Modern Vision-Language-Action Models

Andrew Szot

ML Ph.D. (co-advised with

Ram Ramrakhya

CS Ph.D. (co-advised with Dhruv Batra)

Chengyue Huang

ML Ph.D.

Some materials by Liconti & Katzschmann, Kim, Claude

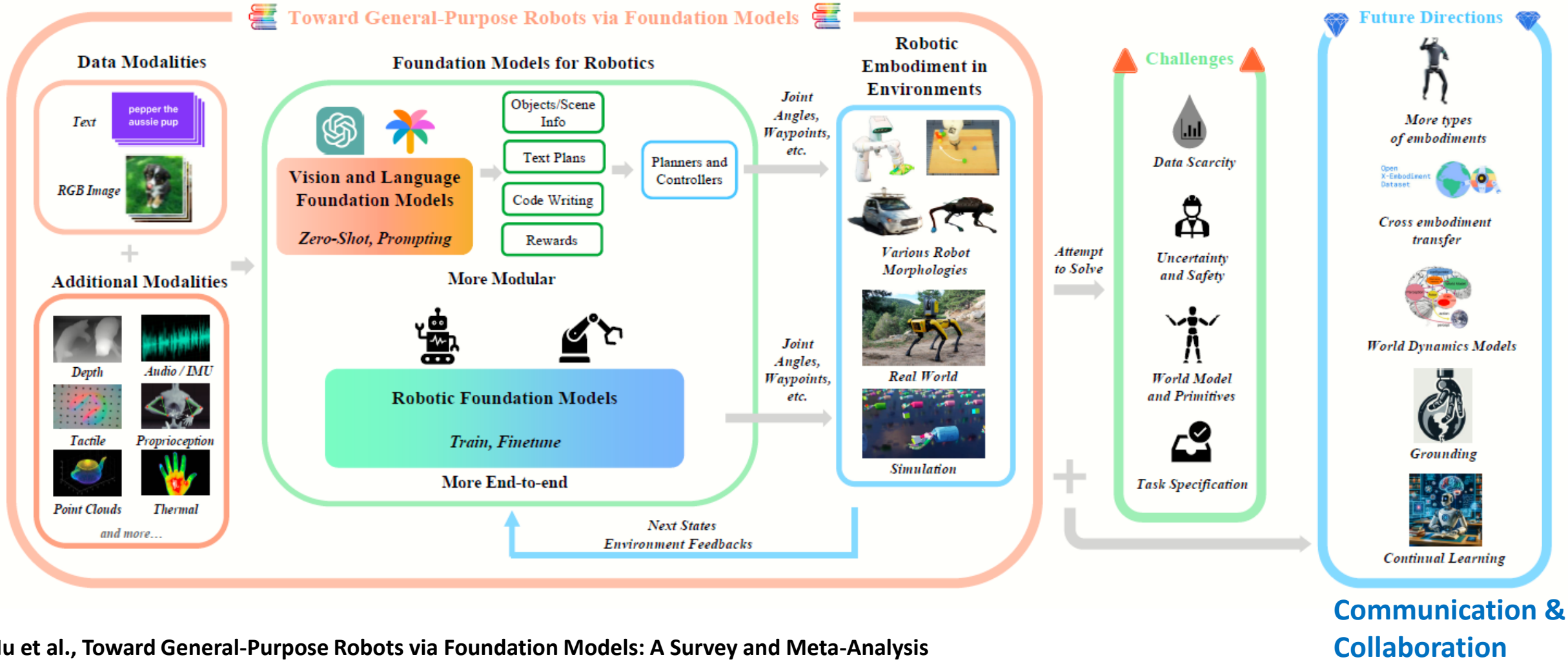
Administrative

- Projects! Due 05/04
- CIOS is open! https://b.gatech.edu/cios
  - Please fill out and provide comments!

Lecture Outline

- From Language Models to Action Models**: LLMs, VLMs, VLAs**
- The Robotics Transformer Lineage**: RT-1, RT-2, RT-X, OpenVLA**
- Action Representations**: Discrete bins, continuous flow, FAST tokens**
- Frontier VLA Systems**: Pi0, Pi0.5, Pi0.7, GR00T, Gemini Robotics**
- Our Work**: GEA and cooperative embodied agents**
- Open Challenges & Future Directions

Robotics & Foundation Models

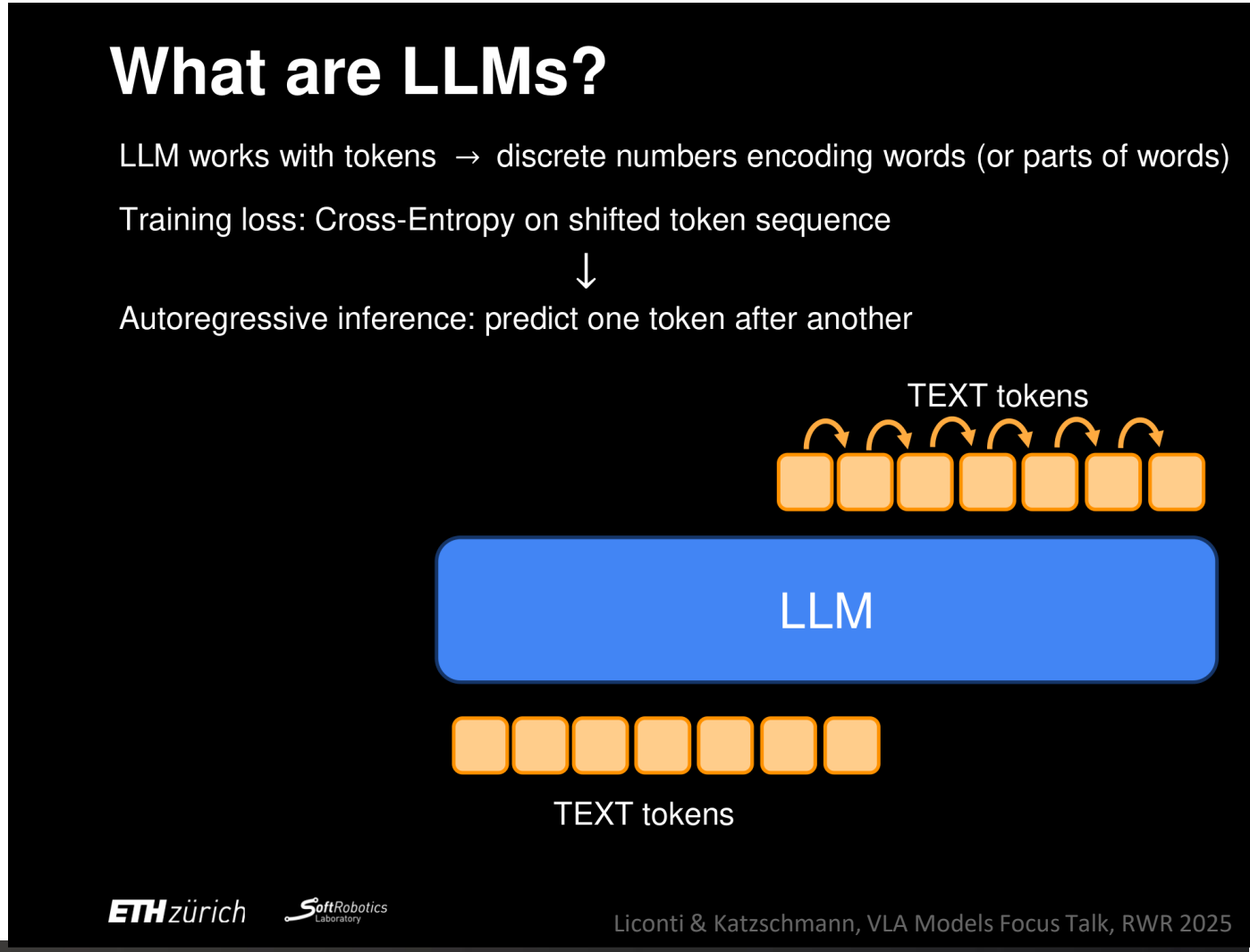


Hu et al., Toward General-Purpose Robots via Foundation Models: A Survey and Meta-Analysis

From Language Models to Action Models

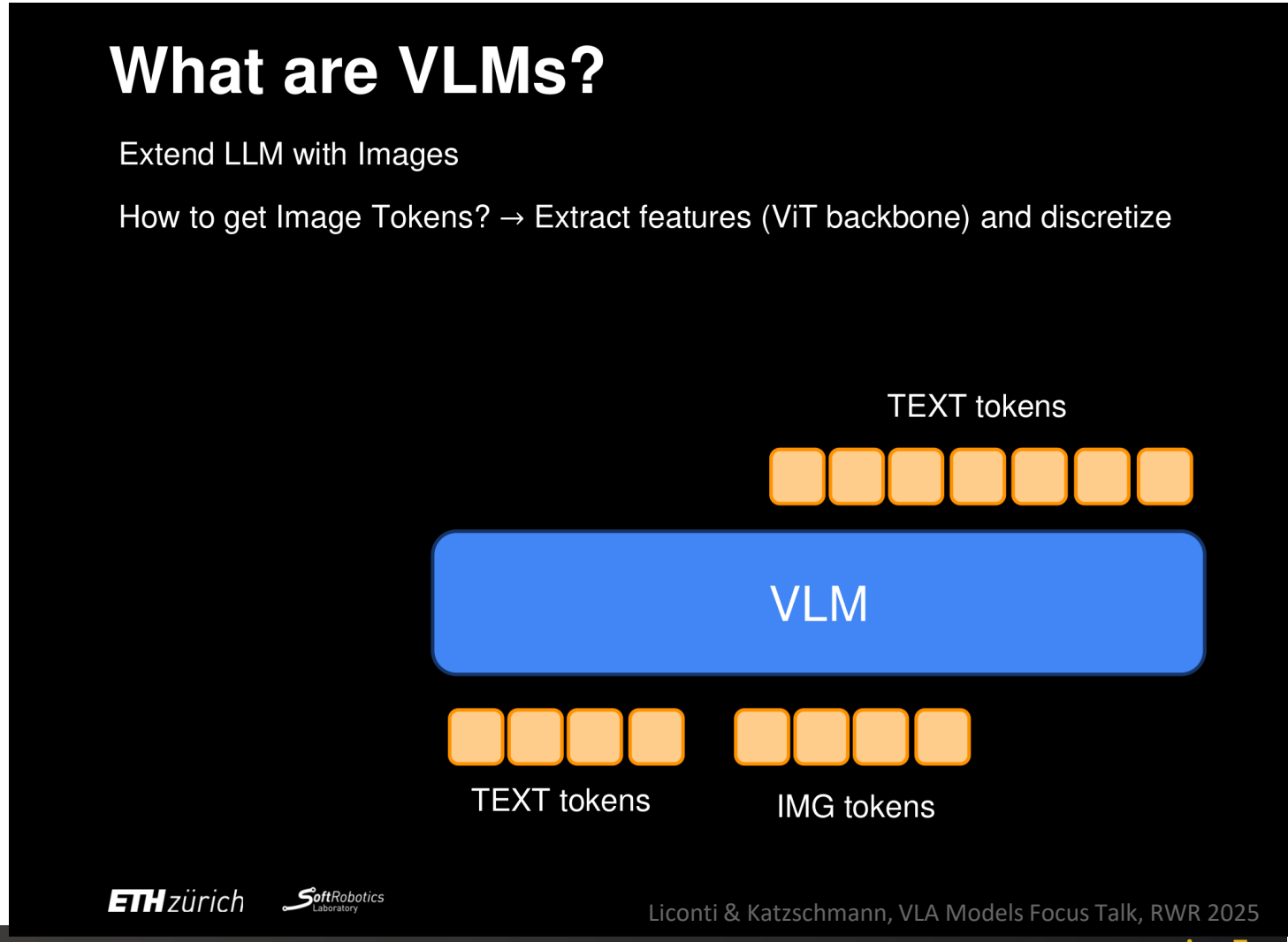
What are LLMs?

- LLMs work with **tokens**: discrete numbers encoding words
- Training: next-token prediction via cross-entropy loss
- Autoregressive generation: predict one token at a time
- Scale + data + compute = emergent capabilities



What are VLMs?

- Extend LLMs with **image tokens**
- Extract visual features via ViT backbone
- Discretize and project into LLM token space
- Train on image-text pairs for visual understanding
- Examples: CLIP, LLaVA, PaLI, All modern frontier models



What can these help with in robotics?

- **There are lots of challenges in robotics**
  - Sense -> Plan -> Act
  - Large action/search space
  - Long-horizon execution
  - Physics
  - …

- **One key difficulty in robotics in the past:**
  - How do we encode common-sense reasoning?
  - Can we make the tasks *language-conditioned (ala NLP)*
  - Previously: Per-task engineered models, knowledge graphs, etc.

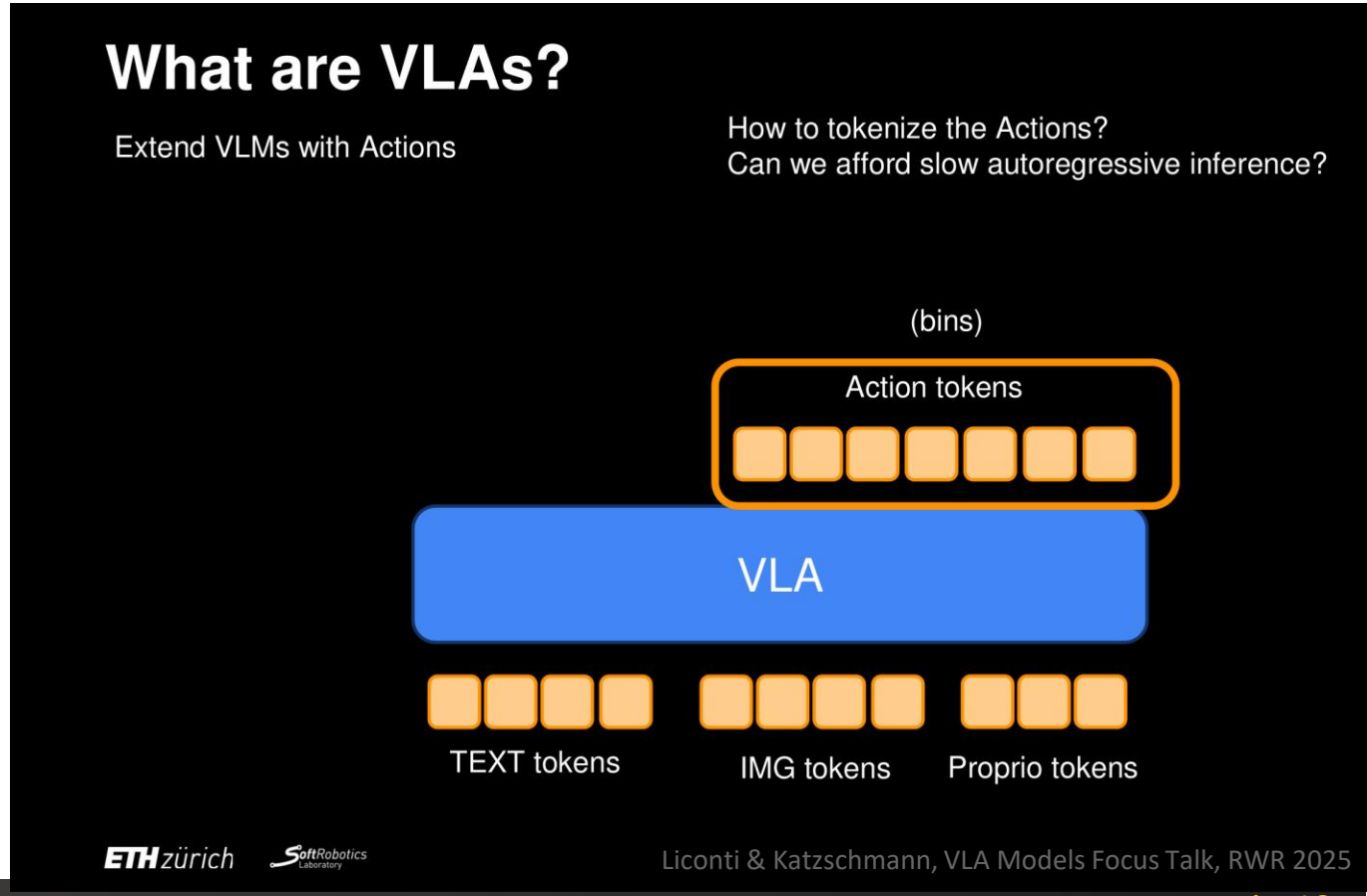
Early Attempts: SayCan

- **Early attempts use LLMs to plan**
  - Output sub-tasks given natural language task
  - Or output code (code-as-policies)

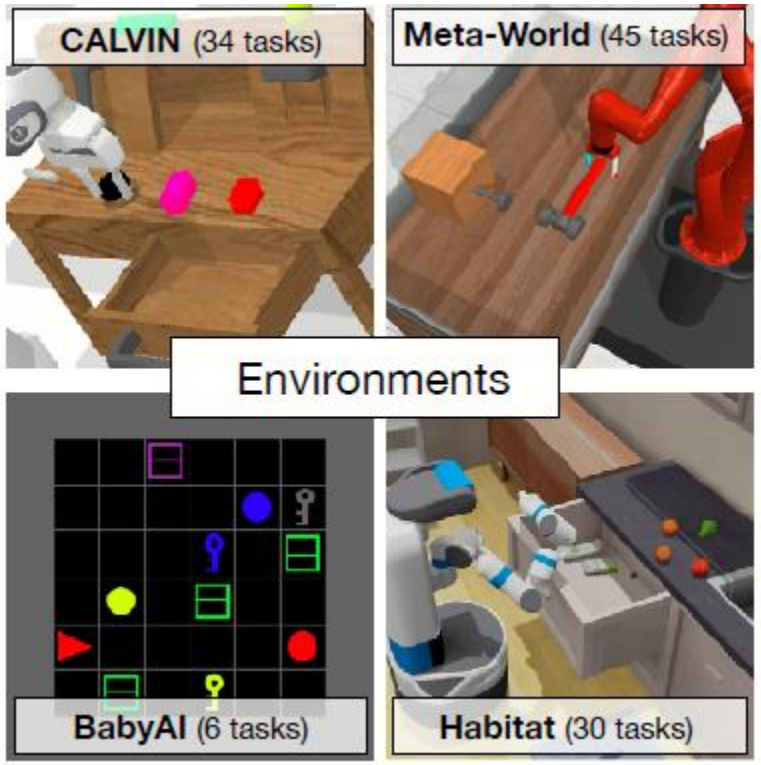
Do As I Can, Not As I Say:
Grounding Language in Robotic Affordances

What are VLAs?

