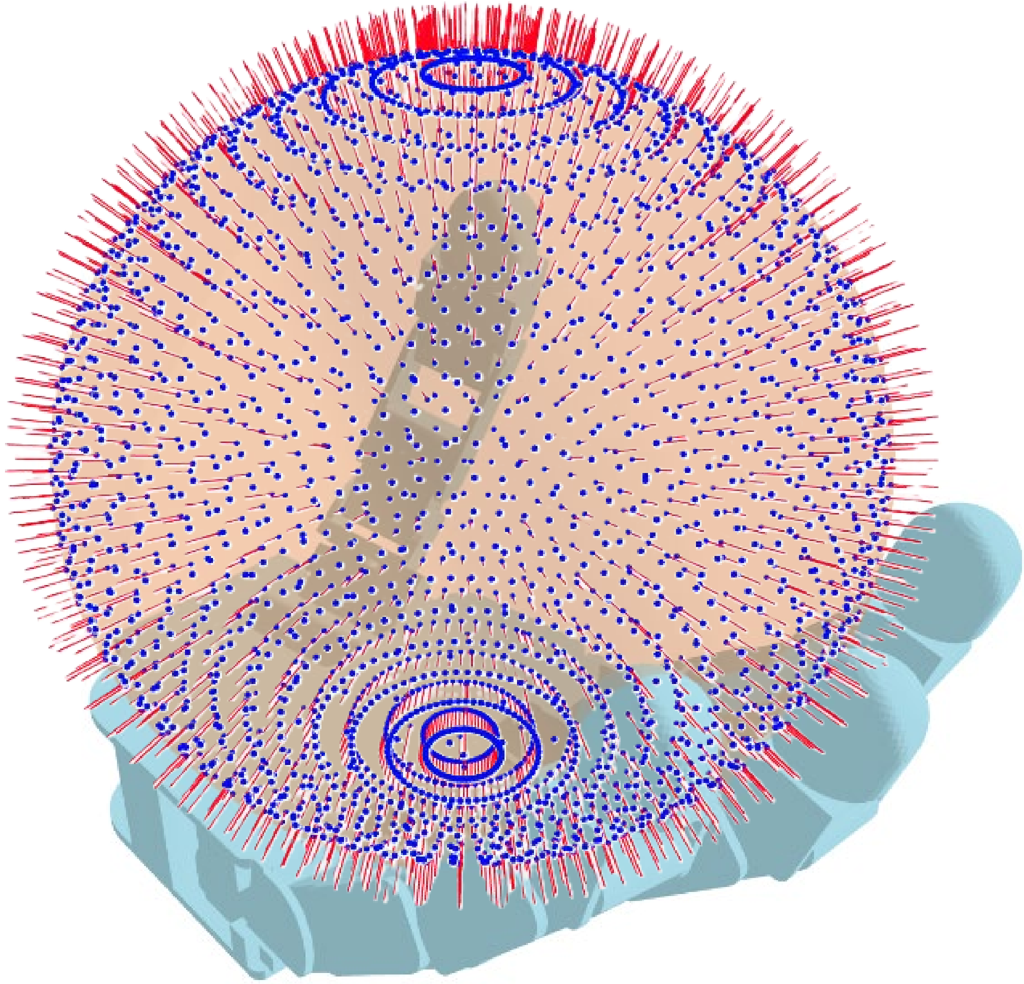
- Sphere creation applies to different hands
- Close the fingers to grasp the sphere

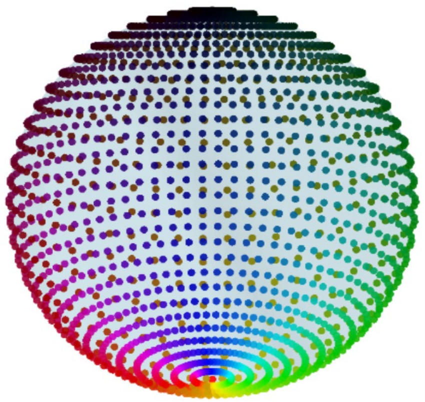


Human Hand

A unified action space for different robot hands

- Map spherical coordinates to the hand

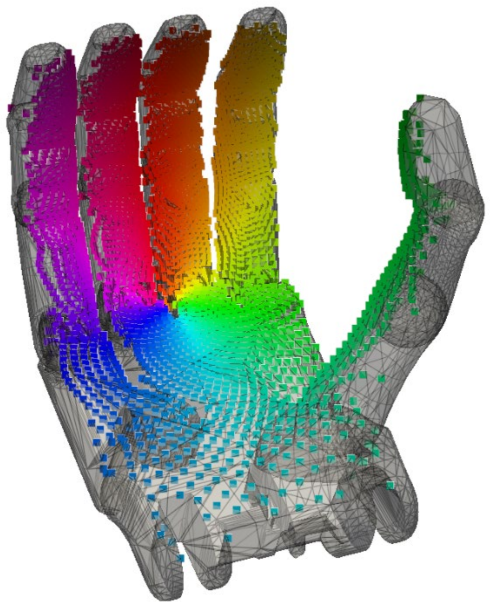


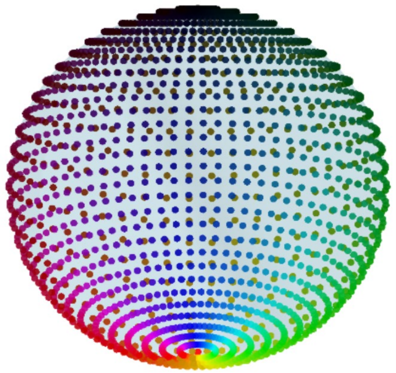


(λ, ϕ)

A unified action space for different robot hands

- Map spherical coordinates to the hand (a representation of the hand)





(λ, ϕ)

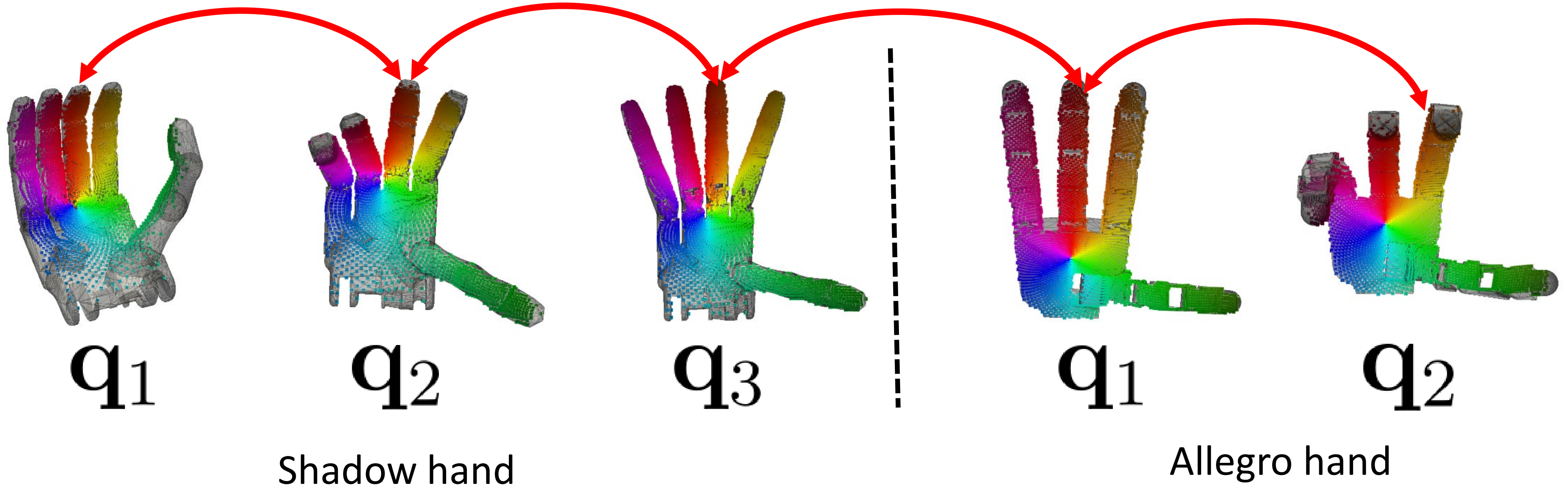
A hand G is represented by a set of interior points $P_G = \{\mathbf{v}_g \mid \mathbf{v}_g \in \mathbb{R}^3\}$

Each point $\mathbf{v}_g$ is associated with a spherical coordinate (λ, ϕ)

Spherical coordinates $\Phi_G = \{(\lambda_{\mathbf{v}_g}, \phi_{\mathbf{v}_g}) | \mathbf{v}_g \in P_G; \lambda_{\mathbf{v}_g}, \phi_{\mathbf{v}_g} \in [0, 1]\}$

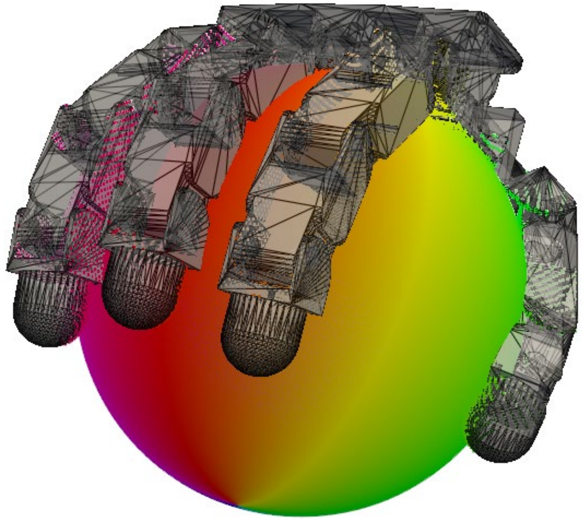
A unified action space for different robot hands

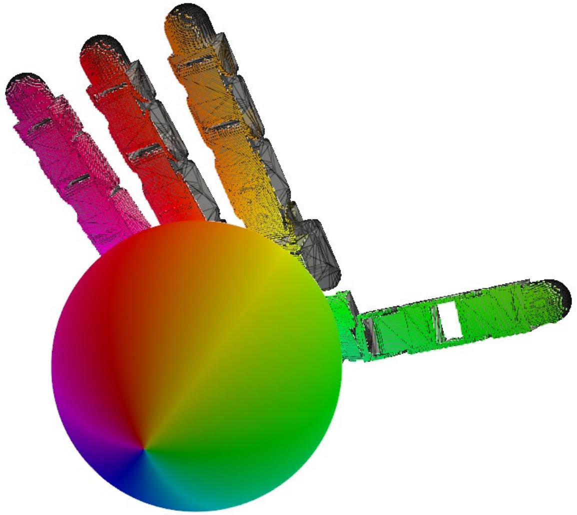
- Property 1: the locations of the hand points change according to grasp configuration $P_G = \{\mathbf{v}_g \mid \mathbf{v}_g \in \mathbb{R}^3\}$ $\quad P_G(\mathbf{q})$



- Property 2: the spherical coordinate for each point **remains the same across configurations and hands** (correspondences!!)

