Social Learning

Saurabh Gupta

Solving a RL Problem

Better Reward Signals

Sim2Real

Better Optimization

Convert into a Supervised Training Problem

Solve a Related but Supervision-rich Problem

Build Models and Plan with Them

Model-free RL with sparse rewards

Known reward, known model. Model-based RL

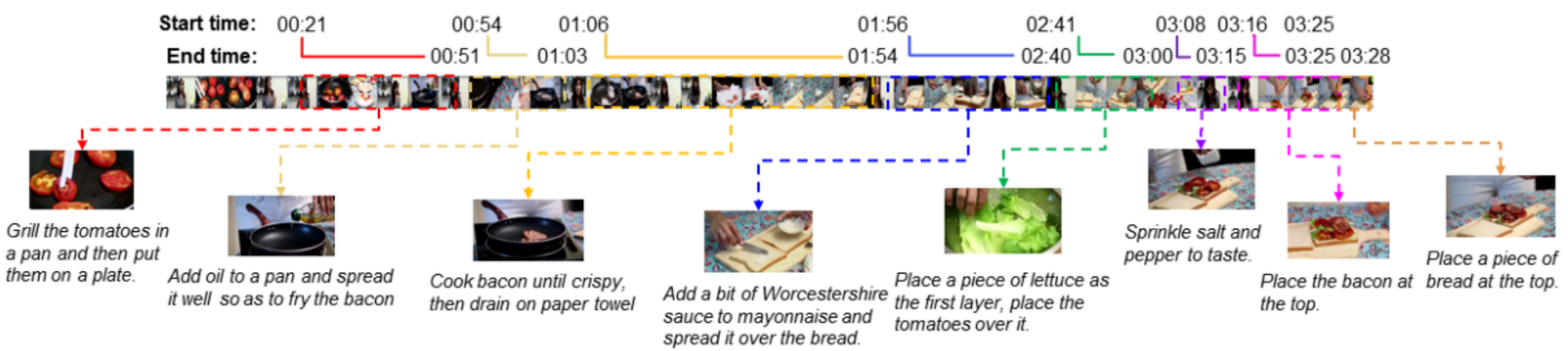
Social Learning

Learn by observing other agents solving the same problem.

Social Learning

What all can we learn?