- Extend VLMs with **action tokens**
- Key design questions:
  - How to tokenize continuous robot actions?
  - Can we afford slow autoregressive inference?
  - How to handle proprioceptive state?
- Input: images + language + proprioception
- Output: action sequence (joint angles, end-effector poses)

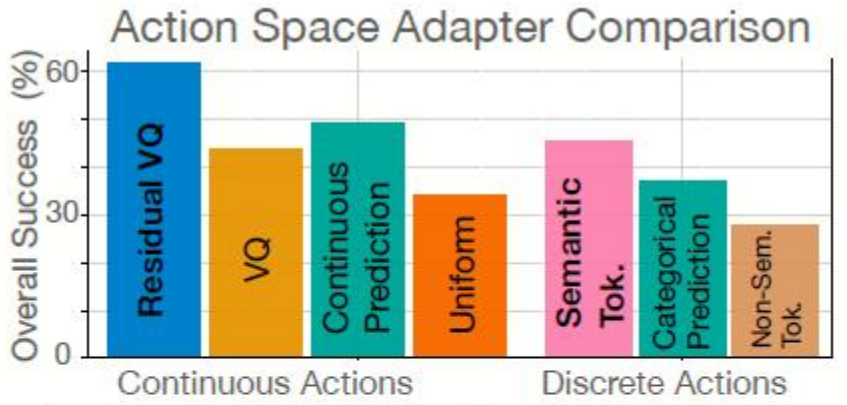


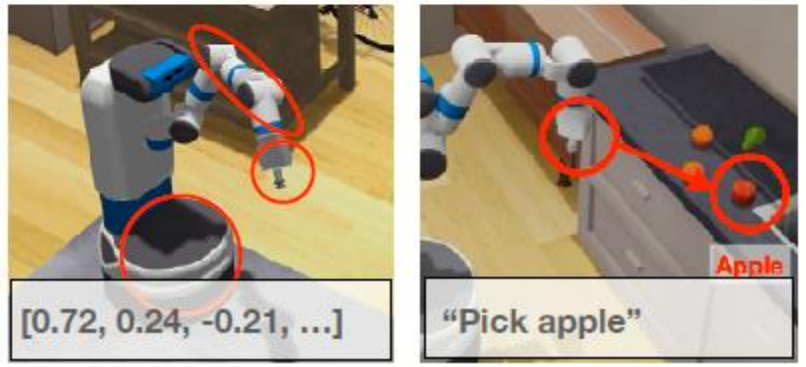
GEA: Our Generalist Vision-Language Action Model



Multimodal Large Language Model

Action Space Adapter





Lots of great concurrent work! OpenVLA, LLARVA, PI0, etc.
Ours: 1) Action Tokenization (NeurIPS), 2) SFT->RL, 3) Multi-task generalist

Szot et al., Grounding Multimodal Large Language Models in Actions, NeurIPS 2024
Szot et al., From Multimodal LLMs to Generalist Embodied Agents: Methods and Lessons, CVPR 2025

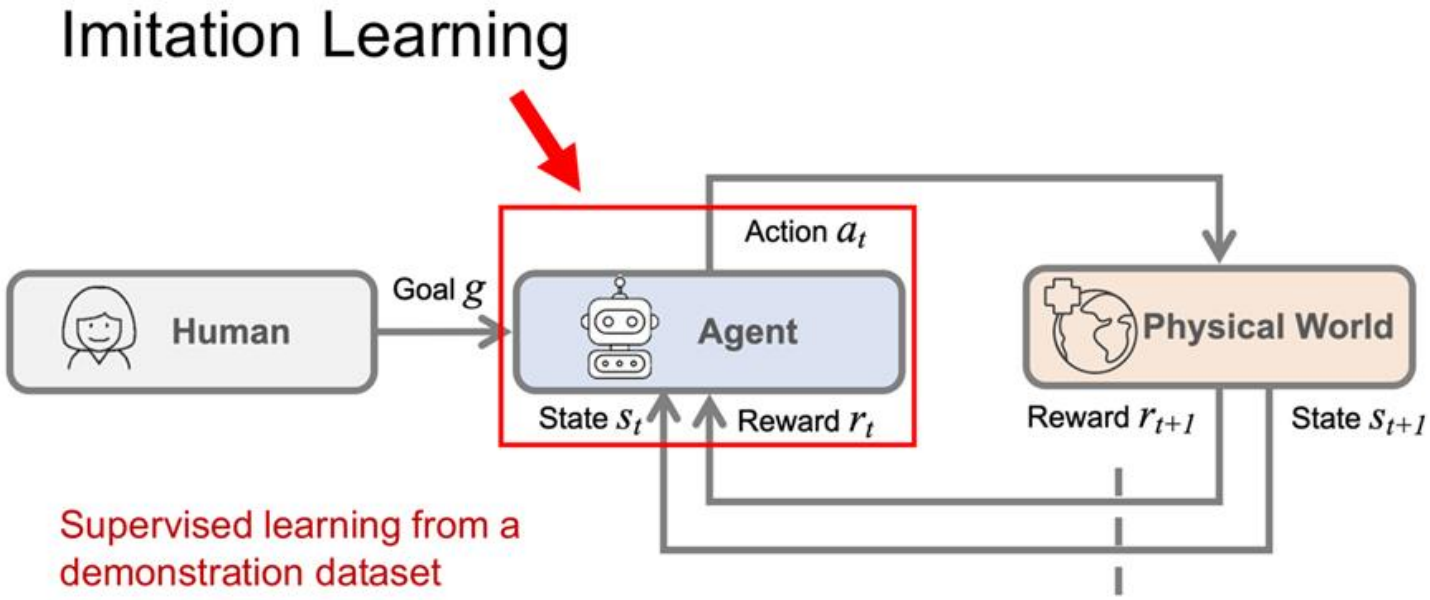
Andrew Szot
ML Ph.D. (co-advised with Dhruv Batra)

Ways of Applying to Robotics

Data is a key issue in robotics!

Some options

- Simulation vs. Real
  - Enables oracle planning, etc.
- RL vs. Imitation Learning
- Learn from Videos



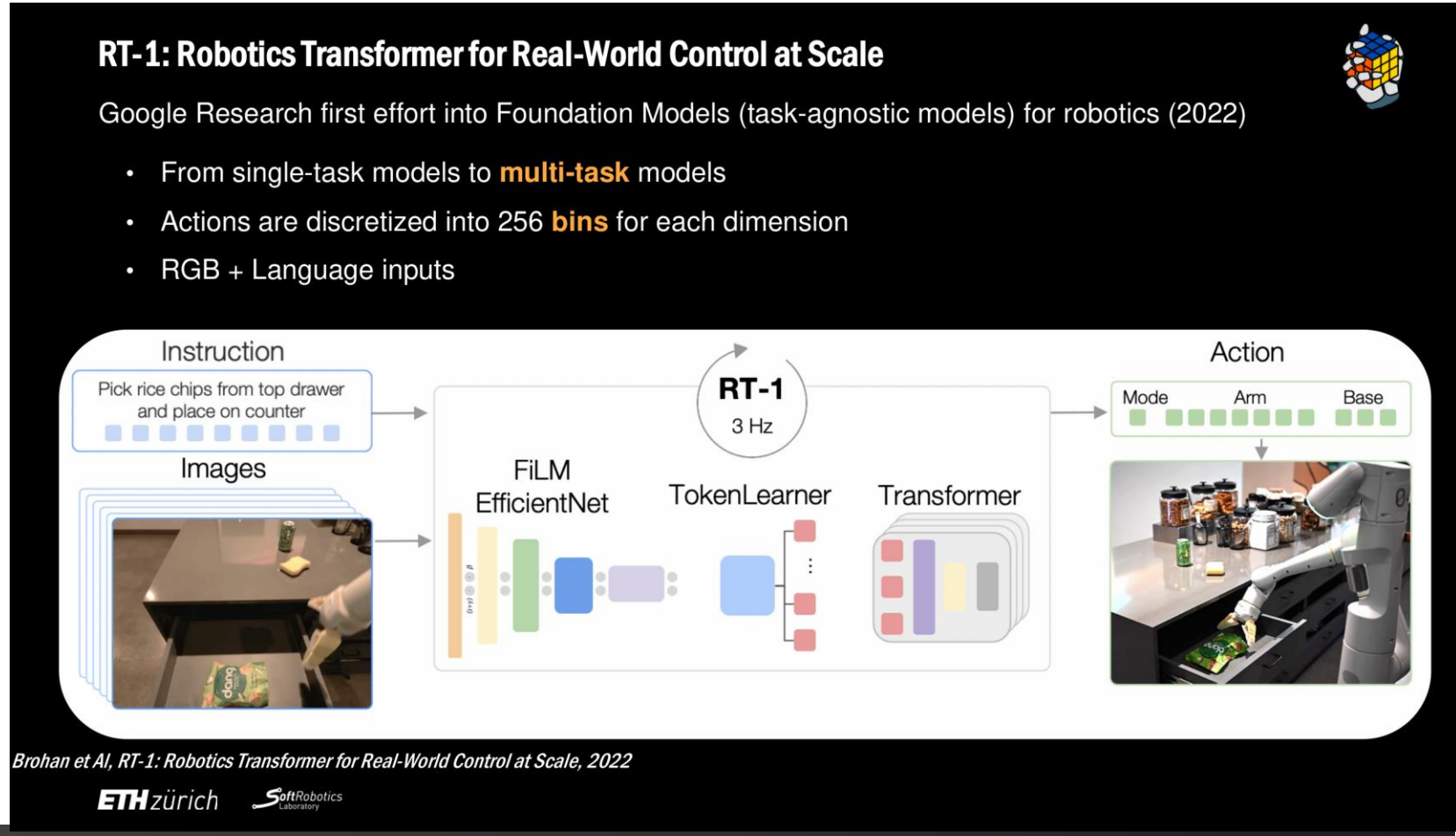
Behavior Cloning (BC)

- Learn policy from expert demos
- Simple but suffers from distribution shift
- What can't successful cases show?

Li, Robot Learning, Stanford 2025

RT-1: Robotics Transformer at Scale

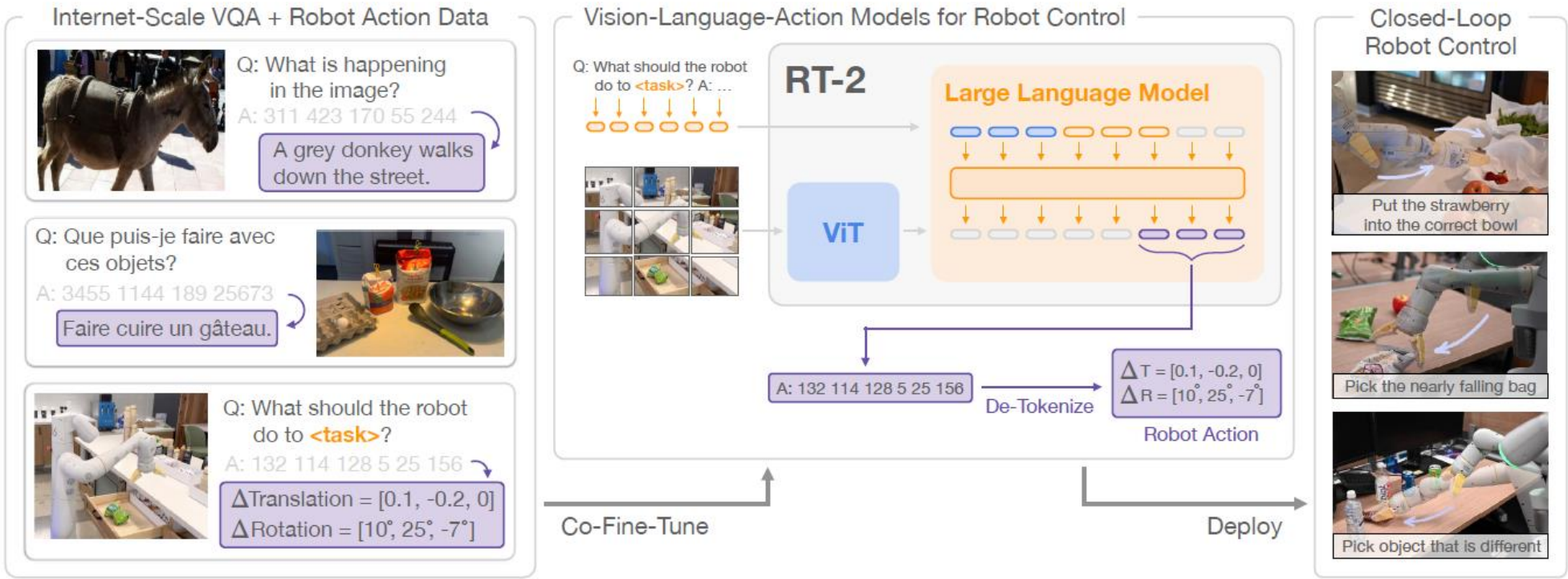
- First large-scale multi-task robotic transformer
- Actions discretized into **256 bins** per dimension
- Trained on 130K+ real-world demonstrations
- 7 skill categories across 700+ tasks
- Showed that scale + diversity improves generalization
- Key limitation: no pretrained vision-language backbone