Unified Hand Action Space (UHAS)



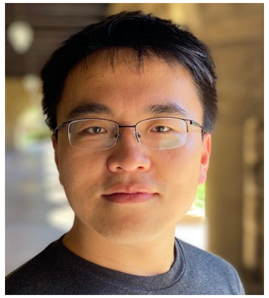


Luis Felipe Casas

Robert Teal

Keval Shah

Abhijit Tadepalli
ASU



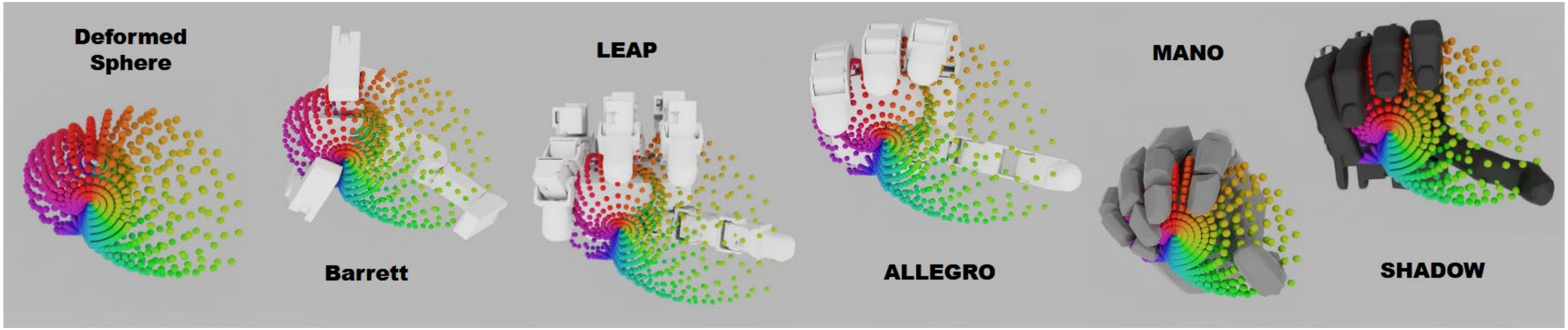
Wanxin Jin
ASU

Can we use this sphere to control any robotic gripper/hand?

[4th Workshop on Dexterous Manipulation]() at Robotics: Science and Systems (RSS), 2026

Unified Hand Action Space (UHAS)

- **Our Idea:** the deformation of the sphere will drive the movement of the hand (the hand should touch the deformed sphere correctly)
- Action space: deformation of the sphere (shared by any hand!)

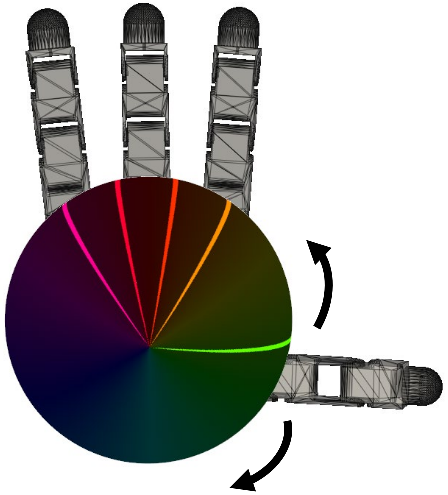


Different hands with the same deformed sphere

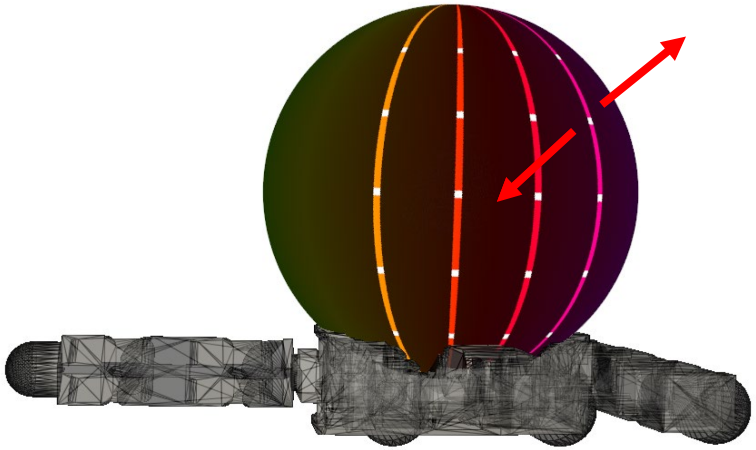
UHAS: deforming the sphere

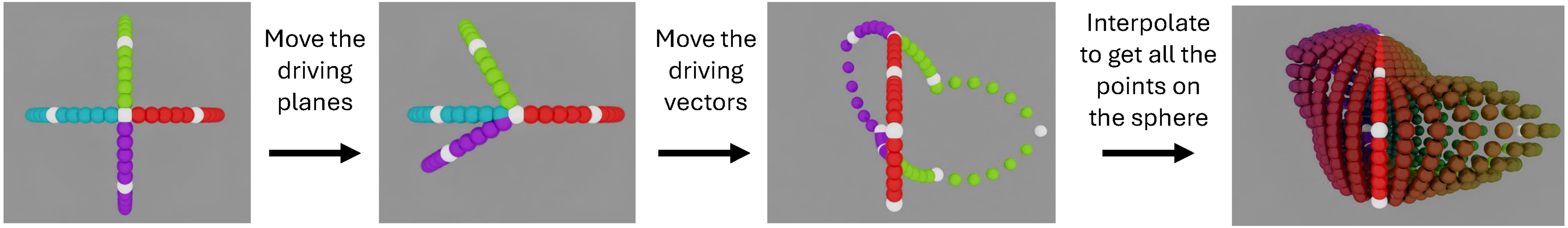
- Deforming every point on the sphere is too expensive for control

Define several "driving planes"



Define several "driving vectors" on each driving plane





(a) Top View
4 driving planes

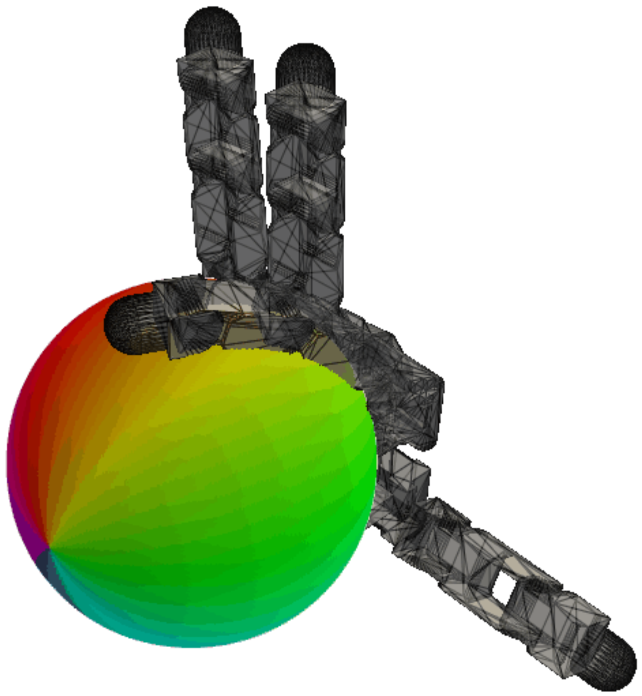
(b) Top View
4 driving planes

(c) Side View
4 driving planes

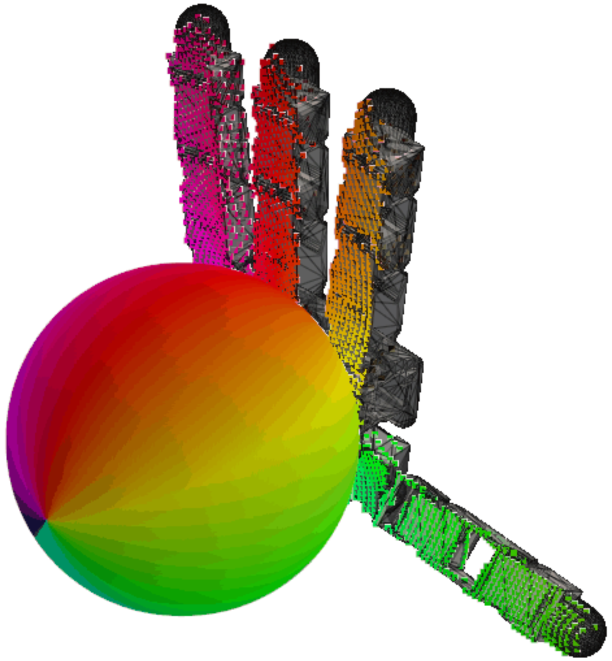
(d) Side View
deformed sphere

UGAS: Cascaded Inverse Kinematics (CIK)