High-level plans

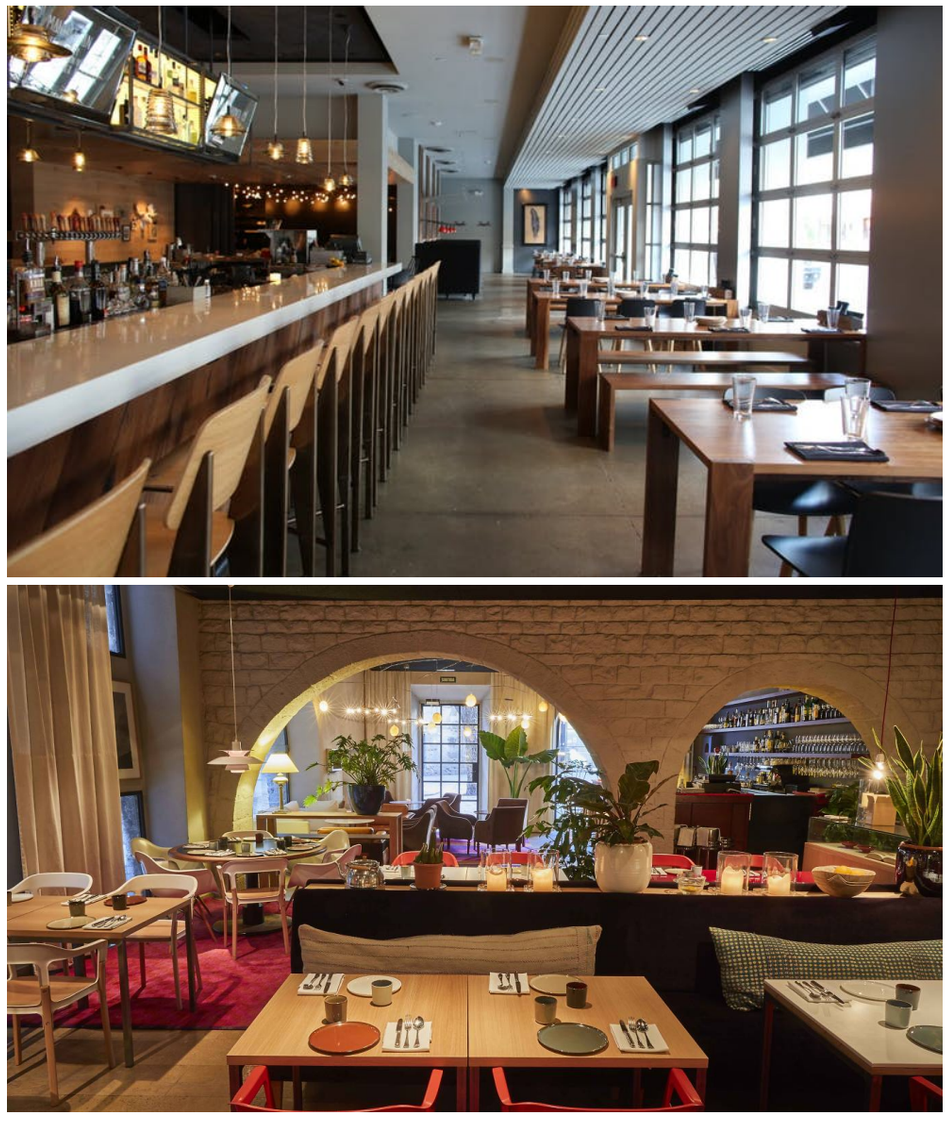


YouCook2 Dataset: http://youcook2.eecs.umich.edu/

Social Learning

What all can we learn?

High-level semantic priors



Finding a bathroom in a new restaurant

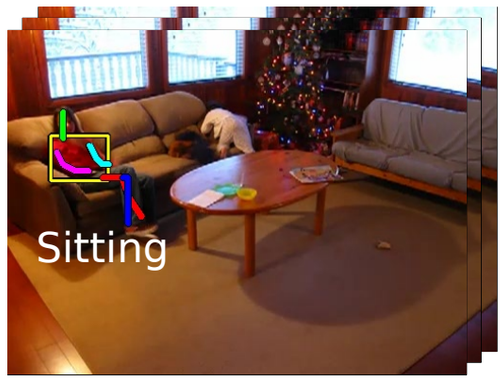
Learn by mining spatial co-occurrences from online videos

Semantic Visual Navigation by Watching YouTube Videos Matthew Chang, Arjun Gupta, Saurabh Gupta. NeurIPS, 2020

Social Learning

What all can we learn?

Environmental affordances (third-person time-lapse)