Brohan et al., RT-1: Robotics Transformer for Real-World Control at Scale

RT-2: VLM Backbone -> VLA

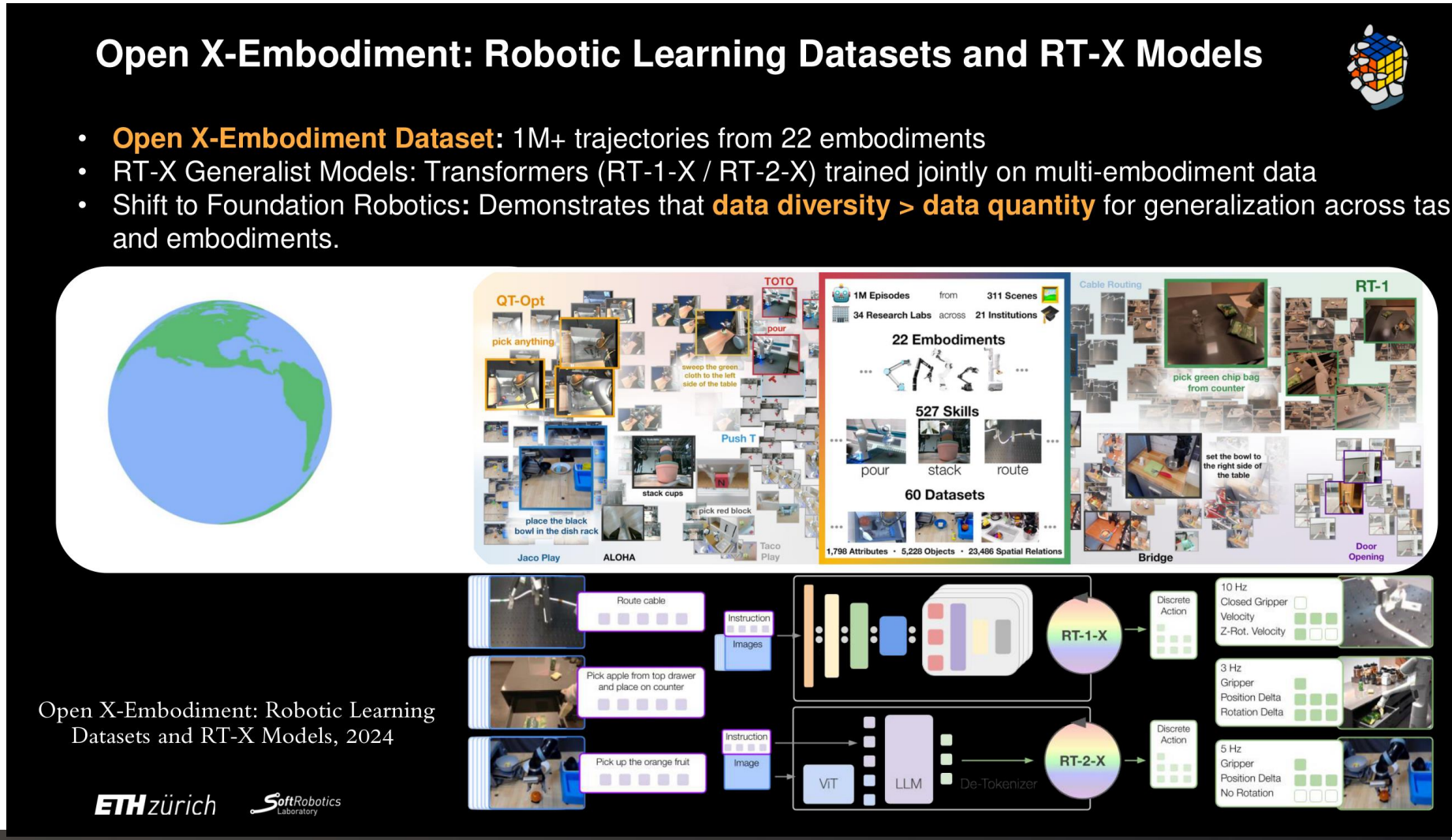
- Built on pretrained VLM (PaLI-X / PaLM-E)
- Key insight**: actions as text tokens in VLM vocabulary**
- Co-fine-tuned on web vision-language + robot data
- Web knowledge transfers to robotic control
- Emergent capabilities: novel object understanding
- 55B parameters (much larger than RT-1)



Brohan et al., RT-2: Vision-Language-Action Models Transfer Web Knowledge, 2023

Open X-Embodiment & RT-X

- Open X-Embodiment Dataset: **1M+ trajectories**
- 22 different robot embodiments
- RT-X: generalist model trained on full dataset
- Demonstrated positive cross-embodiment transfer
- Established community benchmark for generalist policies
- Foundation for subsequent VLA work



Open X-Embodiment: Robotic Learning Datasets and RT-X Models, 2024

Open X-Embodiment Collaboration, 2023

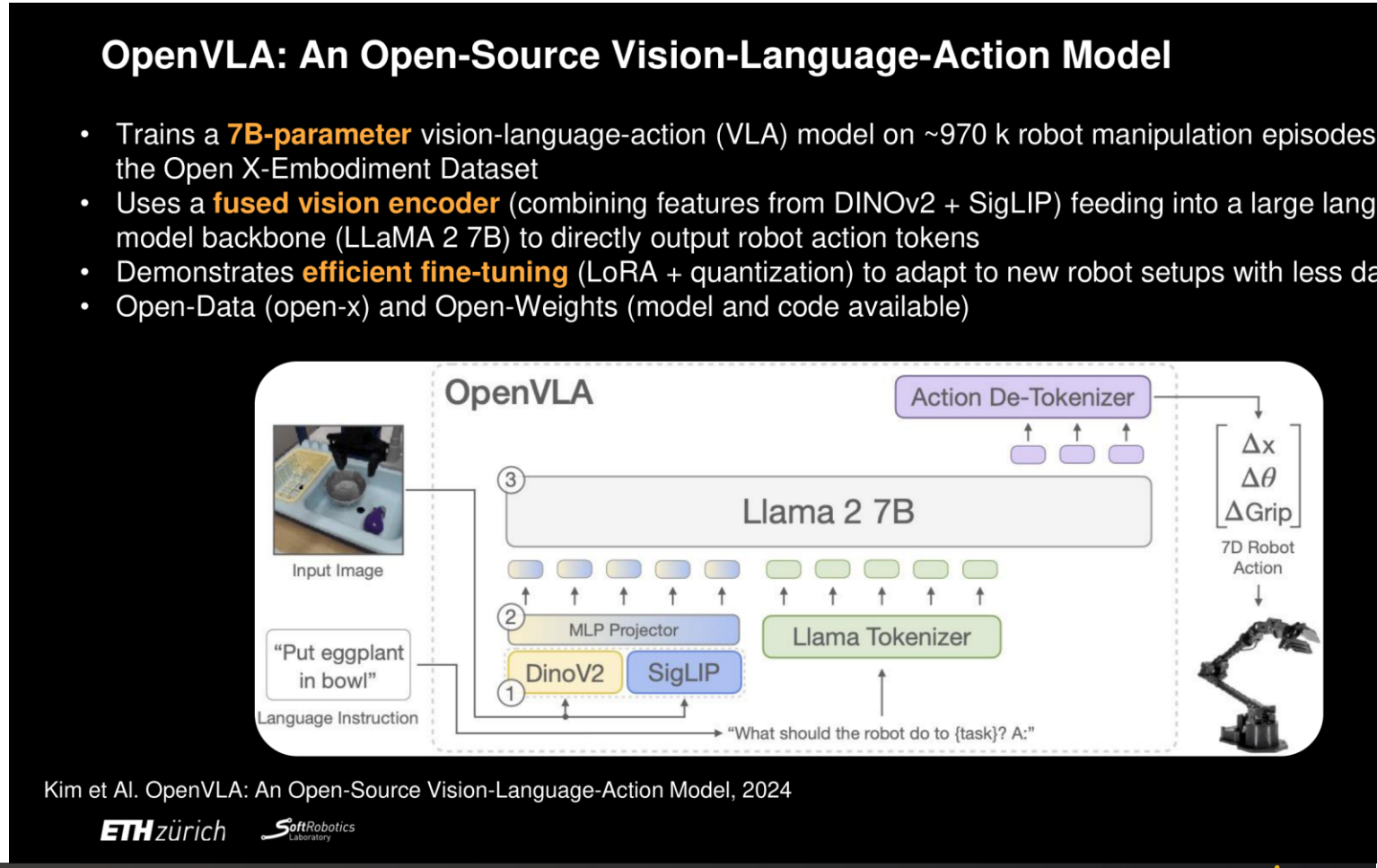
OpenVLA: Open-Source 7B VLA

- 7B-parameter **open-source VLA model**
- Trained on ~970K robot manipulation episodes
- Built on Prismatic VLM backbone
- Actions as discrete bins (like RT-2)
- Strong baseline but limited by:

Autoregressive action discretization

No action chunking

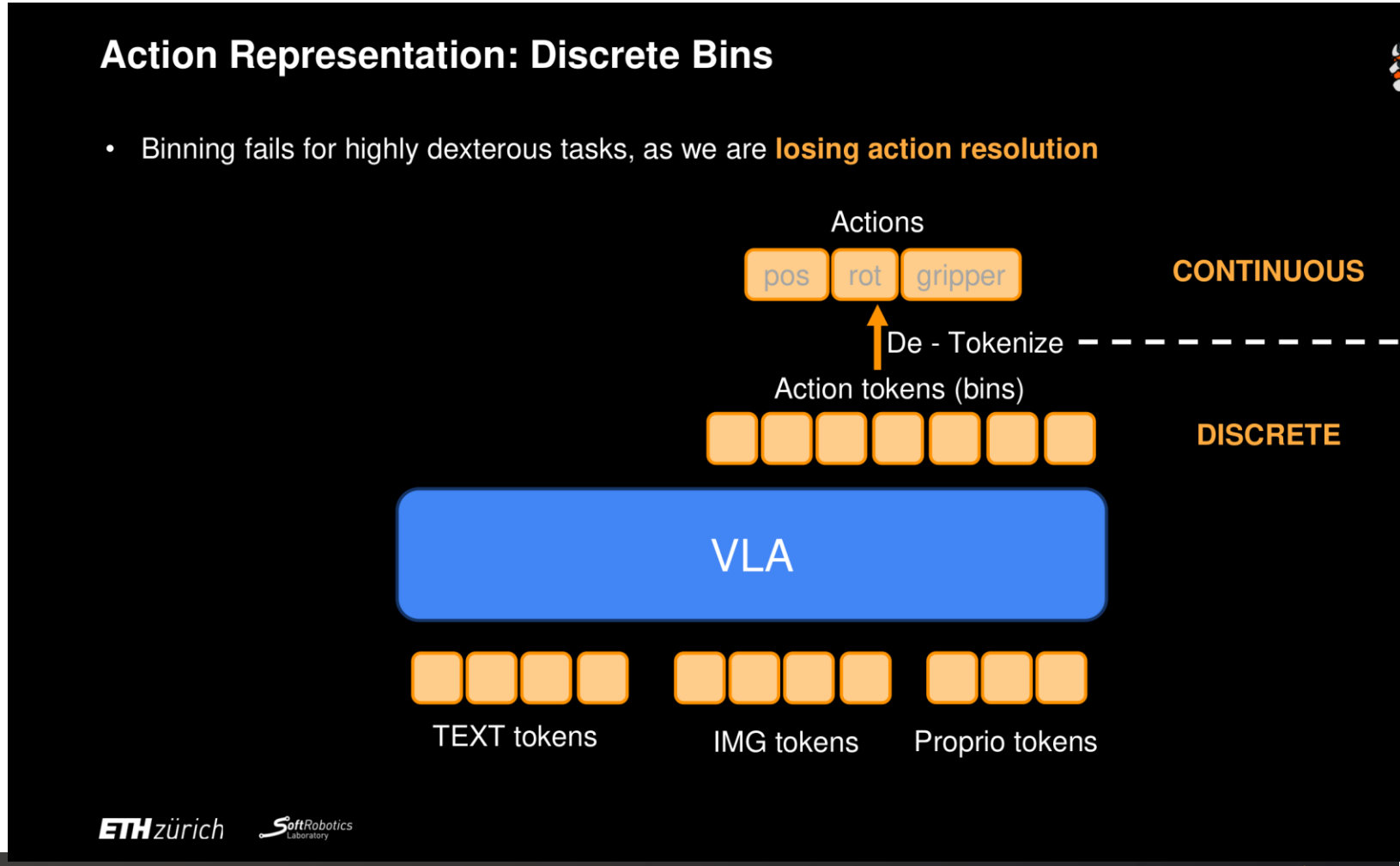
Lower dexterity on precision tasks



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

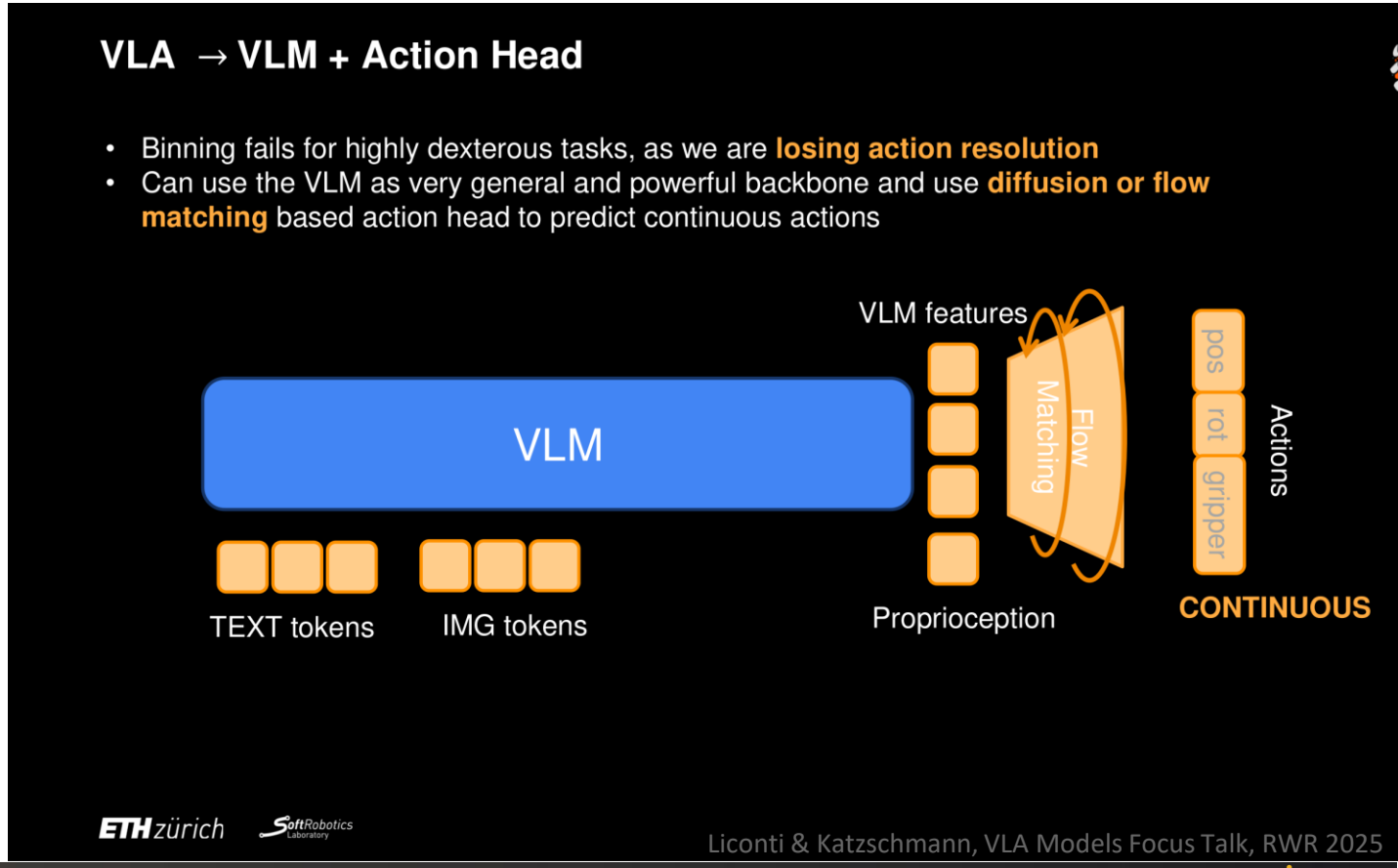
Action Representation: Discrete Bins

- Discretize each action dimension into N bins (e.g., 256)
- Treat actions as language tokens
- Advantages: reuse LLM training infrastructure
- Disadvantages:
  - Loses action resolution for dexterous tasks
  - Slow autoregressive decoding
  - Cannot easily predict action chunks



VLA = VLM + Action Head

- Alternative: use VLM as a **feature extractor**
- Add a dedicated action head for continuous output
- Advantages:
  - Full action resolution preserved
  - Faster parallel action decoding
  - Natural action chunking support
- The VLM provides semantic understanding
- The action head provides motor control