- Given a deformed sphere, we solve IK to obtain the hand configuration



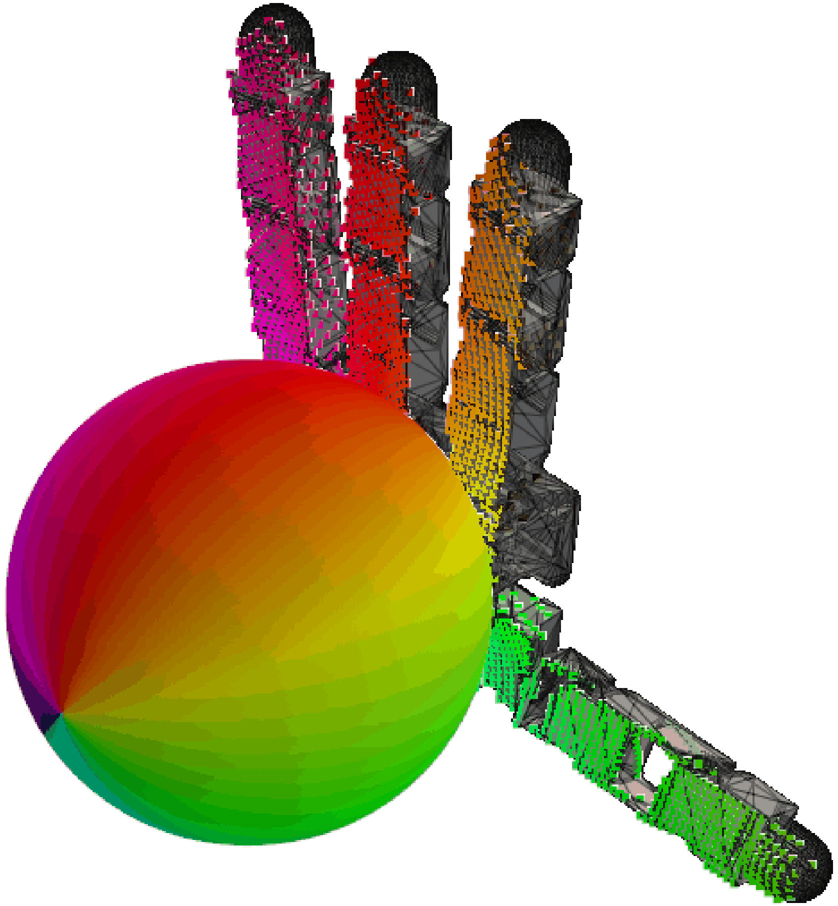
1. Solve lateral joints

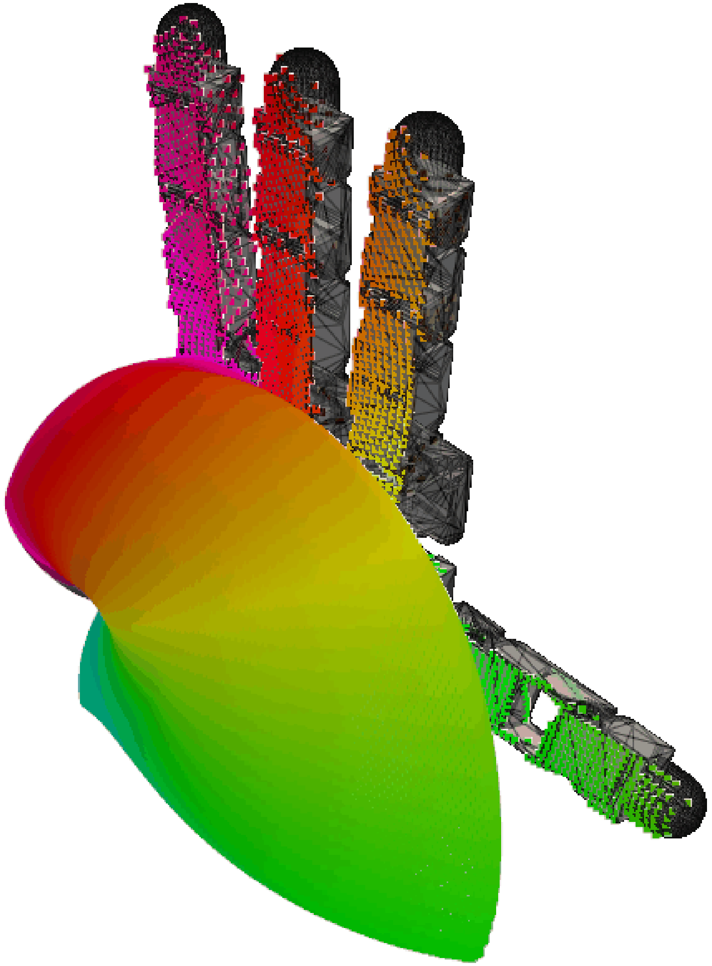


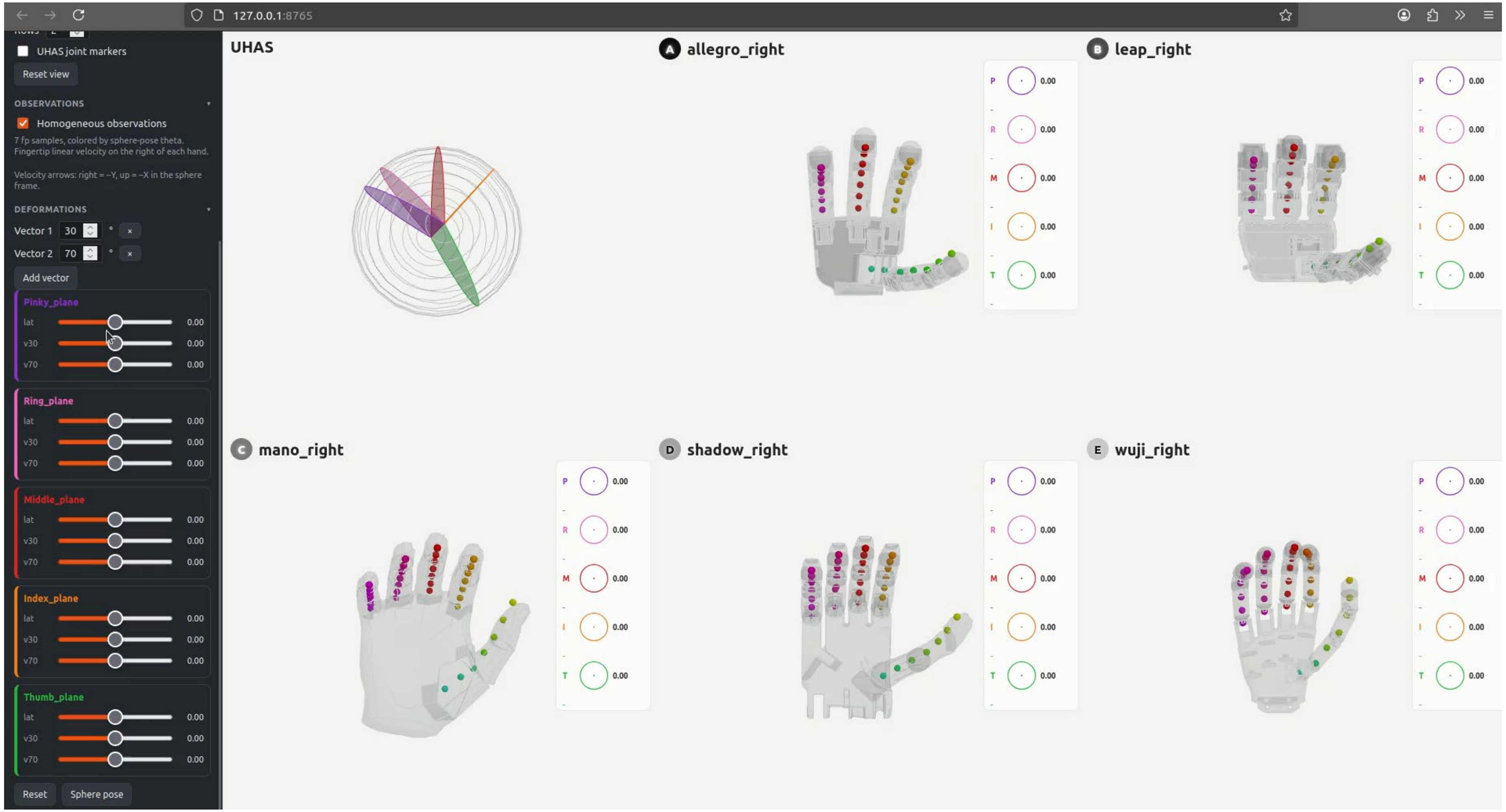
2. Solve encompassing joints

We solve for each joint one at a time, in the order of the kinematic tree.

UGAS: Cascaded Inverse Kinematics (CIK)







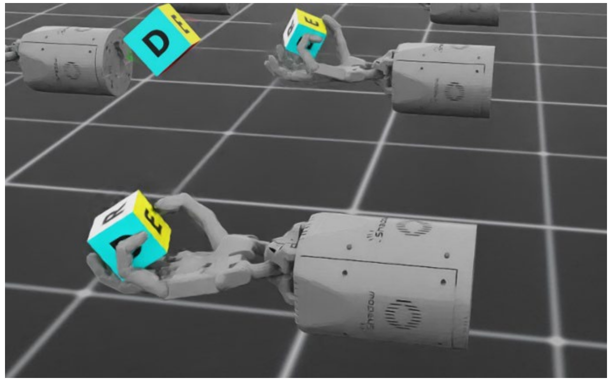
Control actions

Driving planes $\Delta\theta$

Driving vectors Δr

Inverse Kinematics $\longrightarrow$ Hand configuration $\mathbf{q}$

UHAS for In-hand Manipulation



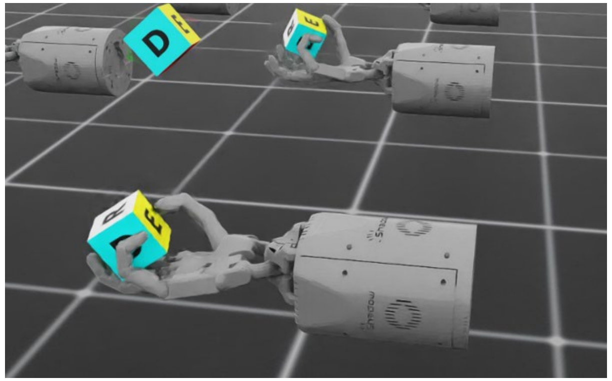
- Task: repose a cube to a target orientation
  - 10 consecutive reposing within 30 seconds
  - RL training in Isaac Lab with PPO and our sphere controller

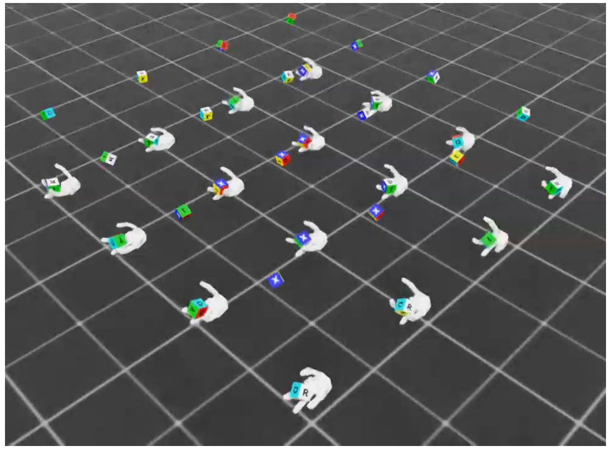
Action

State s_t → Policy → [Driving planes $\Delta\theta$; Driving vectors Δr] $\xrightarrow{\text{Cascade Inverse Kinematics}}$ Hand configuration $\mathbf{q}$

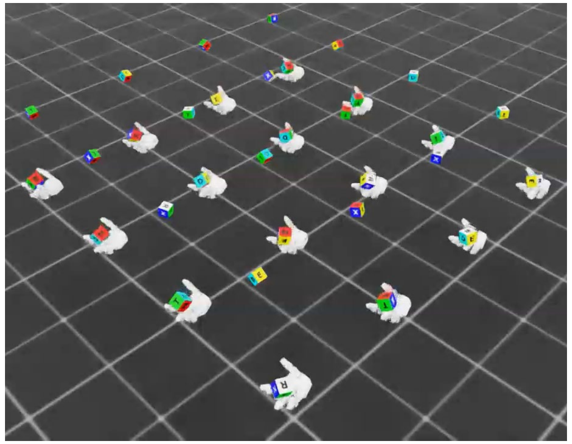
UHAS for In-hand Manipulation

- Task: repose a cube to a target orientation
  - 10 consecutive reposing within 30 seconds
  - RL training in Isaac Lab with PPO and our sphere controller

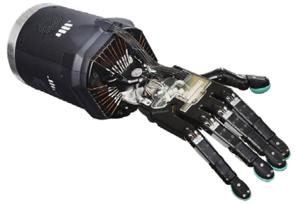


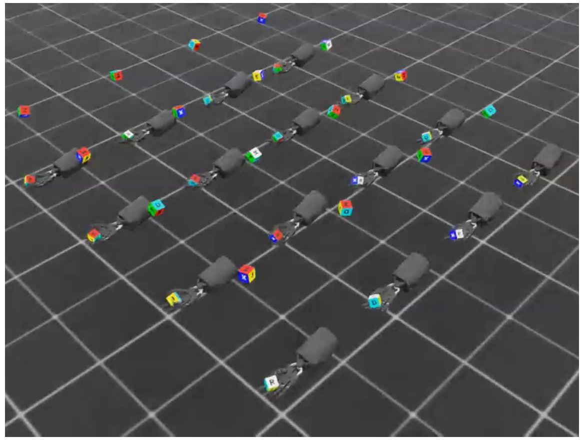


Allegro (4 fingers)
9.6 ± 1.7

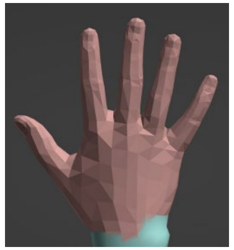


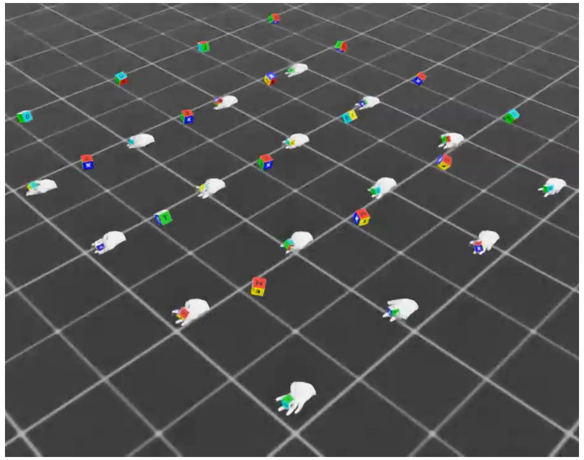
Leap (4 fingers)
9.8 ± 1.1





Shadow Hand (5 fingers)
9.6 ± 1.6



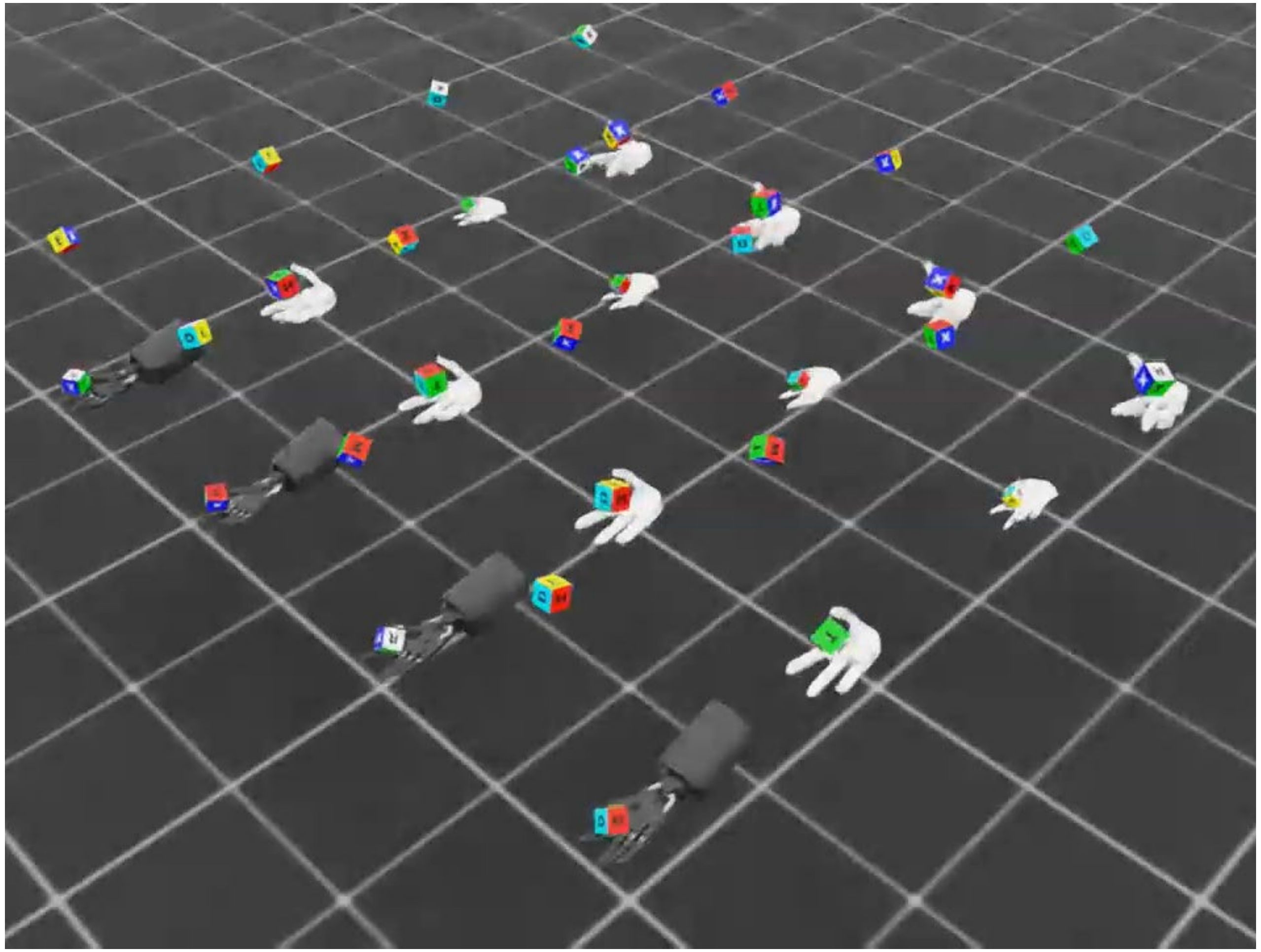


MANO human (5 fingers)
9.9 ± 1.0

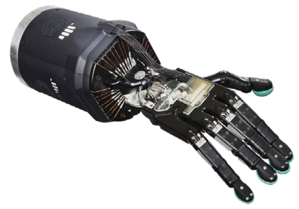
UHAS for In-hand Manipulation

Allegro (4 fingers)
9.5 ± 1.9

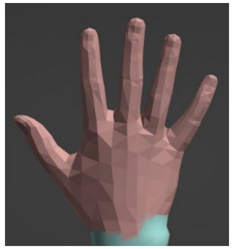
Leap (4 fingers)
9.5 ± 1.9



Training on all 4 hands



Shadow Hand (5 fingers)
9.2 ± 2.3



MANO human (5 fingers)
9.8 ± 1.2

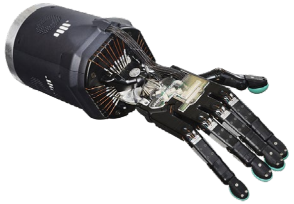
Zero-Shot Policy Transfer (3 In Domain -1 Out of Domain)

Training

Testing

Allegro (4 fingers)
7.7 ± 3.4

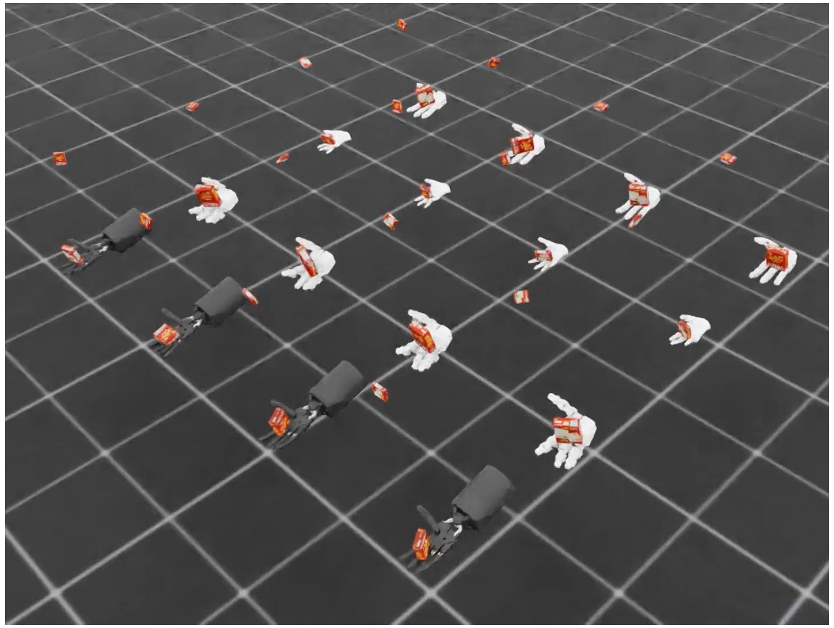
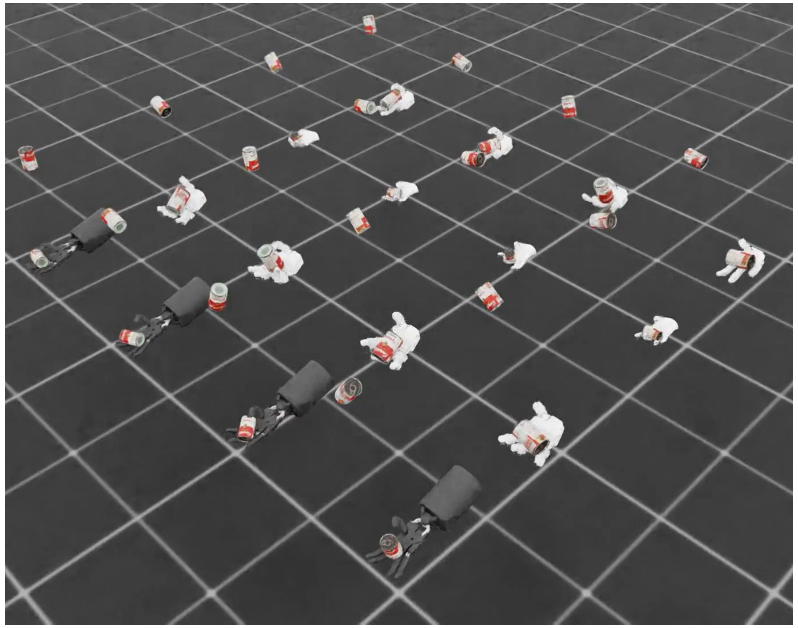
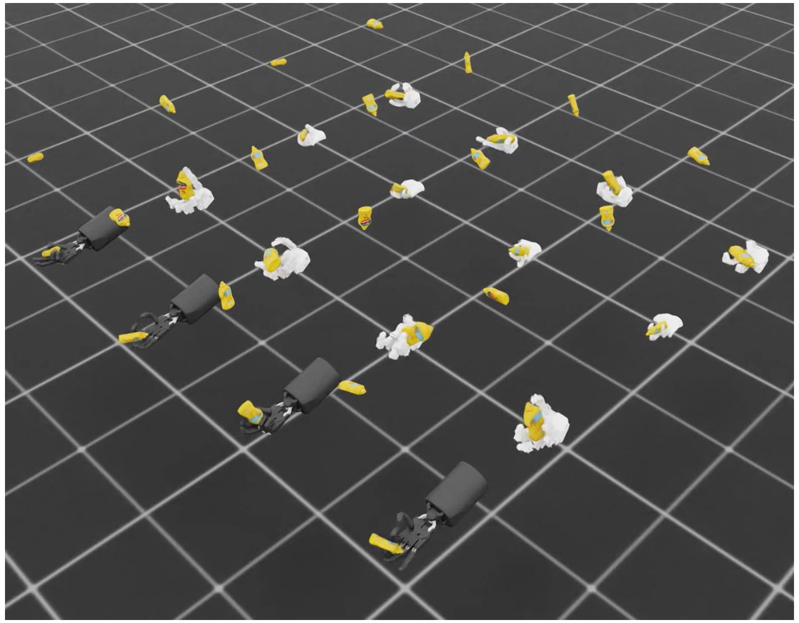
Leap (4 fingers)
7.7 ± 3.5