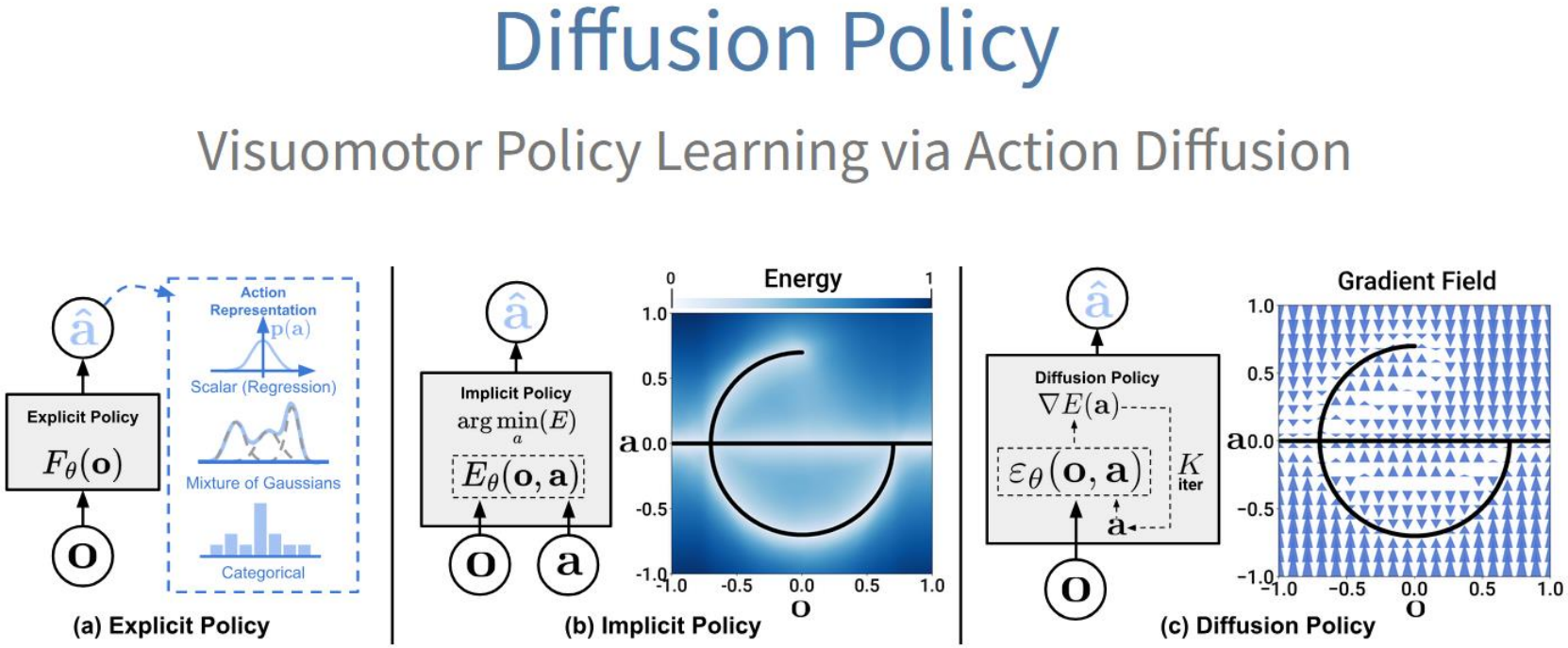


Action Head: Diffusion Policies

- Learn action trajectory
- $a_{t:t+H} \sim p\theta(\cdot|o_t)$

- Typically output many steps into future, execute 1/less number of steps

- Start from noised version, progressively denoise conditioned on observation



Pi0: Vision-Language-Action Flow Model

- 3.3B parameters **(PaliGemma 3B + 300M action expert)**
- Flow matching instead of autoregressive tokens
- Action chunking: predict H=50 future actions at once
- Blockwise causal attention mask:

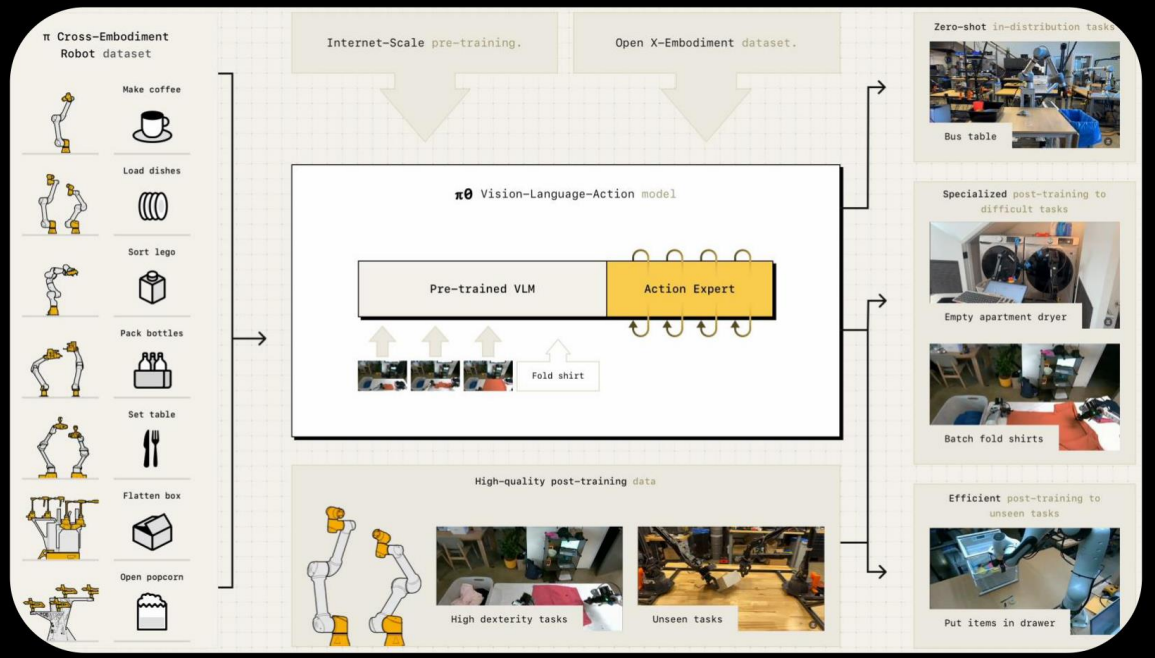
VLM inputs | Robot state | Noisy action tokens

- Cross-embodiment: 7 robot configs, 68 tasks
- 10,000+ hours of dexterous manipulation data

Black et al., Pi0: A Vision-Language-Action Flow Model, 2024

$\pi 0$

- Trained across **multiple robots and tasks (8 robots in-house + OpenX data)**
- VLM pre-trained on web-scale image+text data (Paligemma), and then augmenting it w **action output** capability (via flow-matching) so it can output motor commands at up to



physicalintelligence.company

ETH zürich SoftRobotics Laboratory

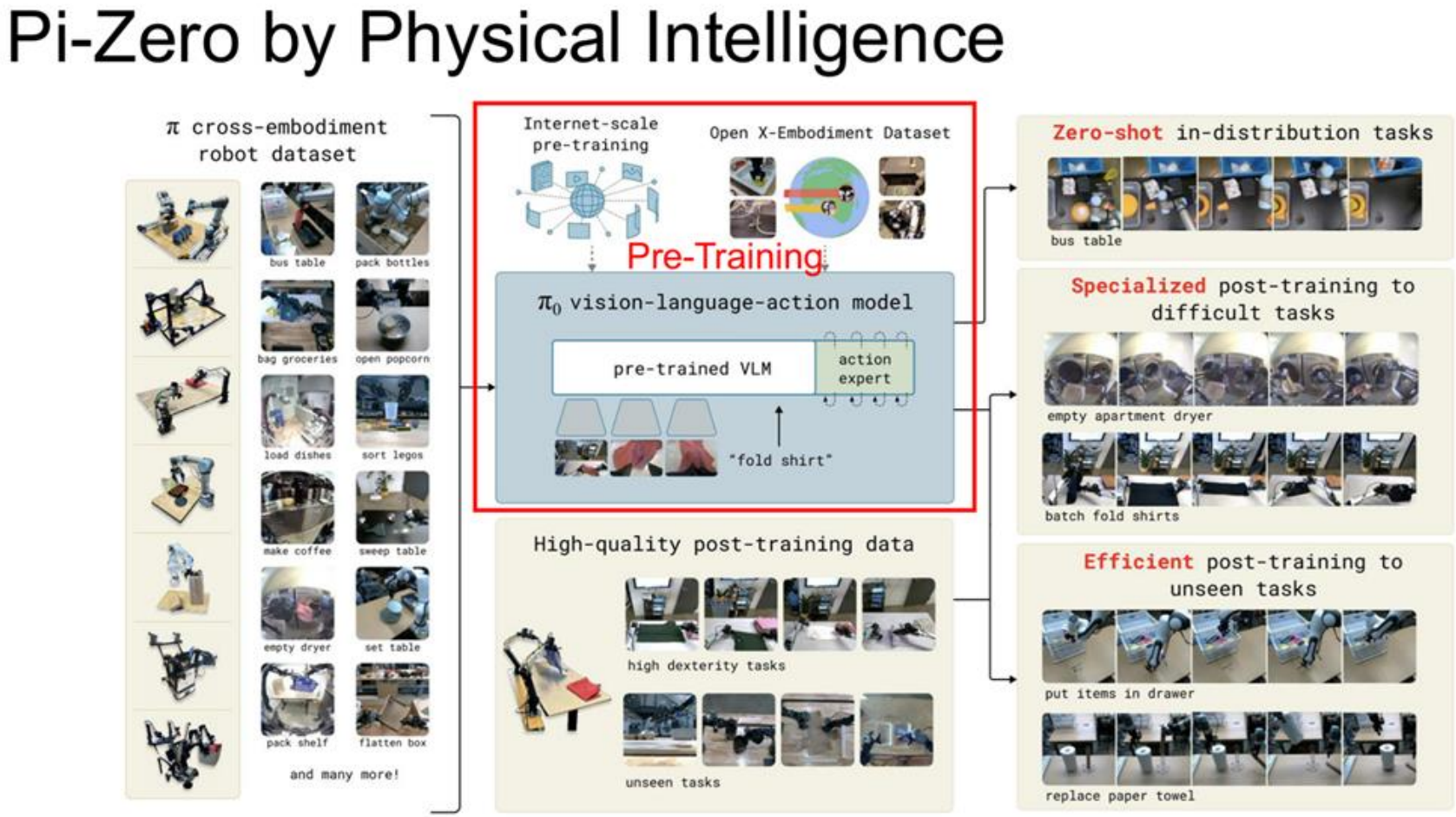
Pi0: Flow Matching for Continuous Actions

Why flow matching?

- Robot actions are continuous, not discrete tokens
- Flow matching learns a denoising ve field
- Noise -> clean action chunk via 10 Eu steps
- ~73ms inference on RTX 4090

Training:

- Sample noise, interpolate
- Predict vector field toward demonstrated actions
- Shifted beta distribution emphasizes noisy timesteps



Black et al., Pi0: A Vision-Language-Action Flow Model

Pi0: Pretraining + Post-Training Recipe

Pretraining

- Broad cross-embodiment data
- 903M timesteps (PI data)
- + OXE, Bridge v2, DROID
- Teaches coverage & recovery

Post-Training

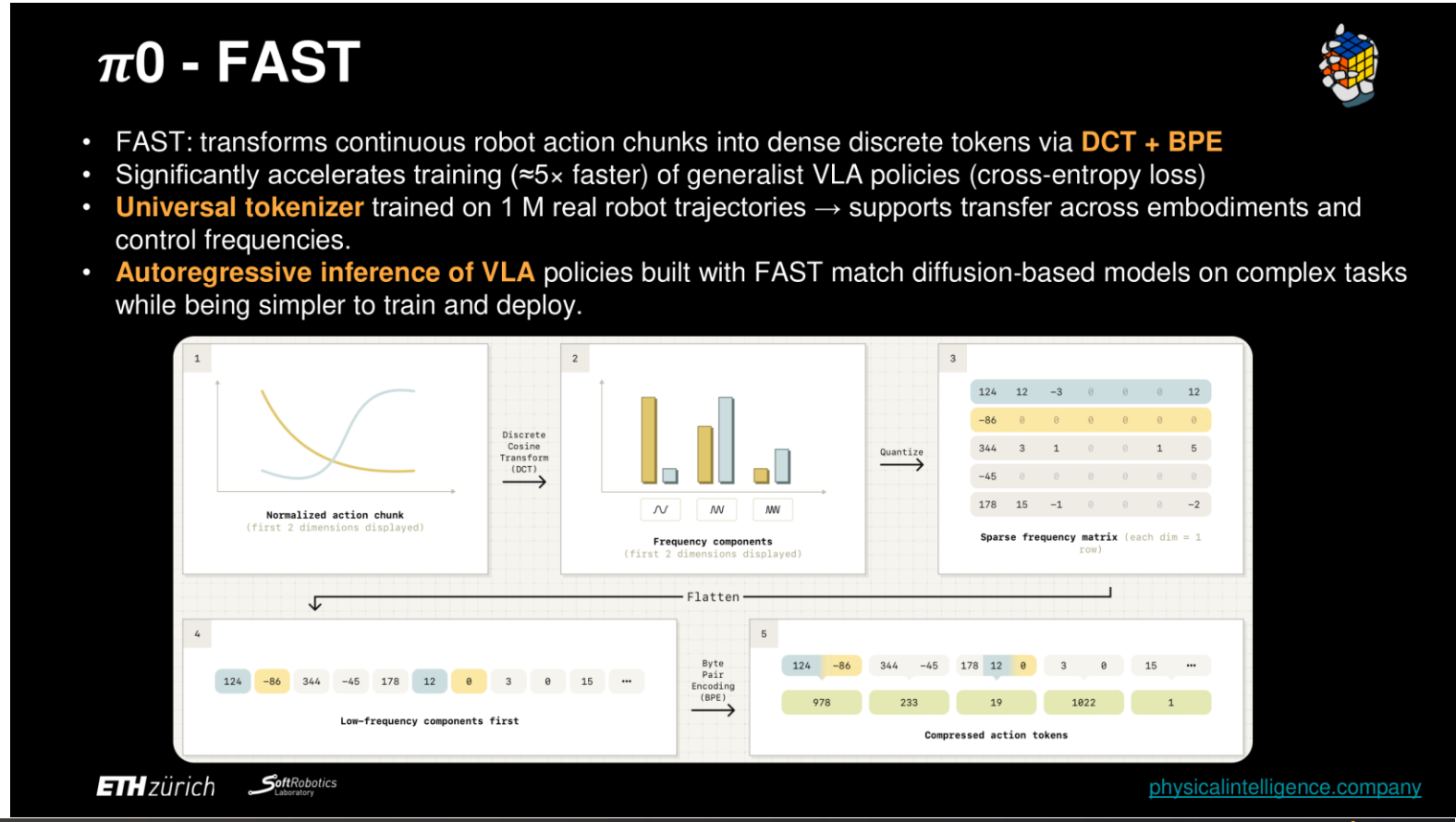
- High-quality curated demos
- Fluent task execution
- Critical for dexterous tasks

Result**: Outperforms OpenVLA,**
Octo, ACT, Diffusion Policy

Black et al., Pi0: A Vision-Language-Action Flow Model, 2024

Pi0-FAST: Better Action Tokenization

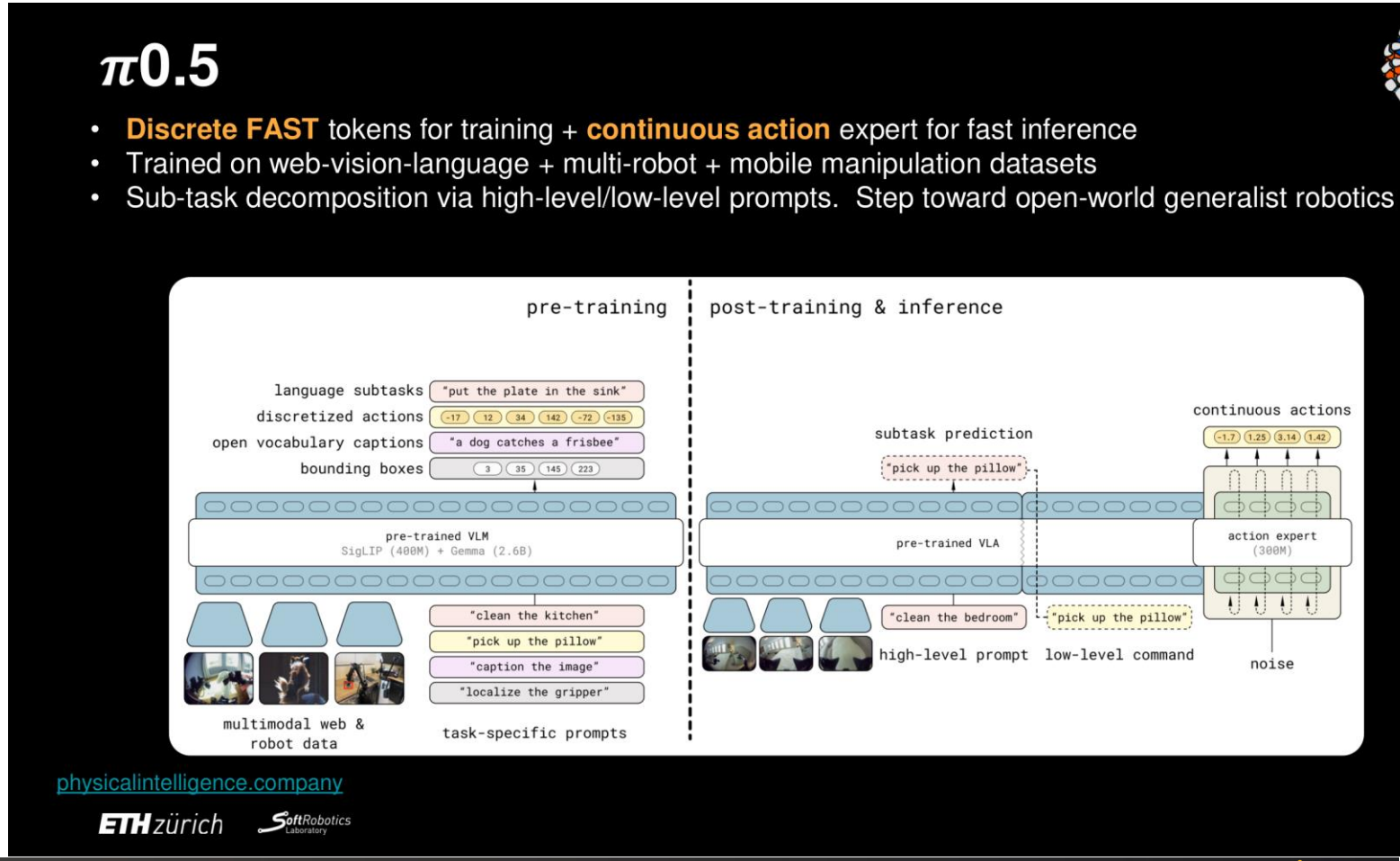
- FAST**: Frequency-domain Action Sequence Tokenization**
- Transform action chunks via DCT (Discrete Cosine Transform)
- Then BPE to create dense discrete tokens
- ~5x faster training than naive tokenization
- Bridges discrete (trainable) and continuous (precise)
- Compatible with standard LLM training pipelines