Shadow Hand (5 fingers)
4.4 ± 3.7

MANO (5 fingers)
8.9 ± 2.6

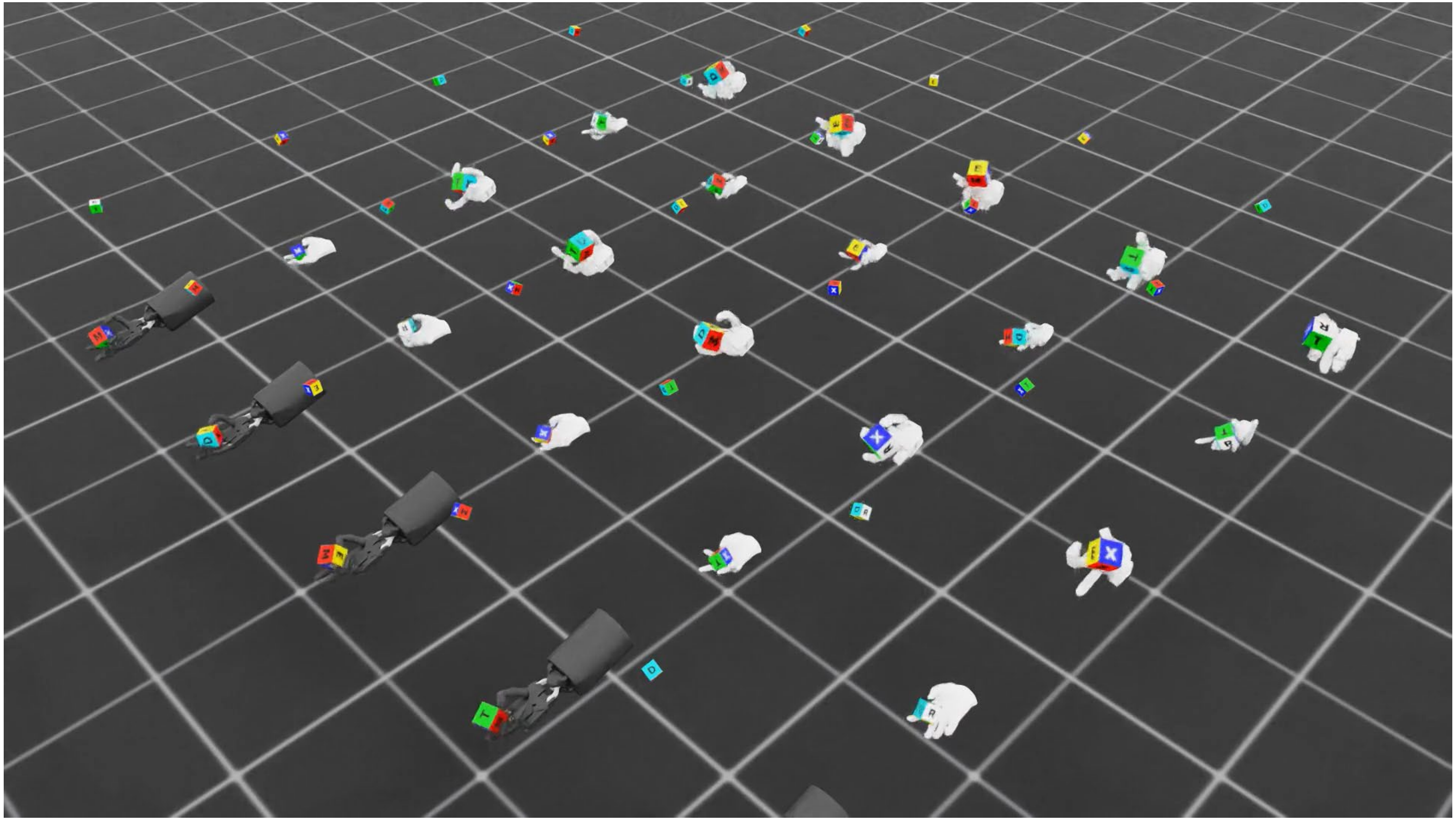
UHAS for In-hand Manipulation

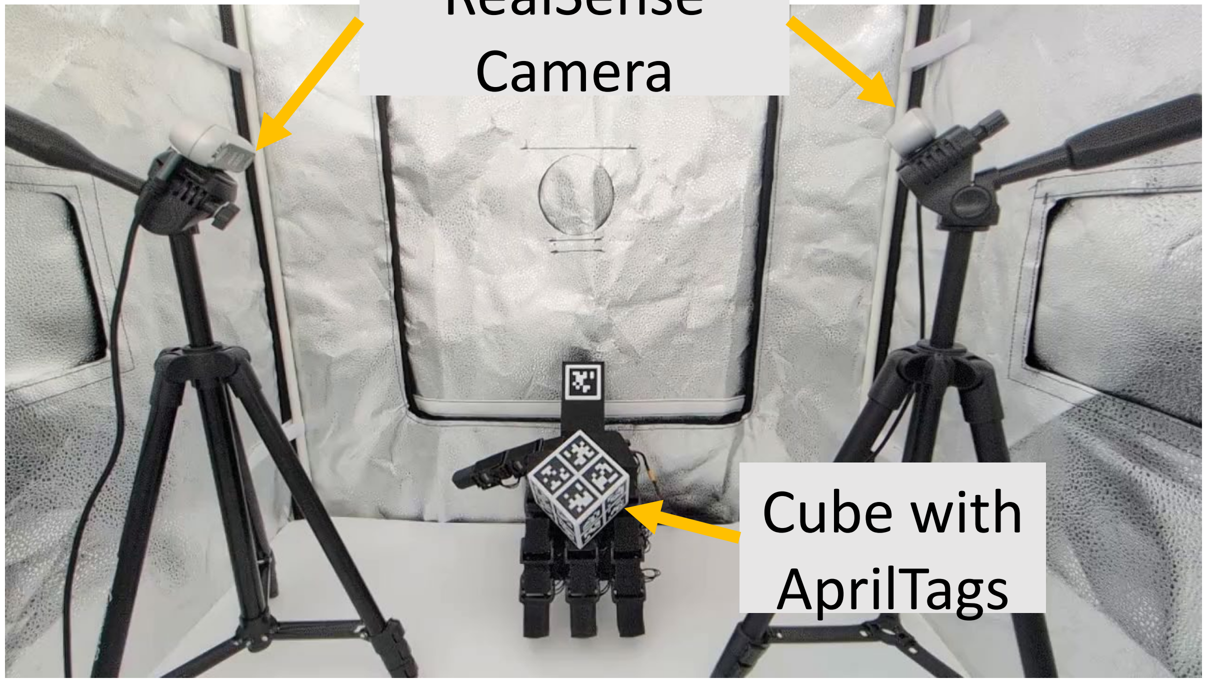
YCB Object	Test Hand	Single-Hand	Joint Control	Multi-Hand	Zero-shot
003	Allegro	98.5 / 9.2 ± 2.4	**99.2 / 9.5 ± 1.9**	97.4 / 8.7 ± 2.7	79.0 / 3.2 ± 3.0
Cracker	LEAP	96.8 / 8.3 ± 3.0	96.8 / 8.3 ± 3.0	**97.0 / 8.4 ± 3.0**	83.5 / 4.2 ± 3.5
Box	Shadow	**98.4 / 9.1 ± 2.2**	96.4 / 8.2 ± 3.1	95.0 / 7.6 ± 3.4	70.0 / 2.2 ± 2.5
	MANO	99.2 / 9.7 ± 1.3	**99.5 / 9.7 ± 1.5**	97.0 / 8.5 ± 2.9	88.3 / 5.3 ± 3.8
005	Allegro	99.8 / 9.9 ± 0.8	**99.9 / 9.9 ± 0.5**	99.8 / 9.8 ± 1.2	90.3 / 5.7 ± 3.9
Tomato	LEAP	**99.9 / 9.9 ± 0.8**	**99.9 / 9.9 ± 0.8**	99.7 / 9.7 ± 1.4	93.7 / 7.0 ± 3.7
Soup	Shadow	99.7 / 9.9 ± 1.0	**99.8 / 9.8 ± 1.0**	99.1 / 9.5 ± 1.8	79.4 / 3.5 ± 3.3
Can	MANO	**99.9 / 9.9 ± 0.9**	99.6 / 9.7 ± 1.5	99.7 / 9.8 ± 1.1	98.6 / 9.3 ± 2.2
006	Allegro	97.7 / 8.7 ± 3.0	**99.1 / 9.4 ± 2.3**	97.6 / 8.6 ± 3.0	67.3 / 1.9 ± 2.3
Mustard	LEAP	**97.7 / 8.7 ± 2.9**	96.7 / 8.2 ± 3.2	97.4 / 8.5 ± 3.0	70.0 / 2.2 ± 3.0
Bottle	Shadow	**97.9 / 8.9 ± 2.7**	96.1 / 8.0 ± 3.3	95.7 / 8.0 ± 3.2	67.3 / 2.0 ± 2.3
	MANO	98.5 / 9.3 ± 2.2	**98.8 / 9.4 ± 2.0**	97.1 / 8.6 ± 2.8	86.4 / 4.7 ± 3.6

Adding a New Hand

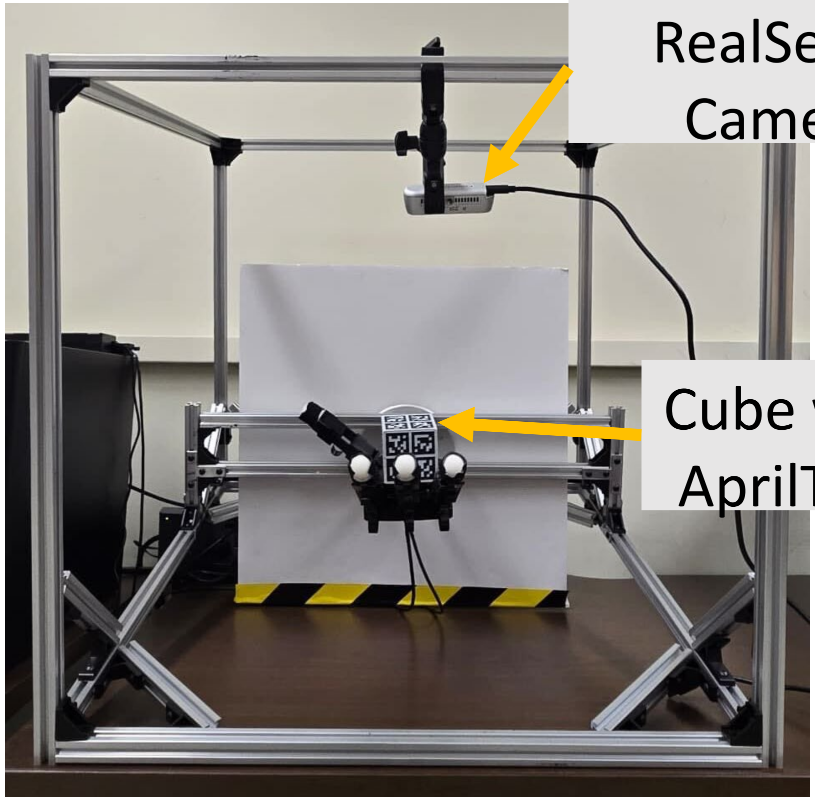
WUJI Hand (5 fingers)



Real-world in-hand manipulation

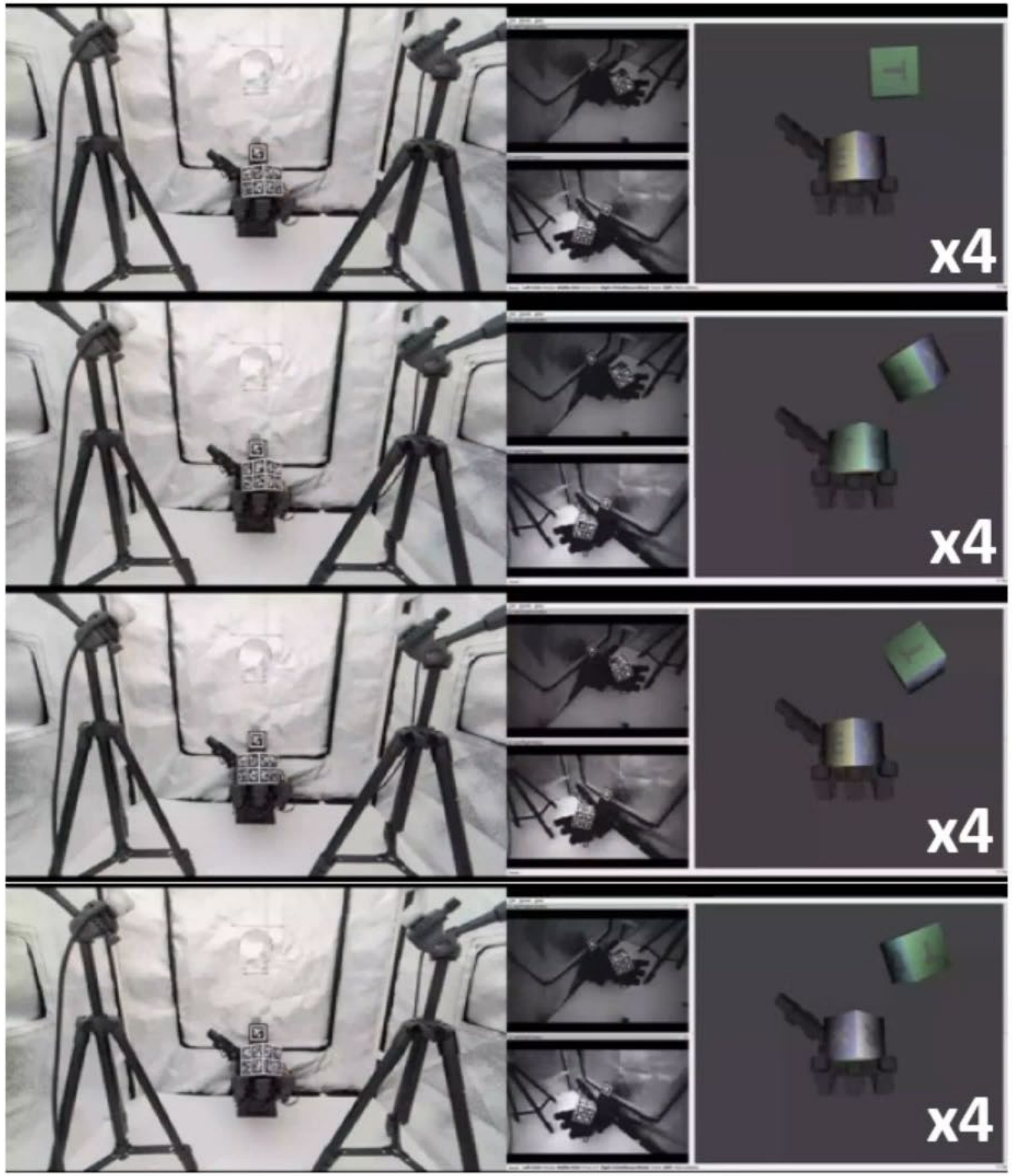


LEAP hand
(4-finger)



Allegro hand
(4-finger)

Real-world in-hand manipulation: LEAP hand



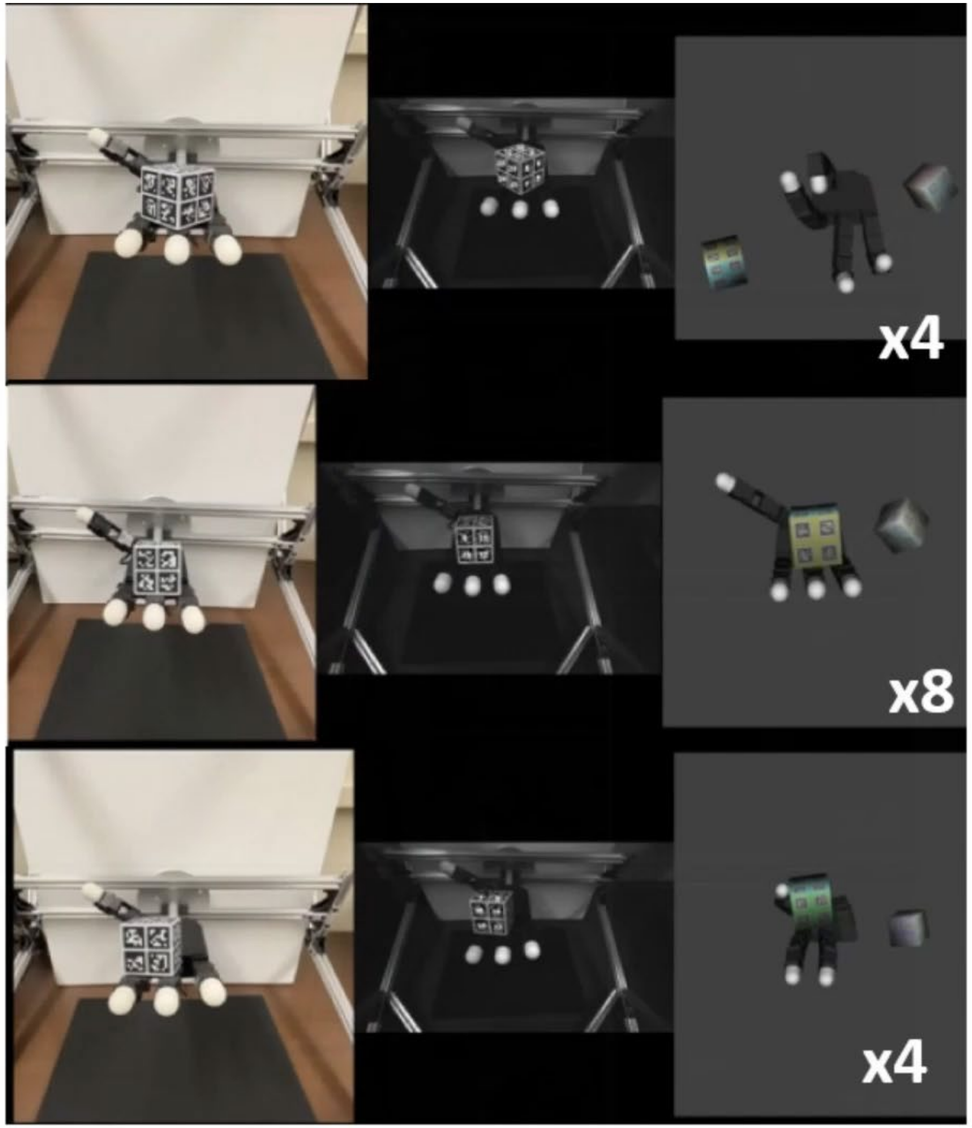
Baseline (Joint Control)	0.6 MEAN Reposes
UHAS (Zero-Shot)	0.9 MEAN Reposes
UHAS (Trained on Multi-Hand)	1.1 MEAN Reposes
UHAS (Trained on LEAP Hand)	2.0 MEAN Reposes

Real-world in-hand manipulation: Allegro hand

UHAS (Zero-Shot)

UHAS (Trained on Multi-Hand)

UHAS (Trained on Allegro Hand)



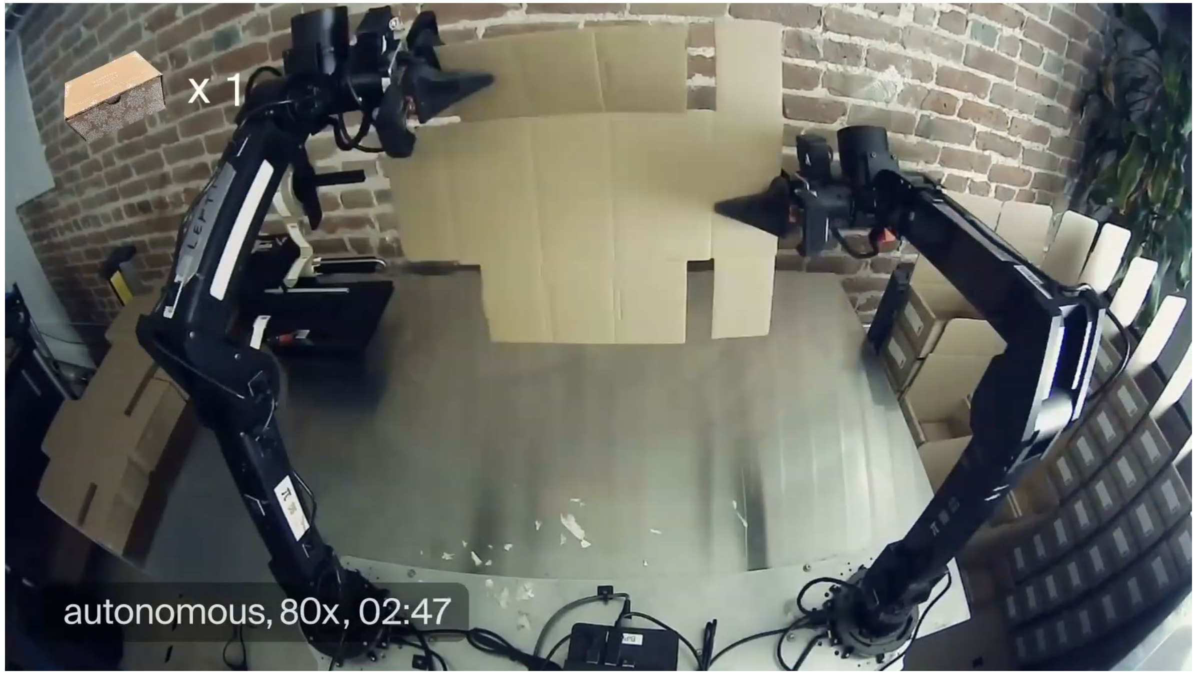
0.8 MEAN Reposes

2.1 MEAN Reposes

2.1 MEAN Reposes

Manipulation with Vision-Language-Action Models (VLAs)

Sunday Robotics
https://www.sunday.ai/blog/act-2-preview



Physical Intelligence
https://www.pi.website/blog/pistar06

How Good are these VLA Models?

- We need benchmarking

Jitendra MALIK
@JitendraMalikCV

We have seen some impressive robot manipulation policies recently. This is great, but for these results to be convincing (and practical) we should insist on **generalization** across 1. Object location (within some range) 2. Different instances of an object category 3. Background clutter. Authors should present experiments which demonstrate the range of variation which can be handled. Far too often the policy doesn't even generalize across all instances of an object category! Legged locomotion policies were convincing only when they worked across different terrain as in ashish-kmr.github.io/rma-legged-rob… We need to do the same for manipulanda.

VLA-Replica: A Low-Cost, Reproducible Benchmark for Real-World Evaluation of Vision-Language-Action Models

Alex S. Huang* Jiahui Zhang* Shiqing Tang Yu Xiang

* Equal Contribution Intelligent Robotics and Vision Lab at the University of Texas at Dallas

https://irvlutd.github.io/VLAReplica/

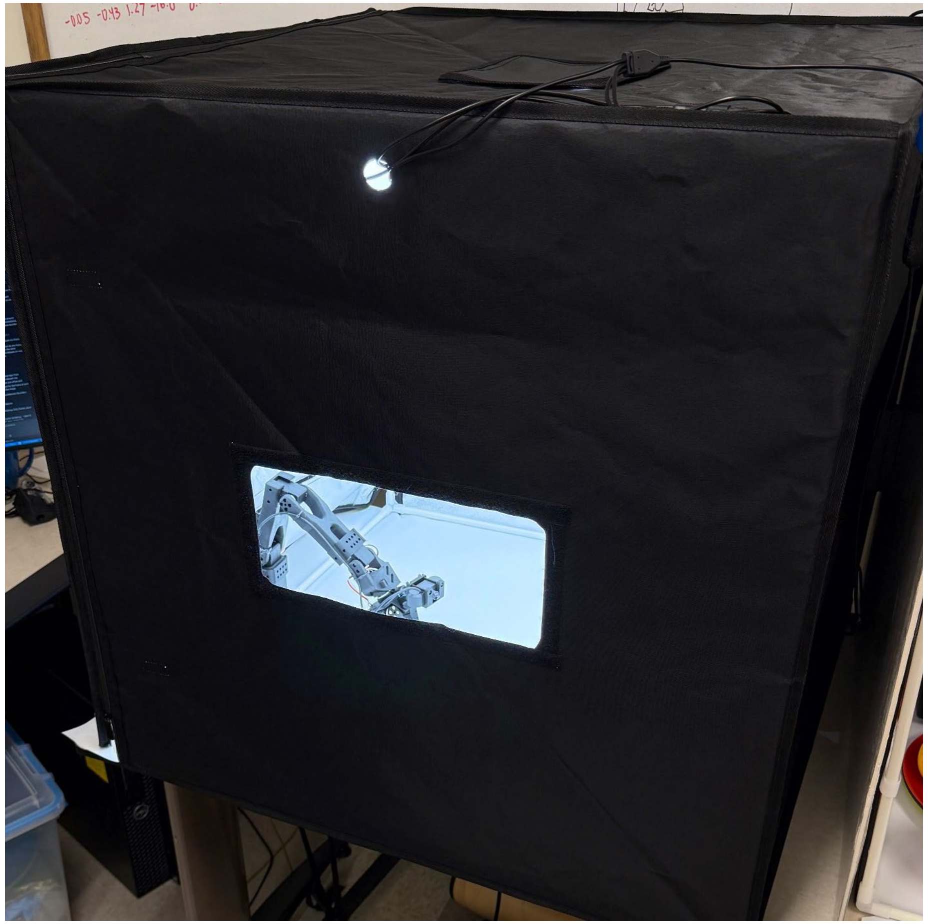
In NeurIPS Track on Datasets and Benchmarks, 2026.