Pertsch et al., Pi0-FAST, 2025

Pi0.5: Knowledge Insulation

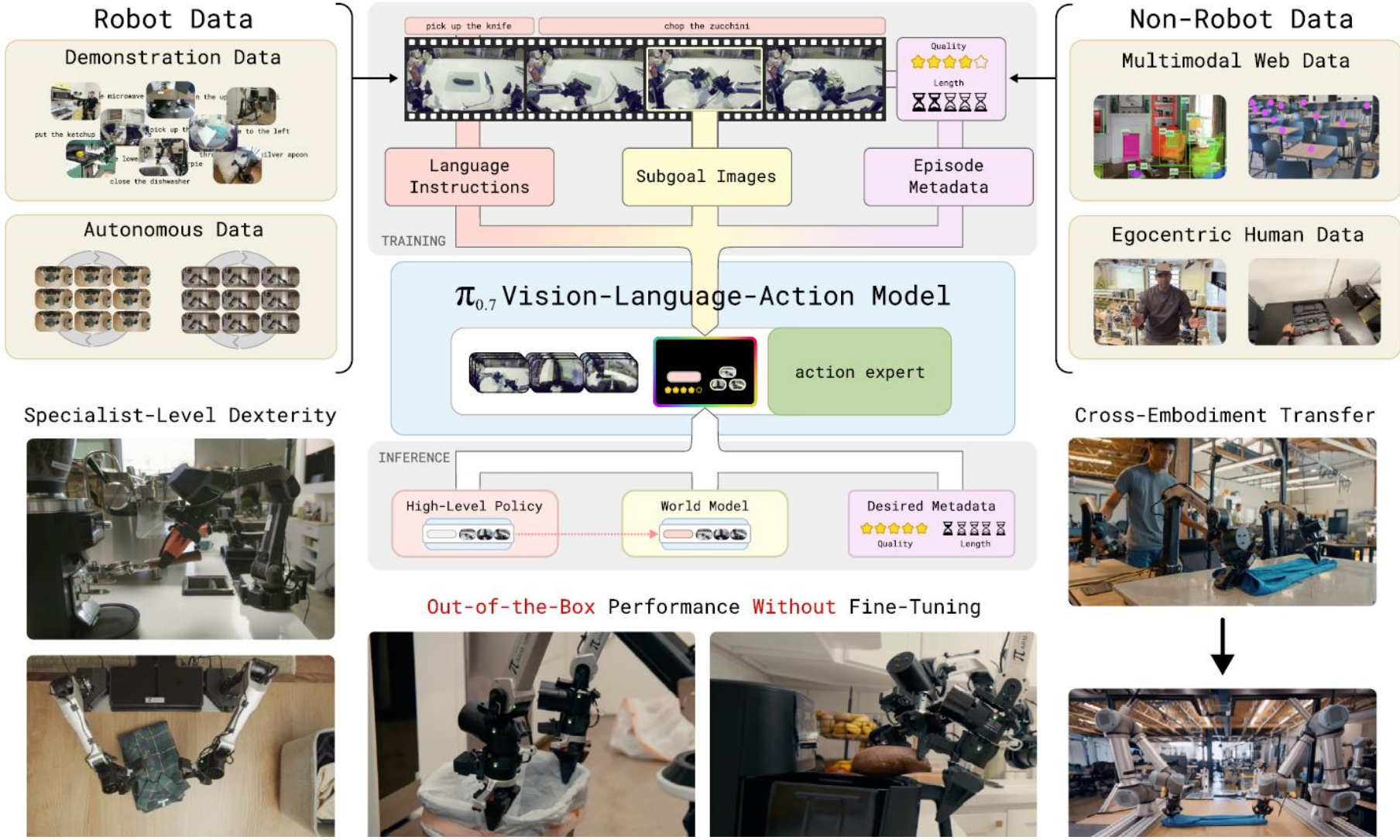
- Discrete FAST tokens for training
- Continuous action expert for fast inference
- Knowledge Insulation**: decouple VLM backbone** from action expert gradients
- VLM supervised with cross-entropy (stable)
- Action expert attends to VLM but no gradient backflow
- Retains semantic knowledge + faster training
- Multi-robot + mobile manipulation



Physical Intelligence, Pi0.5, 2025

Pi0.7: Steerable Generalist Robot Model

- 5B params **(4B Gemma3 backbone + 860M action expert)**
- Key innovation: **rich context conditioning** Language instructions + subtask labels
- Subgoal images (including generated goals)
- Episode quality, speed, strategy metadata
- Control modality labels
- Trains on heterogeneous data including failures
- Context disambiguates demonstrations



Ai et al., Pi0.7: A Steerable Generalist Robotic Foundation Model, 2026

Pi0.7: Emergent Compositionality

- Out-of-the-box dexterity **without task-specific fine-tuning**
- Matches specialist RL-finetuned models
- Cross-embodiment transfer**: folds laundry on UR5e** without laundry data for that robot
- Compositional generalization**: uses new appliances** by combining prior skills + web knowledge
- Follows diverse open-ended instructions
- Signs of language-model-like compositionality in physical behavior

Ai et al., Pi0.7: A Steerable Generalist Robotic Foundation Model, 2026

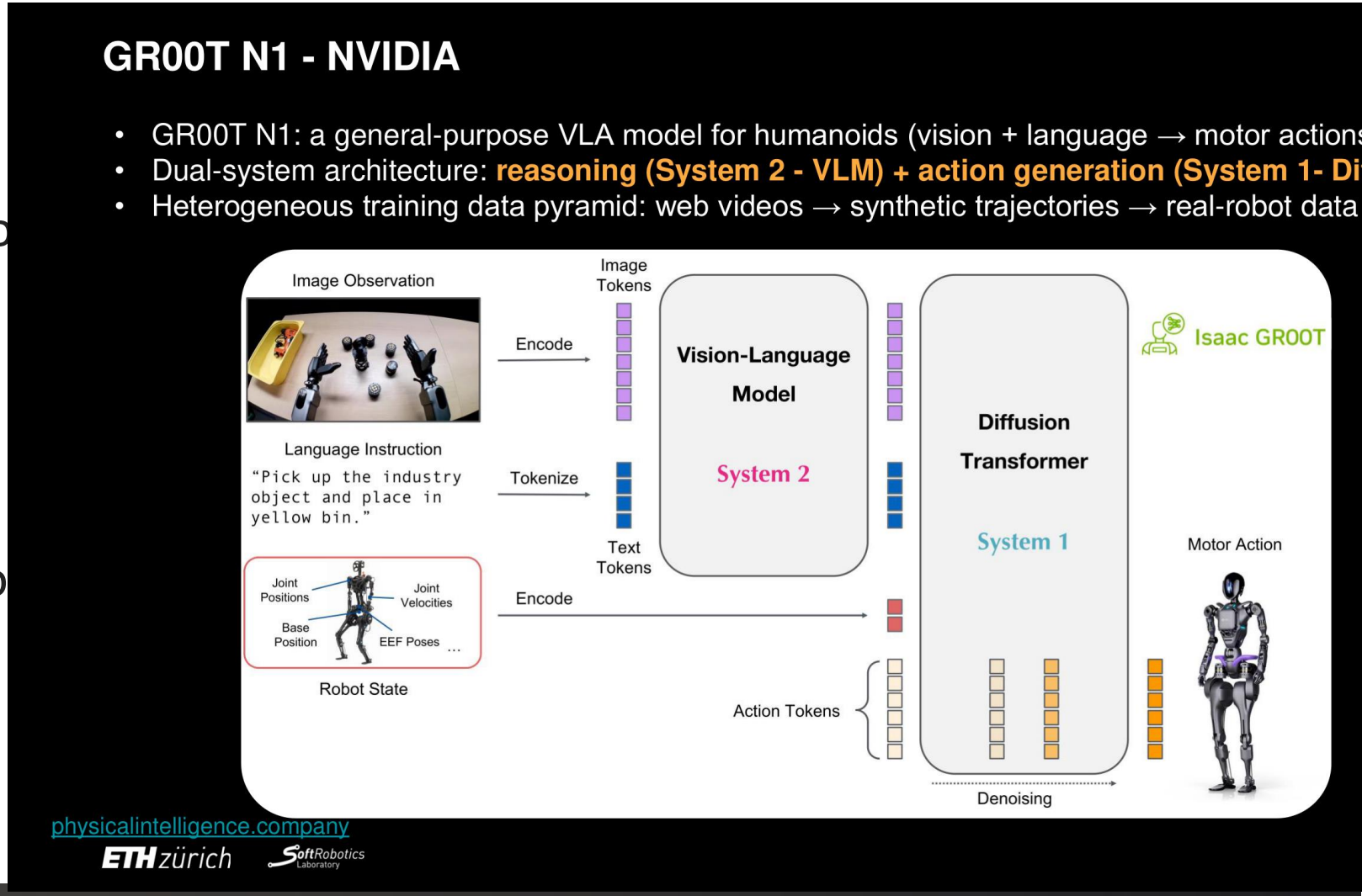
GR00T N1: NVIDIA's Humanoid VLA

- General-purpose VLA for **humanoid robots**
- Dual-system architecture:

System 1: fast reactive motor contro

System 2: slow deliberative reasoning

- Vision + language -> motor actions
- Designed for whole-body contro
- Part of NVIDIA's Isaac robotics platform



NVIDIA, GR00T N1, 2025

Gemini Robotics & Large Behavior Models

Gemini Robotics (Google)

- Multi-embodiment VLA
- Motion Transfer for

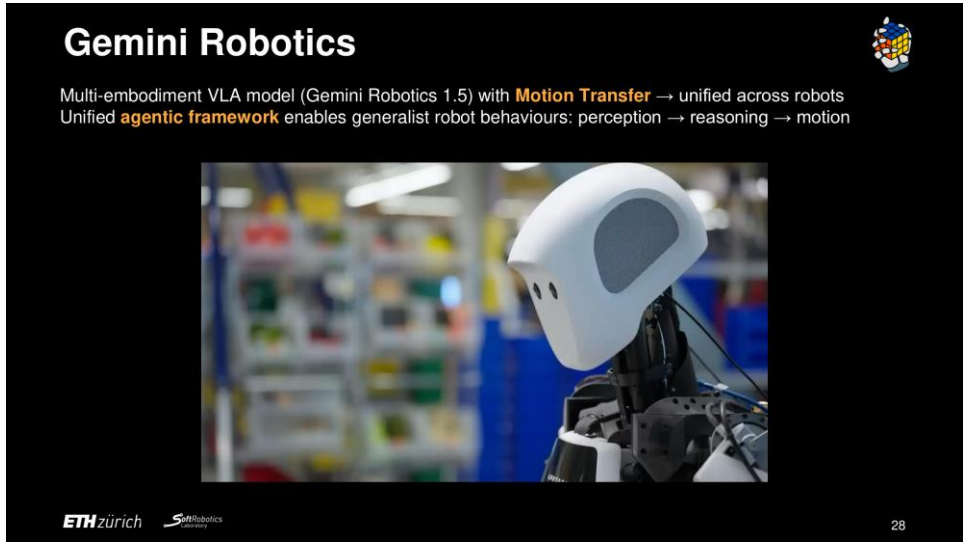
unified cross-robot control

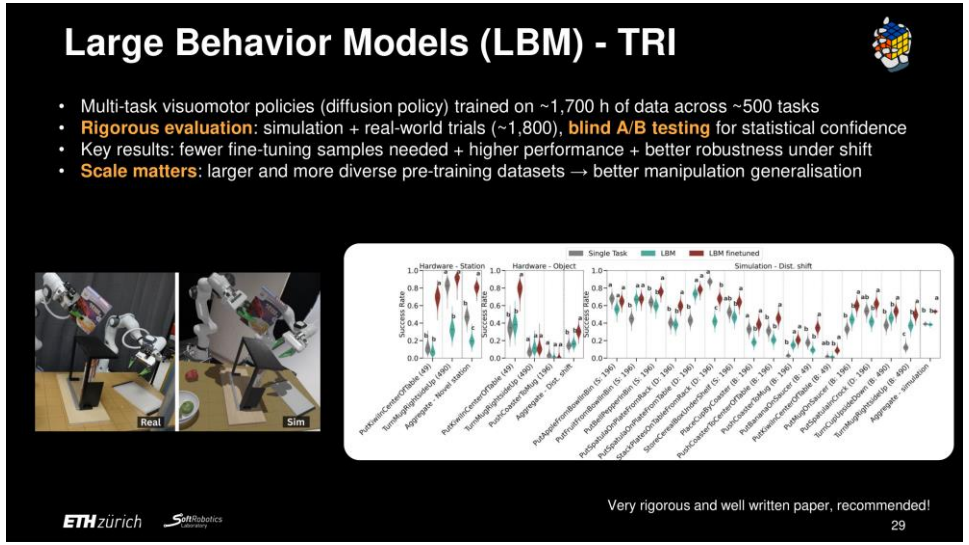
- Agentic framework for

long-horizon planning

LBM (Toyota Research)

- Diffusion policy at scale
- ~1,700h data, ~500 tasks
- Rigorous real-world eval
- Focus on reliable execution

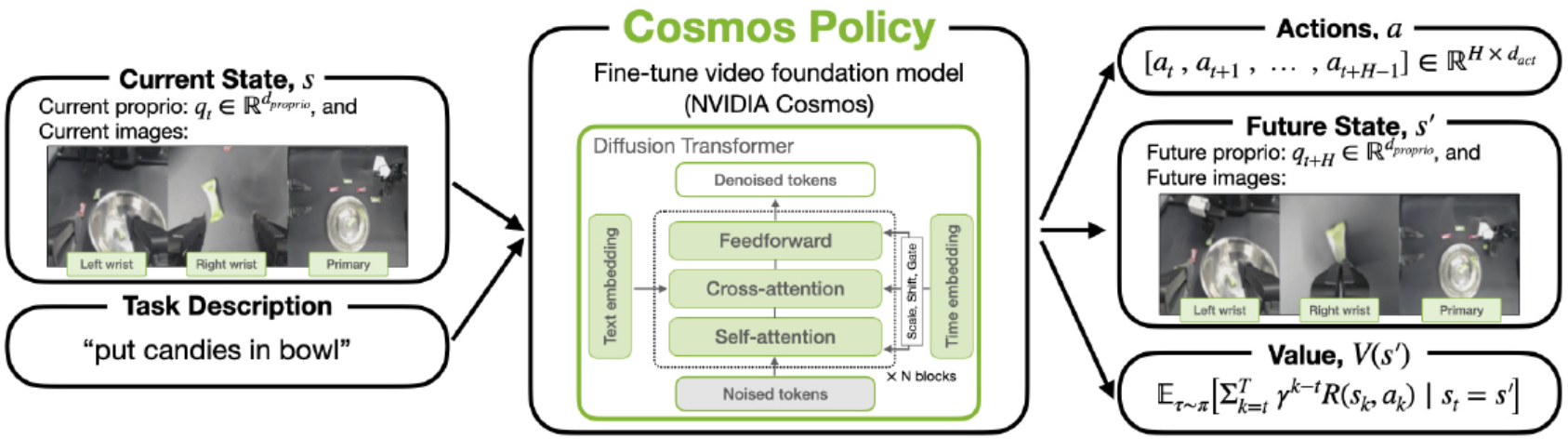




Google DeepMind, 2025; Toyota Research Institute, 2025

Cosmos Policy: Video Models for Robot Control

- Fine-tunes a **pretrained video generation model** into a robot policy
- Actions generated as latent video frames, no architecture changes
- Key insight: video models already understand physics and dynamics
- Repurposes video prediction as visuomotor control
- Outperforms state-of-the-art VLAs on manipulation benchmarks
- Effective in both simulation and real-world bimanual tasks
- Alternative paradigm: world-model-first rather than LLM-first