VLA-Replica Benchmark

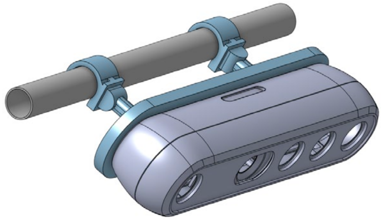
Low-cost Components

SO-101 Arm ~$200

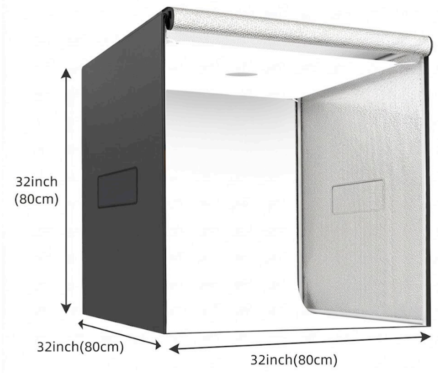
Webcam $13.98



Real Set-up



D455 Camera ~$425



Light Box $152.99

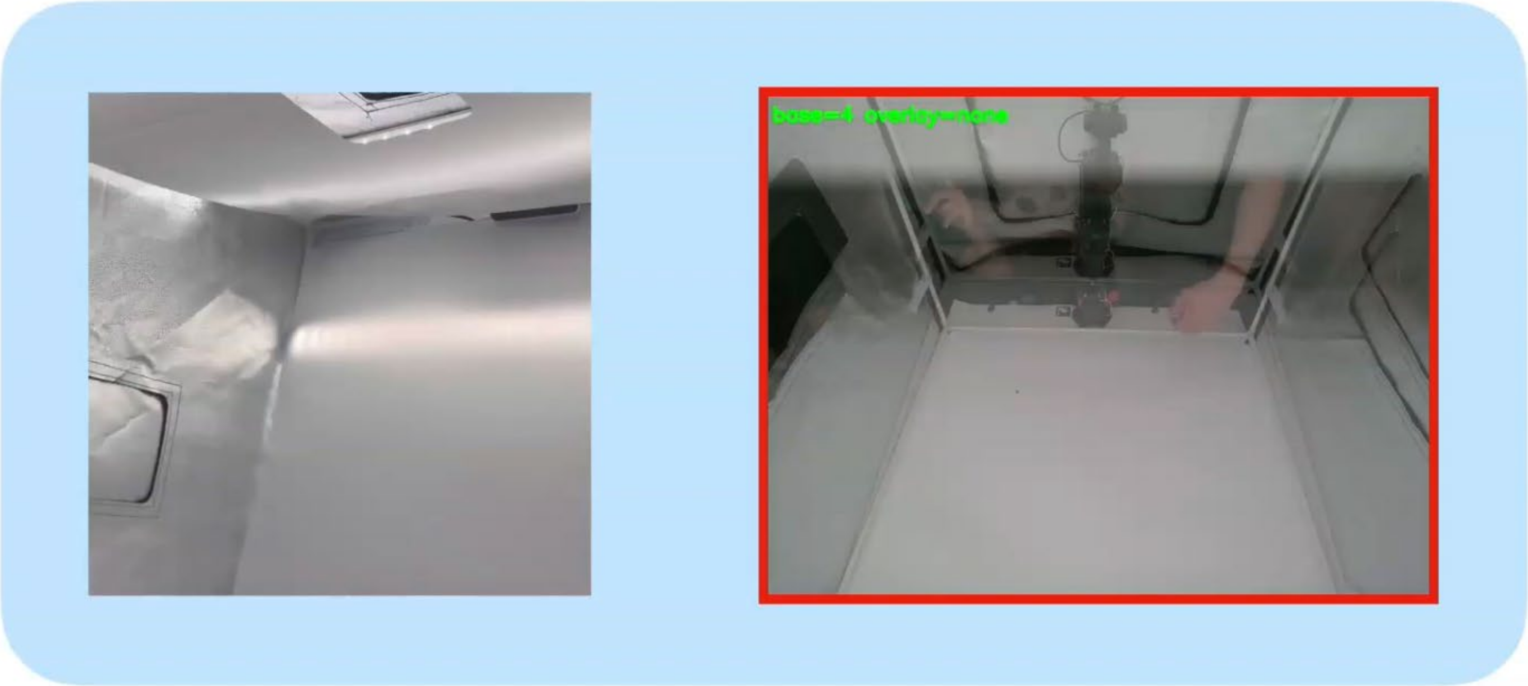
VLA-Replica Benchmark

VLA-Replica Benchmark

Camera Calibration

2X

Using reference images for camera calibration



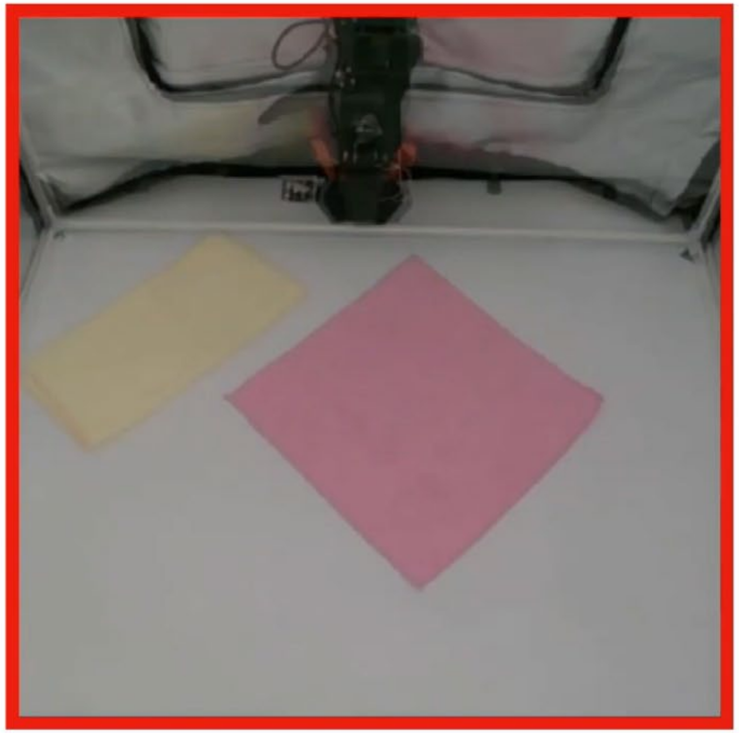
Top Camera Calibration

VLA-Replica Benchmark

2X

Task Evaluation

Task 1: Fold the pink towel in half.



Reference Placement

Align placement with reference frame

VLA-Replica Benchmark: Tasks

We provide:

- 500 demos across 10 tasks
- 10 In-distribution tasks
- 8 Out-of-distribution tasks
- 5 reference frames per task

VLA-Replica Benchmark: Tasks

Task #	Task Type	Training & ID Eval Task Example	OOD Eval Task Example
1	Pick-and-Place	Put bread in **red**/**blue** plate	Put bread on **yellow** plate
2		Put **red** bowl on **purple** coaster	Put **green** bowl on **yellow** coaster
3		Stack **blue** cube on **red** cube	Stack **green** cube on **green** cube
4		Put all **2 or 3** blocks in **blue** box	Put all **3 or 4** blocks in **pink** box
5	Object Interaction	Fold **pink**/**yellow** towel in half	Fold **blue** towel in half
6		Open oven	N/A
7		Clean whiteboard with eraser	N/A
8	Memory	Pour pepper **1,2 or 3** times to **red** plate	Pour pepper **4 or 5** times to **blue** plate
9		Lift **green**/**red**/**blue** bowl **1 or 3** times	Lift **yellow** bowl **2, 4 or 5** times
10		Press button **1 or 3** times	Press button **2, 4 or 5** times

VLA-Replica Benchmark: Leaderboard

VLA-Replica-ID   VLA-Replica-OOD

3 runs eval with std

Rank	Method	Average	Put bread on plate	Put bowl on coaster	Stack block on block	Put all blocks into box	Fold towel	Open oven	Erase whiteboard	Shake pepper times
1	$\pi_{0.5}$ [6]	**0.53 ± 0.05**	0.80 ± 0.20	0.87 ± 0.12	0.33 ± 0.12	0.47 ± 0.12	1.00 ± 0.00	0.53 ± 0.12	0.33 ± 0.12	0.33 ± 0.
2	GR00T N1.7 [8]	**0.42 ± 0.06**	0.80 ± 0.00	0.87 ± 0.12	0.00 ± 0.00	0.29 ± 0.08	1.00 ± 0.00	0.47 ± 0.31	0.40 ± 0.20	0.20 ± 0.
3	MolmoAct2-SO100_101 [7]	**0.39 ± 0.06**	0.93 ± 0.12	0.73 ± 0.12	0.27 ± 0.12	0.27 ± 0.12	0.73 ± 0.12	0.13 ± 0.23	0.53 ± 0.12	0.13 ± 0.
4	π_0 [5]	**0.32 ± 0.03**	0.67 ± 0.12	0.67 ± 0.31	0.00 ± 0.00	0.00 ± 0.00	0.67 ± 0.23	0.20 ± 0.00	0.33 ± 0.12	0.27 ± 0.
5	SmolVLA [3]	**0.29 ± 0.04**	0.80 ± 0.20	0.20 ± 0.20	0.27 ± 0.12	0.00 ± 0.00	0.73 ± 0.12	0.33 ± 0.12	0.20 ± 0.00	0.00 ± 0.
6	ACT [1]	**0.21 ± 0.08**	0.33 ± 0.12	0.07 ± 0.12	0.00 ± 0.00	0.00 ± 0.00	0.60 ± 0.20	0.33 ± 0.12	0.20 ± 0.00	0.27 ± 0.
7	X-VLA [4]	**0.13 ± 0.02**	0.27 ± 0.12	0.13 ± 0.12	0.00 ± 0.00	0.00 ± 0.00	0.53 ± 0.12	0.07 ± 0.12	0.00 ± 0.00	0.20 ± 0.
8	DiT-D [2]	**0.11 ± 0.05**	0.20 ± 0.20	0.13 ± 0.12	0.00 ± 0.00	0.07 ± 0.12	0.20 ± 0.00	0.20 ± 0.35	0.20 ± 0.00	0.07 ± 0.
9	DiT-F [2]	**0.07 ± 0.05**	0.13 ± 0.23	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.27 ± 0.12	0.13 ± 0.23	0.13 ± 0.12	0.00 ± 0.