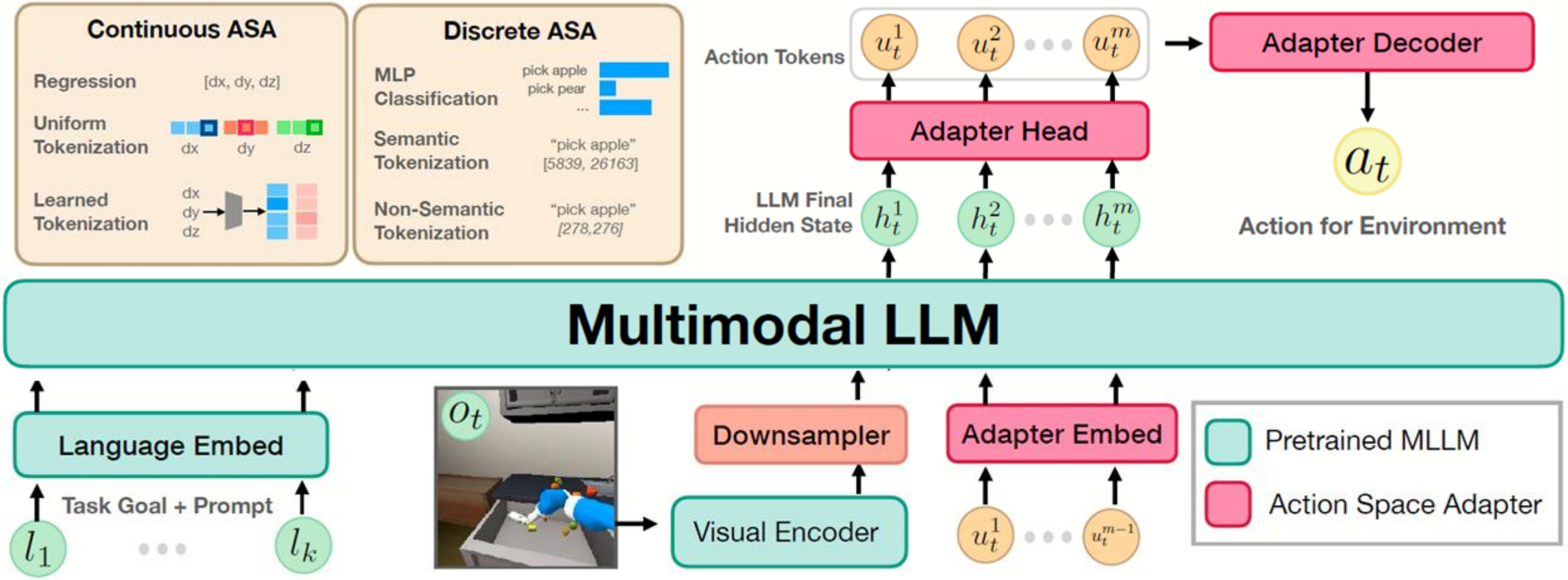


Kim et al., Cosmos Policy: Fine-Tuning Video Models for Visuomotor Control, 2025

VLA Landscape: Architecture Comparison

Model	Params	Action Rep.	Backbone	Year
RT-1	35M	Discrete bins	EfficientNet+Transformer	2022
RT-2	55B	Text tokens	PaLI-X / PaLM-E	2023
OpenVLA	7B	Discrete bins	Prismatic VLM	2024
Pi0	3.3B	Flow matching	PaliGemma 3B	2024
Pi0-FAST	3.3B	DCT+BPE tokens	PaliGemma 3B	2025
Pi0.5	~4B	FAST + flow	PaliGemma	2025
Pi0.7	5B	FAST + flow	Gemma3 4B	2026
GR00T N1	-	Dual system	Proprietary	2025
R3D	-	Diffusion	PointSAM encoder	2026
Cosmos Policy	-	Latent video frames	Cosmos video model	2025

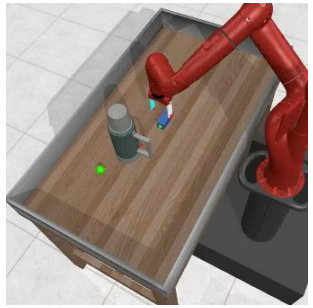
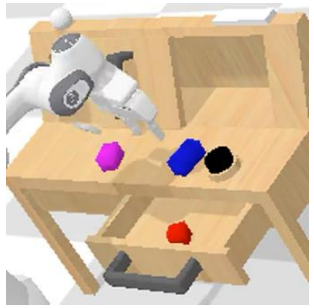
Can we train a **generalist** agent?



We finetune the Action Space Adaptors (ASAs), downsampler, and MLLM

Szot et al., Grounding Multimodal Large Language Models in Actions, NeurIPS 2024
Szot et al., From Multimodal LLMs to Generalist Embodied Agents: Methods and Lessons, CVPR 2025

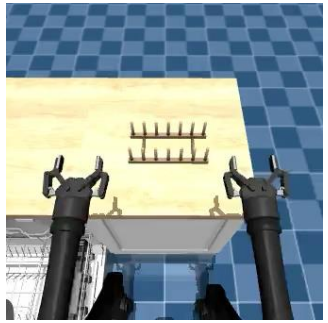
Static Manipulation

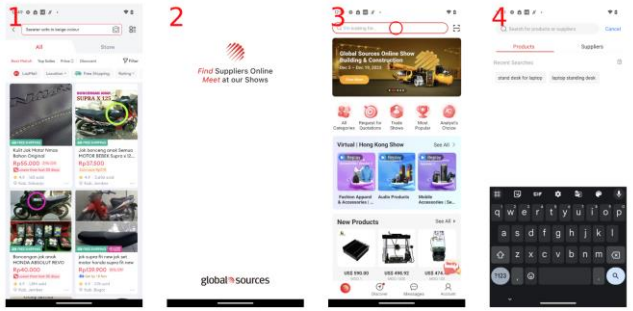
Games


Navigation

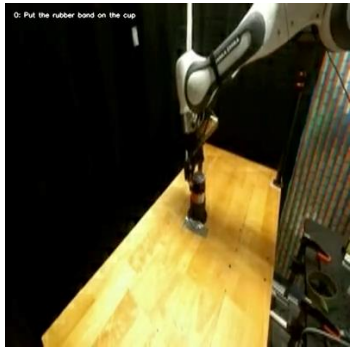

Mobile Manipulation

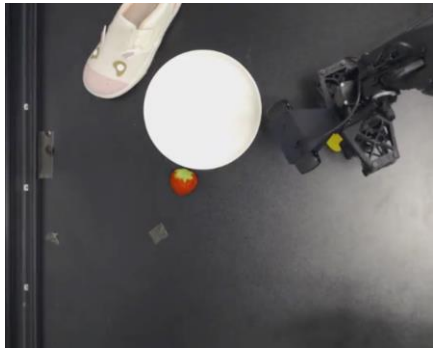



UI Control



Real Robots

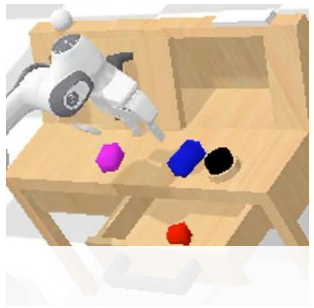




Character Control

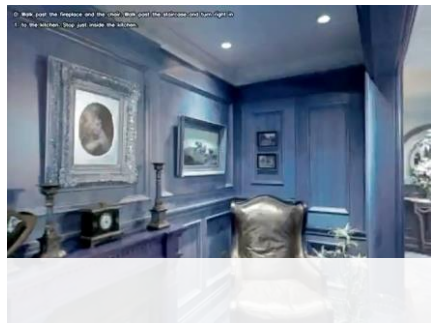

Planning

Static Manipulation

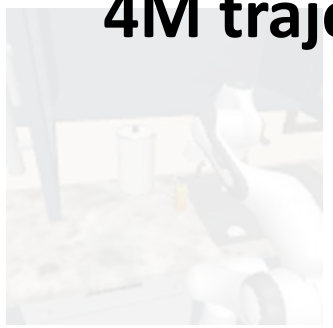


Games

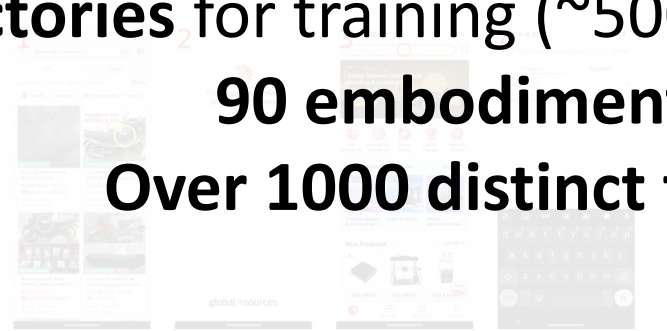
Navigation



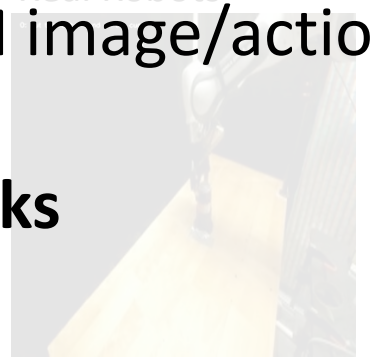
Mobile Manipulation



UI Control



Real Robots

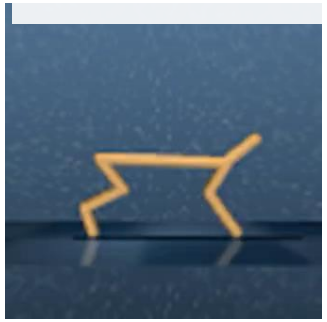


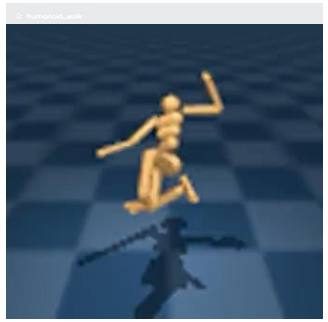
4M trajectories for training (~500M image/actions)

90 embodiments

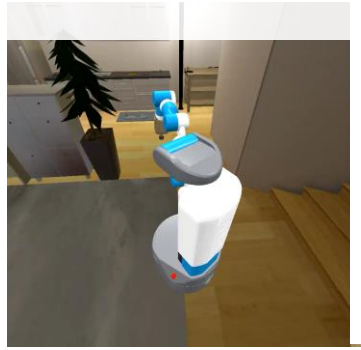
Over 1000 distinct tasks

Character Control





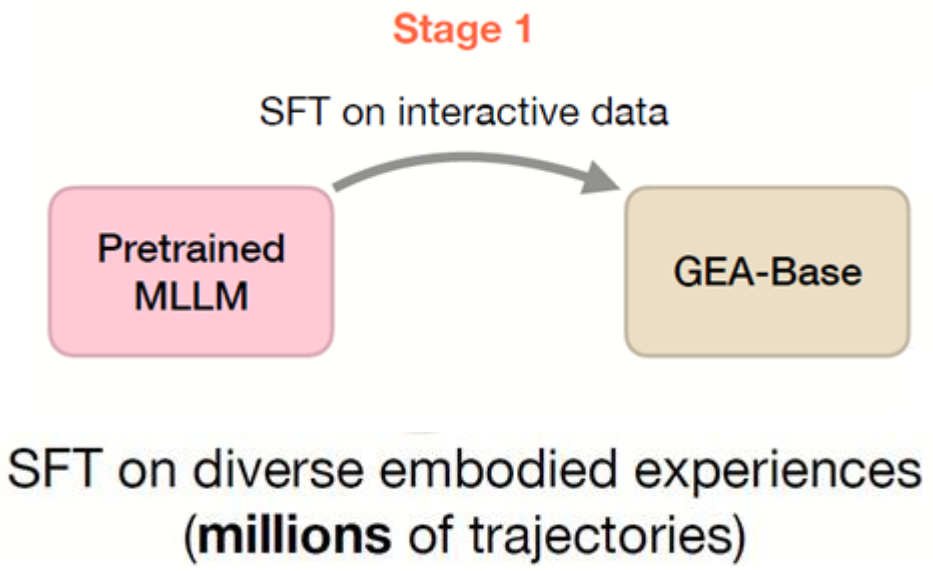
Planning



Training

- We alternate supervised fine-tuning (SFT) and reinforcement learning (RL)

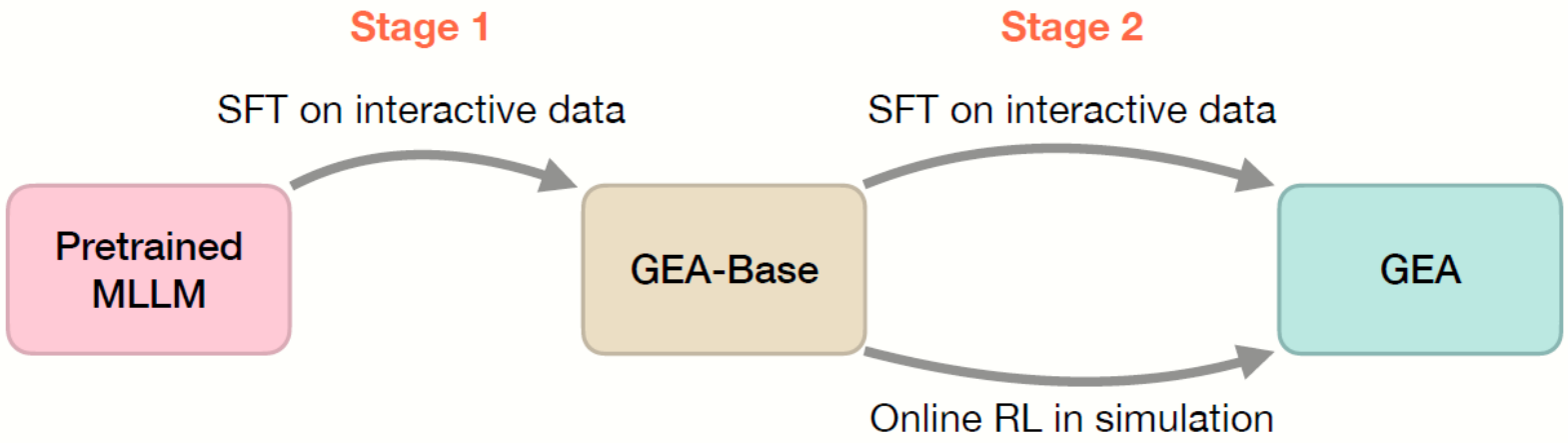
- SFT – Gather human or planner-based demonstrations of successfully executing task
  - Training is just next token prediction (correct action) at any point



Szot et al., From Multimodal LLMs to Generalist Embodied Agents: Methods and Lessons, CVPR 2025.

Training

- We alternate supervised fine-tuning (SFT) and reinforcement learning (RL)
- RL has different characteristics when distributing!
  - Workers take a *variable* amount of time
    - Robot gets stuck/fails, times out at max timesteps
  - Dominating bottleneck is policy *inference* not weight update (training) or environment



Do many different tasks

and

Generalizes to new settings

Szot et al., From Multimodal LLMs to Generalist Embodied Agents: Methods and Lessons, CVPR 2025.