https://irvlutd.github.io/VLAReplica/

VLA-Replica Benchmark: Leaderboard

VLA-Replica-ID | **VLA-Replica-OOD**

3 runs eval with std

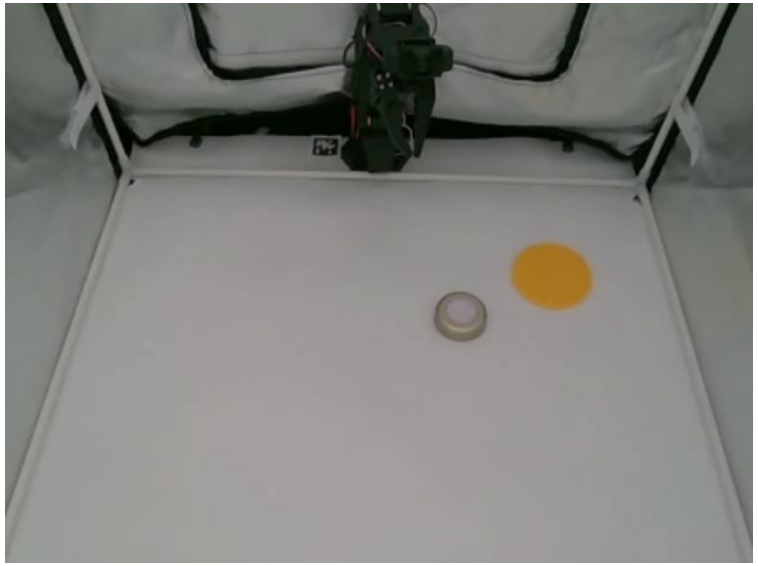
Rank	Method	Average	Put bread on plate	Put bowl on coaster	Stack block on block	Put all blocks into box	Fold towel	Shake pepper n times	Lift bowl n times	Press button times
1	MolmoAct2-SO100_101 [7]	**0.47 ± 0.05**	0.87 ± 0.12	0.87 ± 0.12	0.60 ± 0.00	0.20 ± 0.20	0.87 ± 0.12	0.13 ± 0.12	0.13 ± 0.12	0.07 ± 0.
2	$\pi_{0.5}$ [6]	**0.39 ± 0.10**	0.93 ± 0.12	0.60 ± 0.20	0.20 ± 0.20	0.27 ± 0.12	0.80 ± 0.00	0.27 ± 0.23	0.00 ± 0.00	0.07 ± 0.
3	GR00T N1.7 [8]	**0.35 ± 0.00**	0.93 ± 0.12	0.67 ± 0.12	0.00 ± 0.00	0.13 ± 0.12	1.00 ± 0.00	0.00 ± 0.00	0.07 ± 0.12	0.00 ± 0.
4	π_0 [5]	**0.30 ± 0.02**	0.73 ± 0.12	0.67 ± 0.12	0.20 ± 0.20	0.00 ± 0.00	0.53 ± 0.06	0.22 ± 0.03	0.00 ± 0.00	0.07 ± 0.
5	SmolVLA [3]	**0.27 ± 0.04**	0.73 ± 0.12	0.33 ± 0.12	0.20 ± 0.20	0.13 ± 0.12	0.47 ± 0.12	0.00 ± 0.00	0.20 ± 0.00	0.07 ± 0.
6	ACT [1]	**0.08 ± 0.01**	0.40 ± 0.20	0.27 ± 0.12	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.
7	DiT-D [2]	**0.08 ± 0.04**	0.07 ± 0.12	0.33 ± 0.23	0.00 ± 0.00	0.00 ± 0.00	0.07 ± 0.12	0.07 ± 0.12	0.00 ± 0.00	0.07 ± 0.
8	X-VLA [4]	**0.05 ± 0.04**	0.33 ± 0.31	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.07 ± 0.12	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.
9	DiT-F [2]	**0.04 ± 0.01**	0.13 ± 0.12	0.07 ± 0.12	0.00 ± 0.00	0.07 ± 0.12	0.07 ± 0.12	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.

https://irvlutd.github.io/VLAReplica/

VLA-Replica Benchmark: Pi0.5 Behaviors

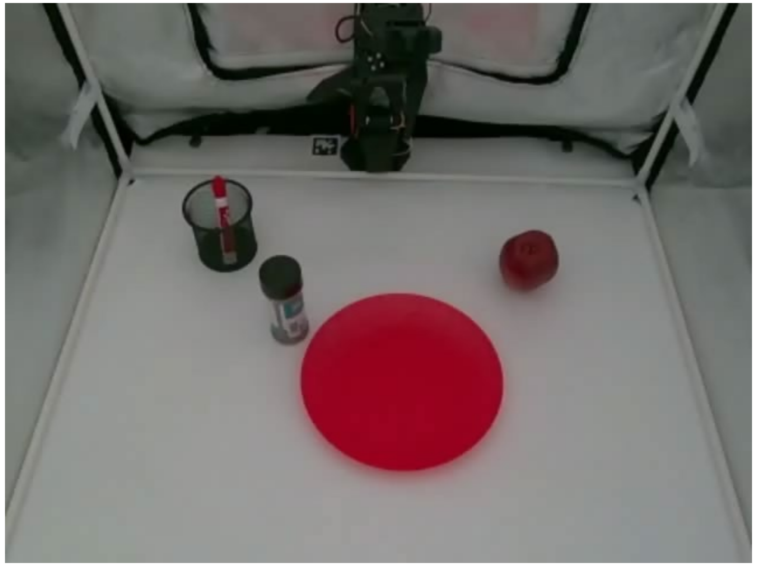
Stack the **yellow** block on the **blue** block

Precise manipulation is challenging



Press the button **three** times

Memory task is challenging



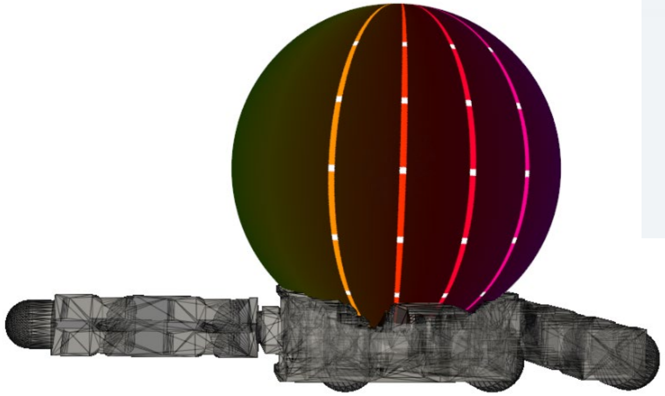
Pour **1** shake of pepper into the **red** plate

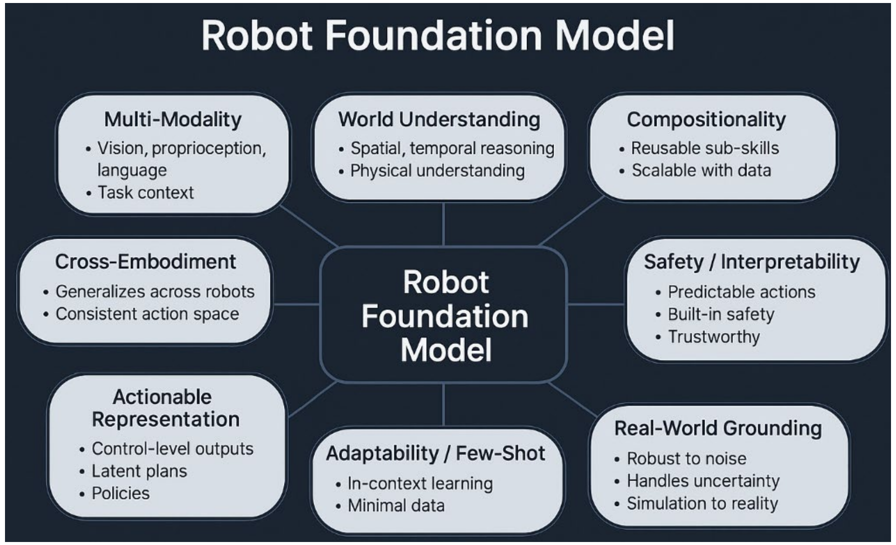
Failure recovery

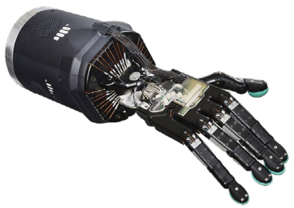
Summary

Use data from all robots and human for learning

Unified Hand Action Space

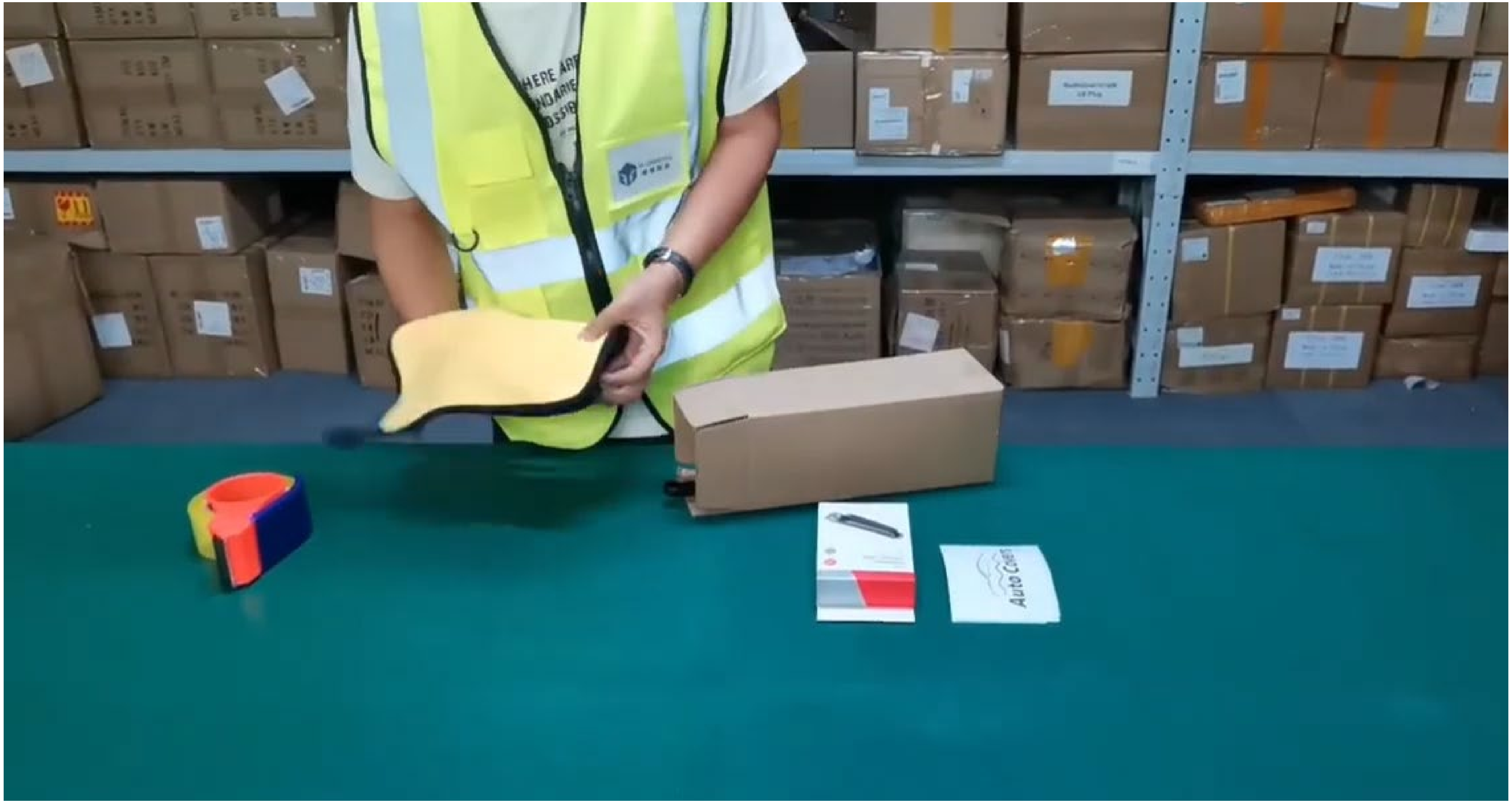






Real-World Evaluation and Benchmarking

Robot Manipulation is still an Open Challenge



IRVL
INTELLIGENT ROBOTICS AND VISION LAB
UT DALLAS

NSF DARPA SONY NVIDIA XPENG FRONTIER MINING

https://labs.utdallas.edu/irvl/

Assisted by
Mr. Garvin R. Solomon

Thank you!