GEA Evaluation

Simulated Robotics

Games

UI Control

Planning

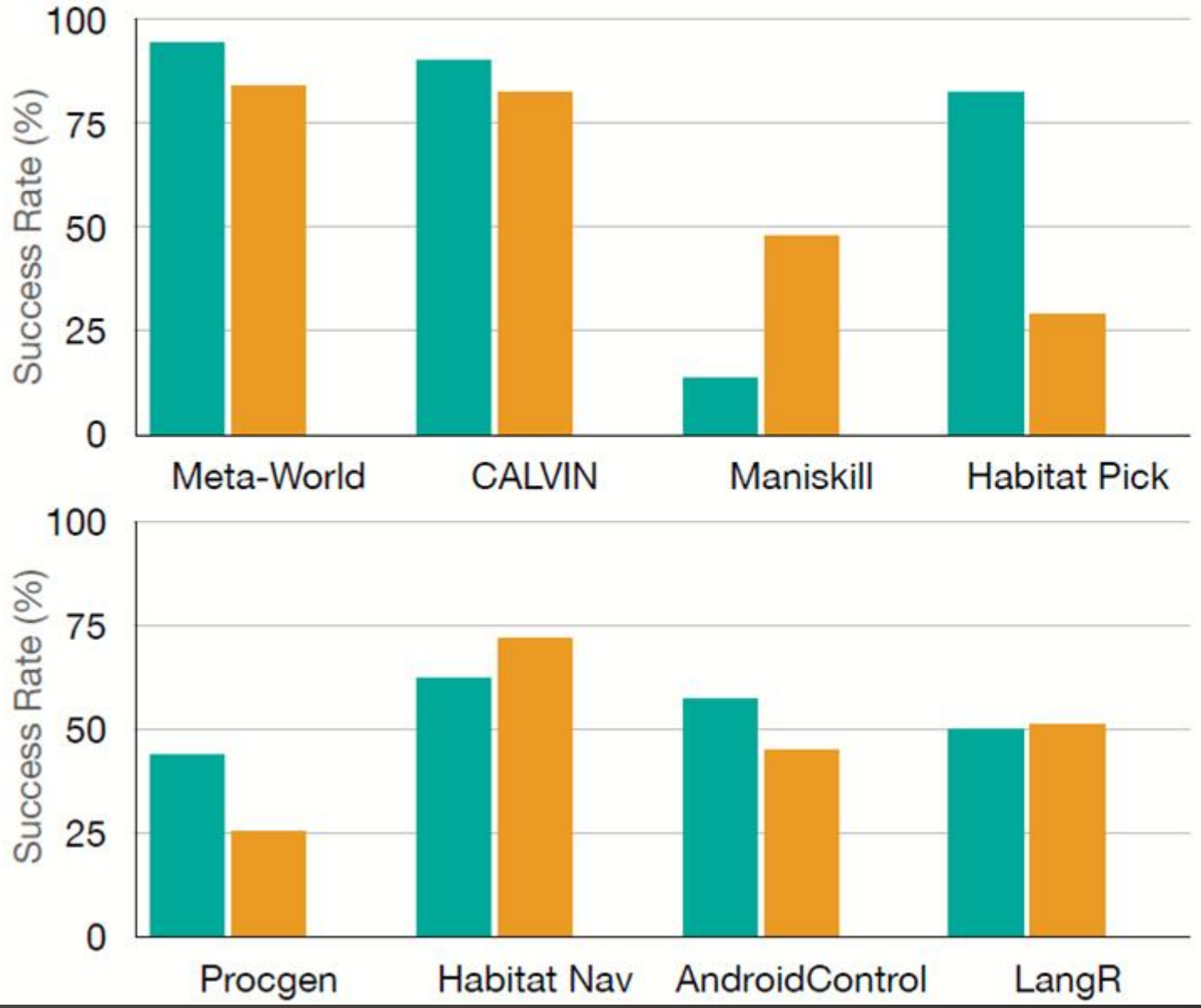
Navigation

Assess Generalization To:

- New Scene Layouts
- New Instructions
- New Backgrounds
- New Objects

Compare against benchmark specific specialists

Zero-Shot Generalization



GEA: Single model evaluated on *all* benchmarks

Domain specialist: Prior SOTA *per* benchmark

Success rate on *unseen* settings

Are we done? Just scale it?

- **What this "solves":**
  - Natural language open-world tasks
    - Language-boosted generalization
  - Multi-model by nature
  - Common-sense reasoning! (no more knowledge hacking)

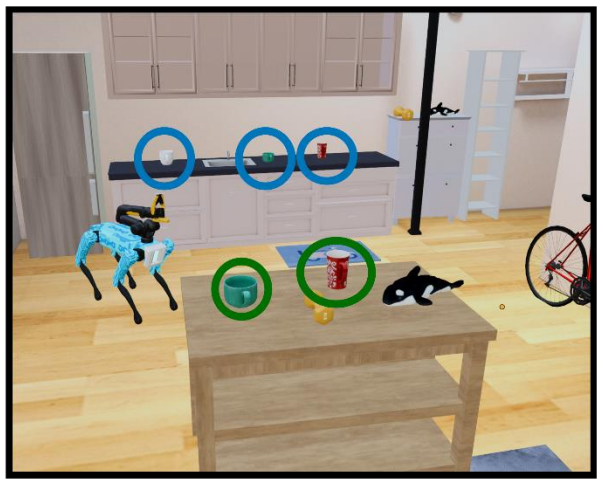
Necessary, but maybe not sufficient

- **What this doesn't address**
  - Robustness to distribution shifts
  - Reliability, safety, …
  - Ambiguity, lack of context, personalization in requests

➔Communication & collaboration!

Ask-To-Act Task

Task: Bring the cup and place it on the coffee table

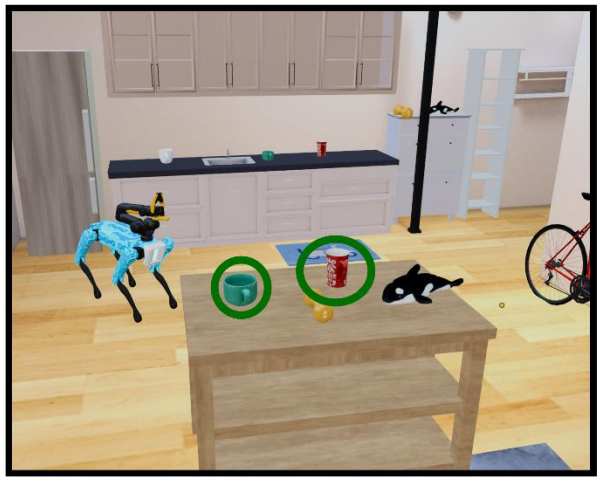


Navigate to kitchen

Q: Is the cup on the table?

A: Yes

Navigate to table



Q: Is it the red cup?

A: No

Pick 

Navigate to coffee table

Place cup on coffee table

- **Goal**: How can we build agents that can solve tasks which require resolving ambiguity?
- **Given an instruction:** Bring the cup and place it on the coffee table
  - Agent needs to look around and reason to ask clarification question
  - Key aspect: When and what should it ask?

Ask minimum number of questions

Ramrakhya et al., Grounding Multimodal LLMs to Embodied Agents that Ask for Help with Reinforcement Learning, arXiv:2504.00907.


Habitat 3.0
Meta
This work:
• 83 scenes ReplicaCAD
• 42 object categories
Evaluation:
1. Novel scenes
2. Novel compositions of ambiguity types

Task – Distinguishing Characteristics

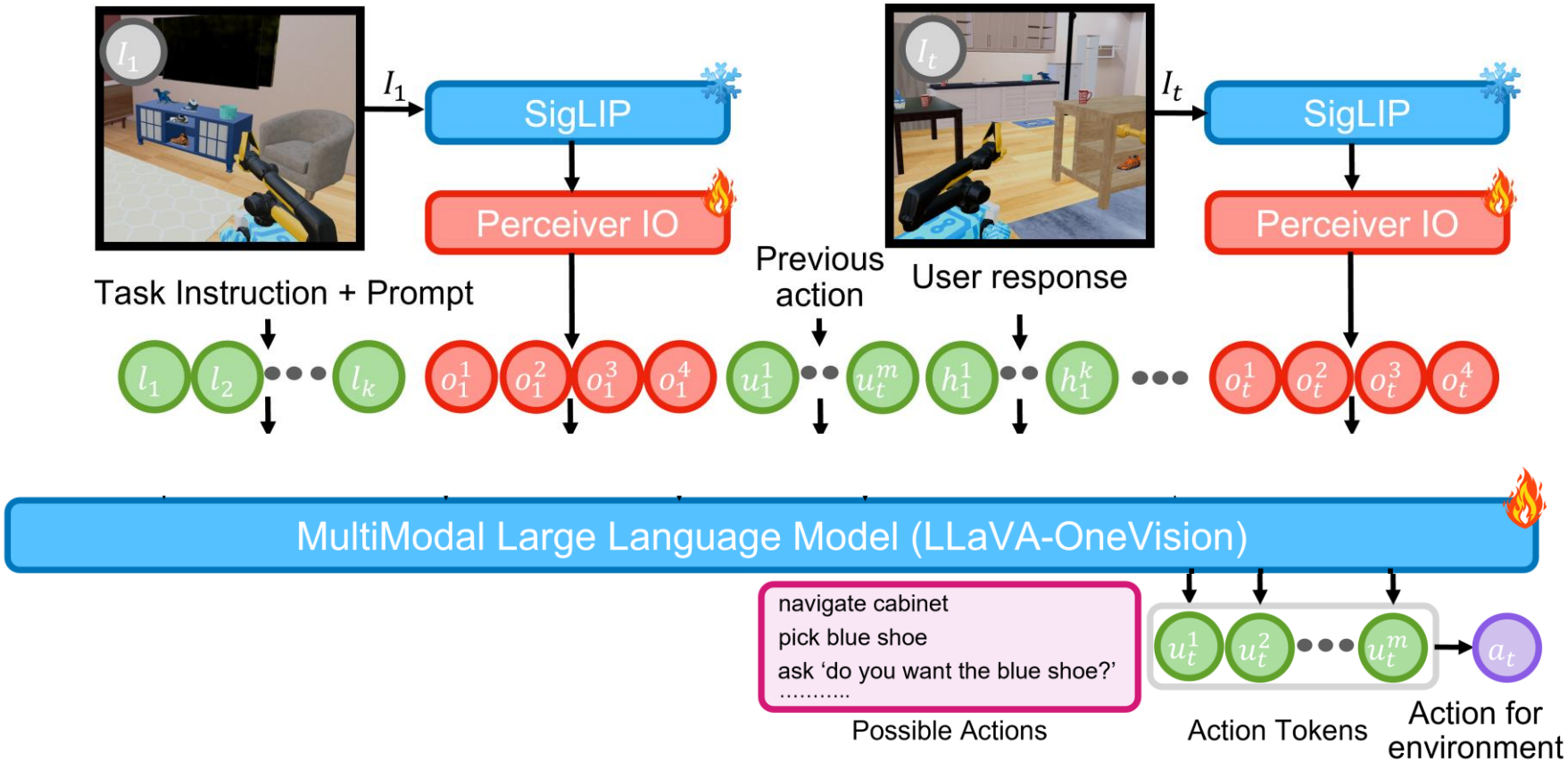
(a.) Attribute recognition

(b.) Spatial reasoning

(c.) Attribute + Spatial reasoning

- Various aspects that can discriminate target

Architecture

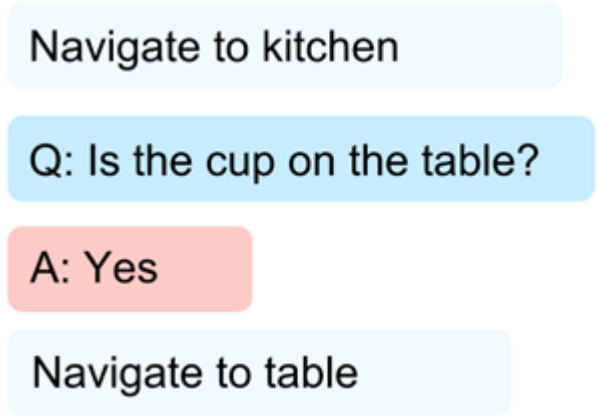


Pre-trained MLLM | Vision Adapter

Challenge: Data for this Task?

- Human demonstrations?

- Reinforcement learning with manually designed reward?



Typical tasks have well-defined success

Training for communication is hard!

Idea: Per-step reward function generated by LLM

- Requires careful task/environment representation
- Leverages privileged information made possible in simulator

Data Generation

```
Instruction: Bring the bowl and put it in the sink

Here's a description of state of the house that specifies which receptacles agent can navigate to and objects that it can interact with:

Receptacles in room:
living_room: gray chair, light table, dark table, couch, tv_stand
kitchen: cabinet, sink, drawer_0, drawer_1
bedroom: blue cabinet

Objects available on different receptacles in the house:
sink contains blue casserole, red plate, white cup
dark table contains red bowl, yellow bowl, black toy
light table contains yellow dumbbell, blue bowl, red bowl
sofa contains red bowl, yellow bowl, red towel
```

- Simulator oracle information gives us:
  - WorldGraph: Information about the scene and relationships
  - What the agent has asked already and human response

We input this into a (language-only) LLM to generate process reward model

- Optimal questions (e.g. attributes) to ask about
- Optimal number of questions

Data Generation

- Reward function: $r_t = r_1 \cdot \mathbb{1}_{\text{success}} + r_2 \cdot \mathbb{1}_{\text{subgoal}} + r_3 \cdot \mathbb{1}_{\text{useful\_question}} - r_4 \cdot \mathbb{1}_{\text{exceed\_budget}}$

- Key challenge:
  - Having an LLM within RL loop is extremely challenging
  - To study our (general) method for this task, we constrain via prompting/decoding the MLLM outputs so that we can precompute

Metrics

Method	Full Obs.	Unseen Scenes			Unseen Tasks		
		SR (↑)	ARS (↑)	QR (↓)	SR (↑)	ARS (↑)	QR (↓)

- **Success Rate**
- **Ambiguity-Resolution Efficiency Score (ARS)**

$$ARS = \frac{\mathbb{1}_{\text{success}}}{1 + \text{abs}(q_{\text{relevant}} - K) + q_{\text{irrelevant}}}$$

- **Question Ration (QR)**

$$(q_{\text{relevant}} + q_{\text{irrelevant}})/K$$

Ramrakhya et al., Grounding Multimodal LLMs to Embodied Agents that Ask for Help with Reinforcement Learning, arXiv:2504.00907.

Quantitative Results

	Method	Full Obs.	Unseen Scenes SR (↑)	Unseen Scenes ARS (↑)	Unseen Scenes QR (↓)	Unseen Tasks SR (↑)	Unseen Tasks ARS (↑)	Unseen Tasks QR (↓)
Privileged Info (+In-Context)	1) WG + ReAct*	✓	56.1	50.5	3.2	47.3	37.4	2.3
	2) WG + ReAct* (FS)	✓	96.2	49.7	2.3	94.7	35.6	1.9
Partially-Obs.	3) WG + ReAct* (FS)	✗	61.8	46.9	3.2	49.2	32.2	2.2
GPT	4) GPT4o + SoM + ReAct	✗	25.2	20.1	1.9	19.8	12.5	1.1
	5) LLaVA-OV SFT	✗	48.2	46.9	**1.5**	34.1	26.1	**0.8**
Ours (RL)	6) LLaVA-OV RL (Ours)	✗	**89.8**	**63.2**	2.6	**65.2**	**32.4**	2.5

- Overall, difficult task even for state of art closed MLLMs
  - Note we tried reasoning models but did not help much (privileged info makes unnecessary?
- RL-based training much more successful!

Ramrakhya et al., Grounding Multimodal LLMs to Embodied Agents that Ask for Help with Reinforcement Learning, arXiv:2504.00907.

Expanded Question Set

	Question	Query Type	Answer type
1)	Is it the [Attr] [Obj]?	Object attribute	Yes/No
2)	Is the object on the [Recep]?	Object location	Yes/No
3)	Is it the [Size] [Obj]?	Object size	Yes/No
4)	Can you describe appearance of [Obj]?	Object attribute	Language
5)	Where is the [Obj] located?	Object location	Language
6)	Describe which [Obj] instance you want and where is it located?	Object attribute and location	Language
7)	Which [Recep] should I place the object?	Target receptacle	Language

	Method	UNAMBIGUOUS			SINGLE OBJECT			MULTI-OBJECT		
		SR (↑)	ARS (↑)	QR (↓)	SR (↑)	ARS (↑)	QR (↓)	SR (↑)	ARS (↑)	QR (↓)
1)	LLaVA-OV SFT	77.8	–	**1.2**	50.2	45.3	**1.6**	32.7	25.3	**1.1**
2)	LLaVA-OV RL (Ours)	**96.8**	–	1.5	**93.5**	**85.3**	1.7	**60.2**	**33.3**	2.7

- Method easily expandable to arbitrary questions
- Making it fully open + LLM rewards in RL loop challenging though

Ramrakhya et al., Grounding Multimodal LLMs to Embodied Agents that Ask for Help with Reinforcement Learning, arXiv:2504.00907.

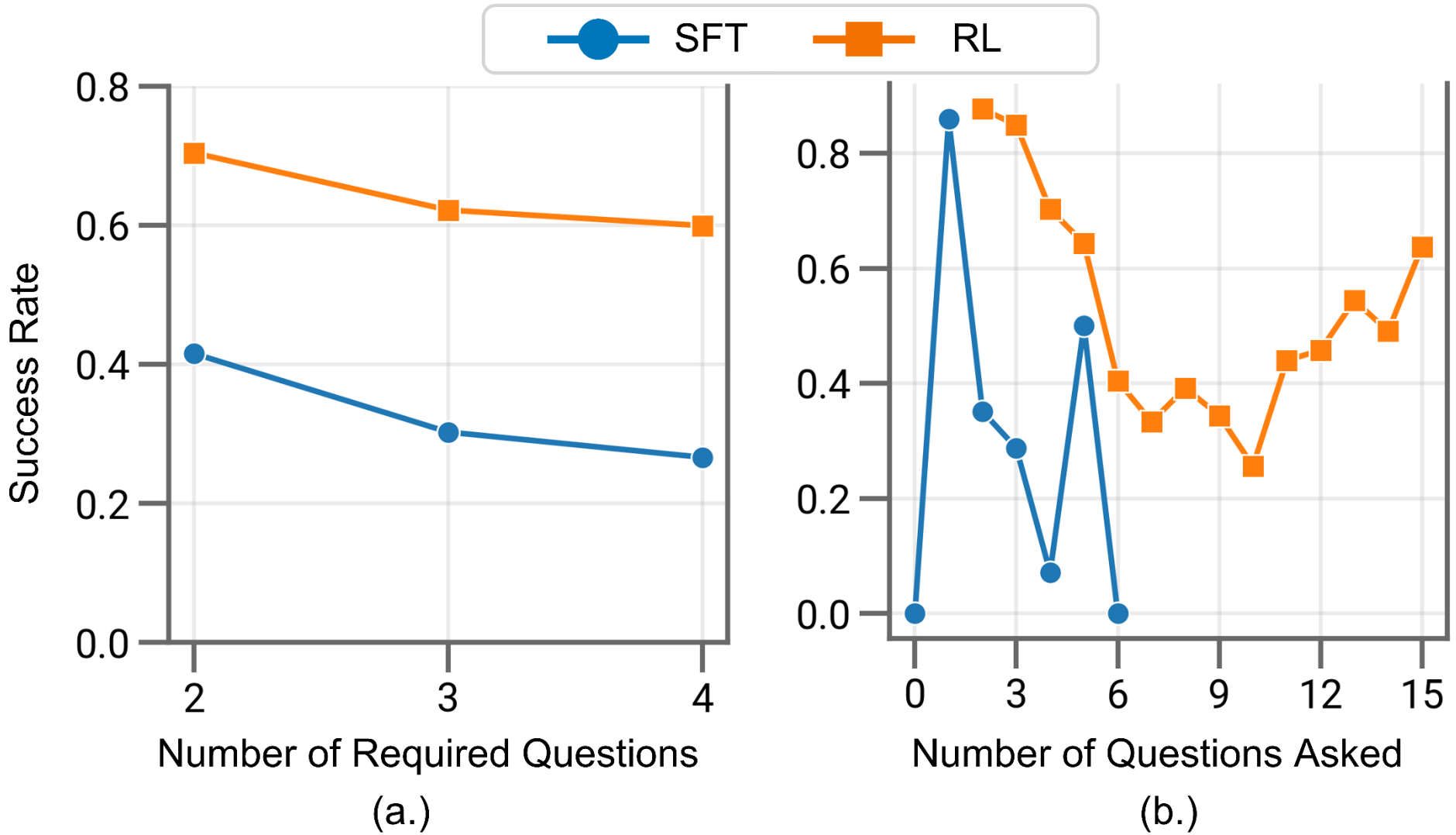
Meta

Analysis – Do We Need a Process Reward Model?

	Method	Unseen Scenes			Unseen Tasks		
		SR (↑)	ARS (↑)	QR (↓)	SR (↑)	ARS (↑)	QR (↓)
1)	Success Reward	0.0	0.0	0.0	0.0	0.0	0.0
2)	Subgoal Reward	34.9	30.7	3.8	16.5	6.9	2.9
3)	Ours	**89.8**	**63.2**	**2.6**	**65.2**	**32.4**	**2.5**

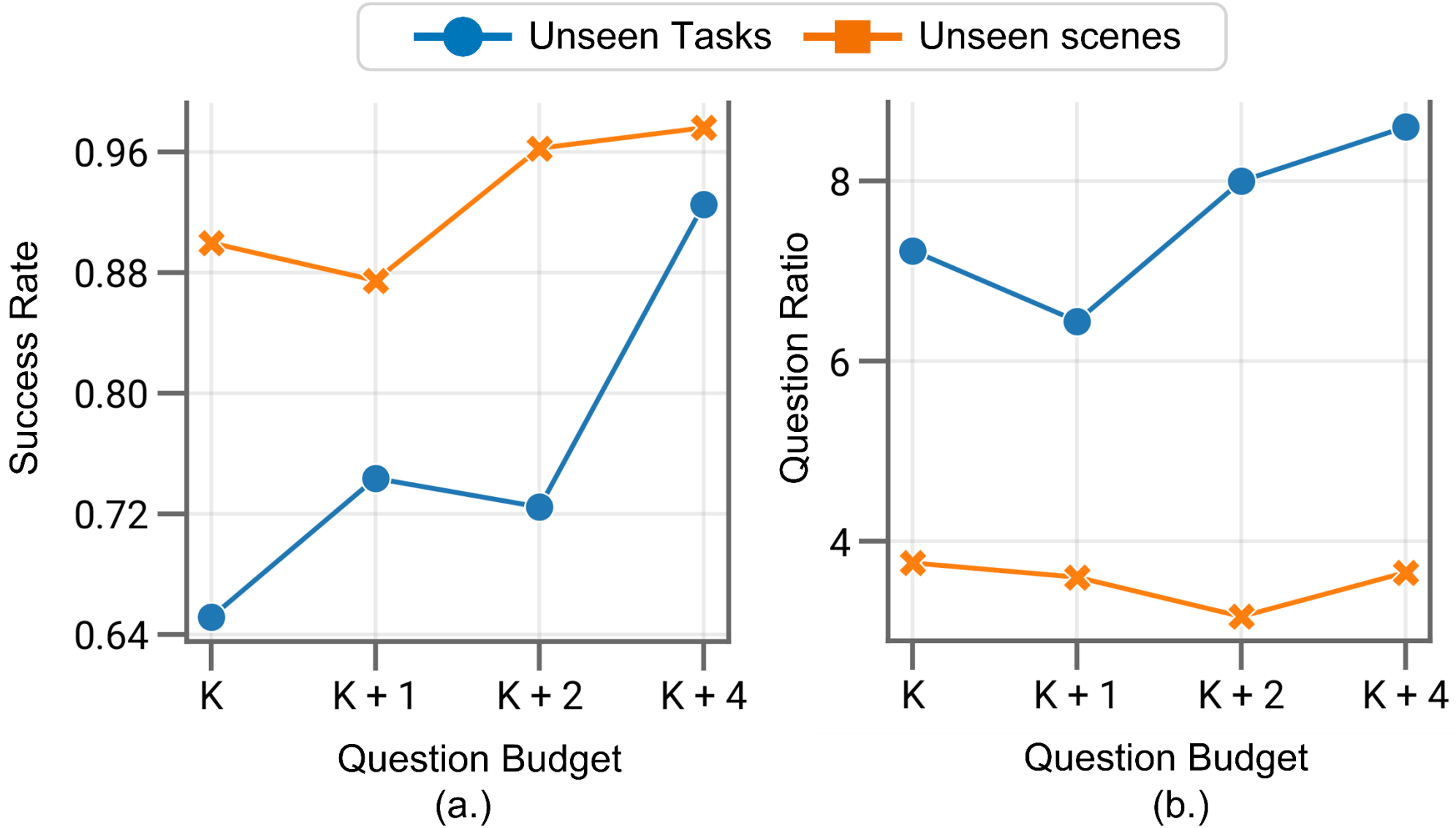
Yes!

Analysis – Do We Need RL?



Yes!

Analysis – Question Budget vs. Performance



Why making ChatGPT for robotics is not straightforward

Data

Cannot simply scrape internet
→ Human videos
→ Synthetic Data
→ Simulation

Control Frequency

Need real time control
→ System 1/ System 2
→ Chunk Quantization (FAST)

Cross-embodiment

Observation and Action Space depends on the Robot Embodiment
→ Scaling Data
→ Latent Actions

ETH zürich Soft Robotics Laboratory

State of VLA Research

Strong industry interest

Google, NVIDIA, Tesla, Figure,

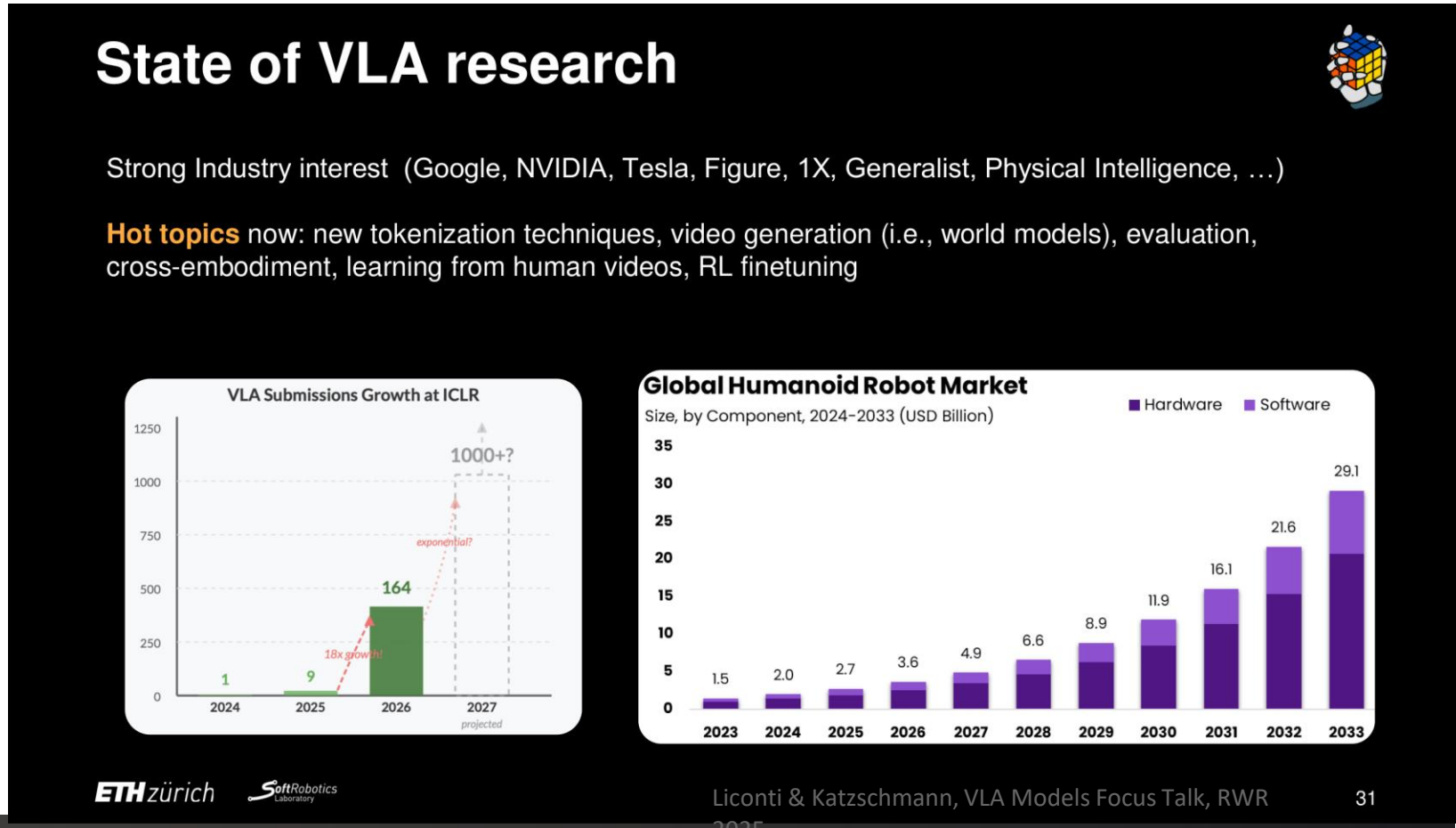
1X, Physical Intelligence, ...

Hot topics now:

- Cross-embodiment transfer
- System 1 / System 2 control
- Data: human video, synthetic, simulation, web-scale
- Real-time inference
- Evaluation methodology
- World models for planning

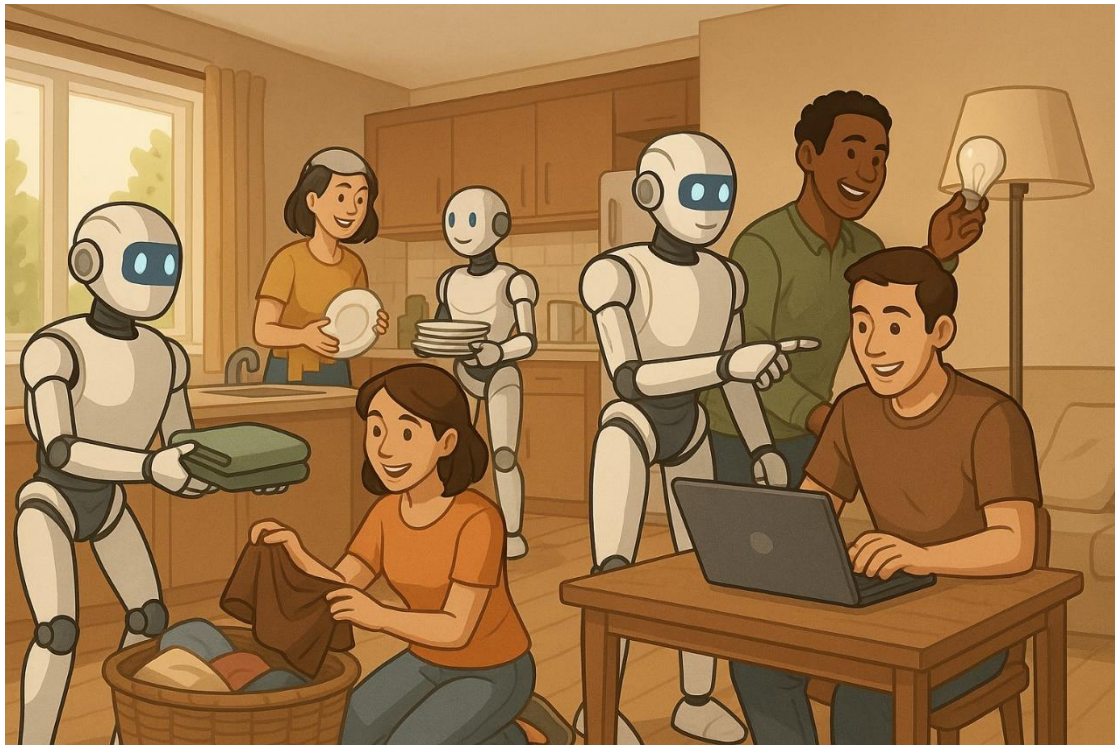
Open questions

- Robustness & generalization



The Future of Collaboration

- **The Future:** Large collaborative human-robot teams
  - The robots now know language!



- **Some research questions:**
  - Should robot-robot communication be “compressed” through language? Or low-level features / embeddings etc.?
  - How can we train complex multi-turn dialogue?
    - How can we train dialogue/communication with MLLMs in the loop?
    - Zero-shot MLLMs are not there (and may not be there in near future?)
    - Need a process of synthetic data generation + curation, across dialogue
    - Can we leverage verifiers/reward-generators in such ambiguous domains?
    - Do we need process reward models (e.g. GRPO)

Acknowledgement and Questions

Andrew Szot

ML Ph.D. (co-advised with Dhruv Batra)

Ram Ramrakhya

CS Ph.D. (co-advised with Dhruv Batra)

Yusuf Ali

CS Ph.D.

Moises Andrade

CS Ph.D.

Shivang Chopra

CS Ph.D.

Jeremiah Coholich

Robo Ph.D.

Shaunak Halbe

ML Ph.D.

Chengyue Huang

ML Ph.D.

Jaehyeon Son

ML Ph.D.

Karmesh Yadav

CS Ph.D. (co-advised with Dhruv Batra)

Xijia (Polina) Zhang

Robo Ph.D.