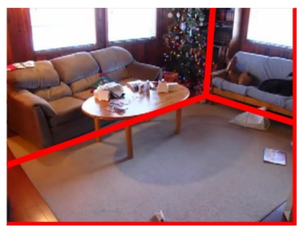


(a) Action and Pose Detections

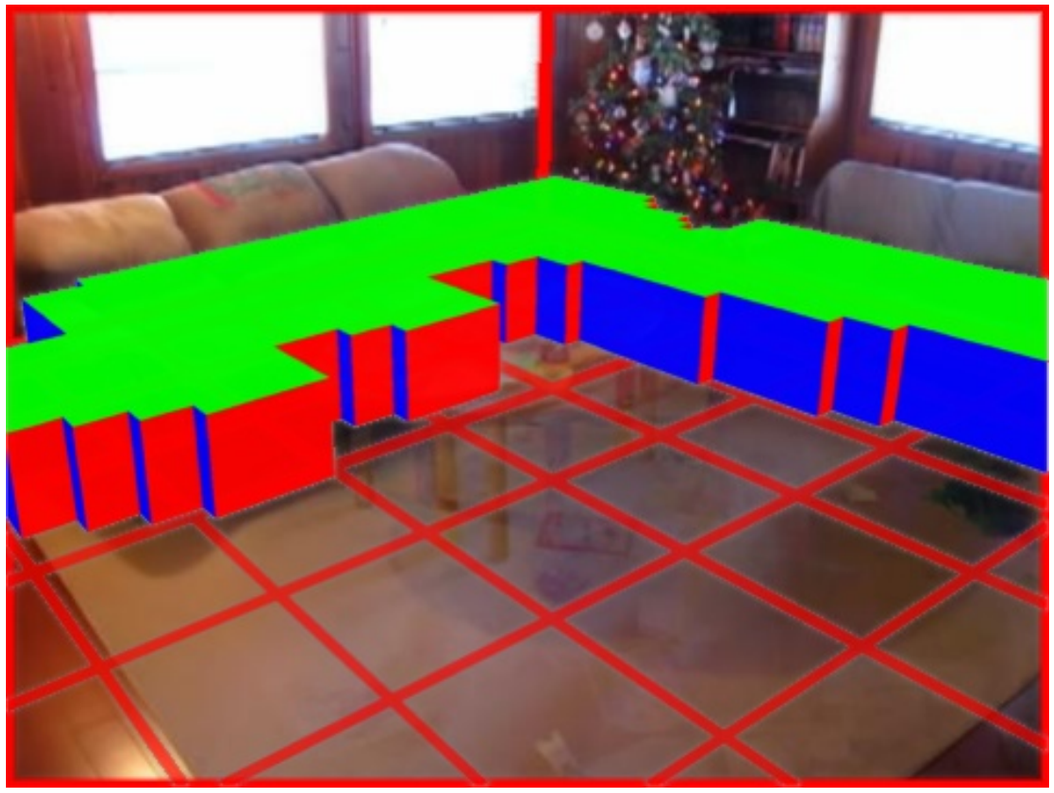
(b) Poses (potentially aggregated over time)

(c) Estimates of functional surfaces

. . .



(d) 3D room geometry hypotheses



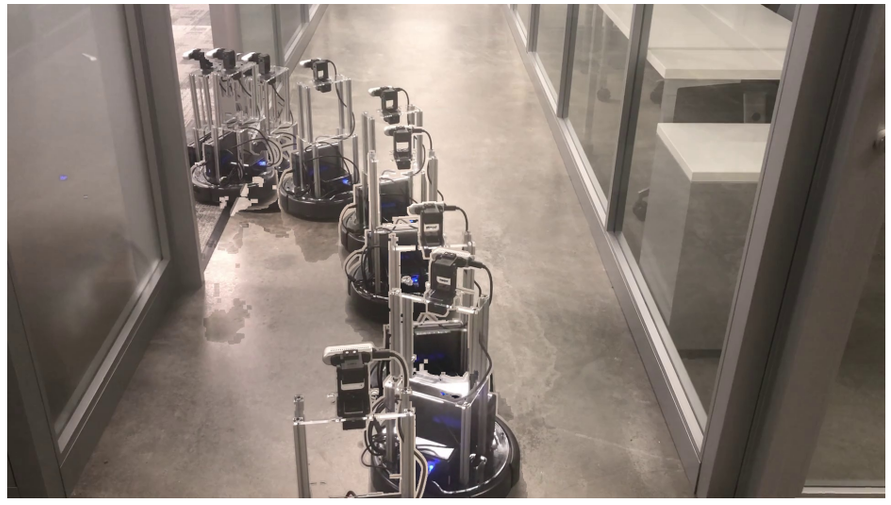
(e) Final 3D scene understanding

People Watching: Human Actions as a Cue for Single View Geometry.
D. F. Fouhey, V. Delaitre, A. Gupta, A. Efros, I. Laptev, and J. Sivic. ECCV 2012.

Social Learning

What all can we learn?

Environmental affordances (first-person videos)



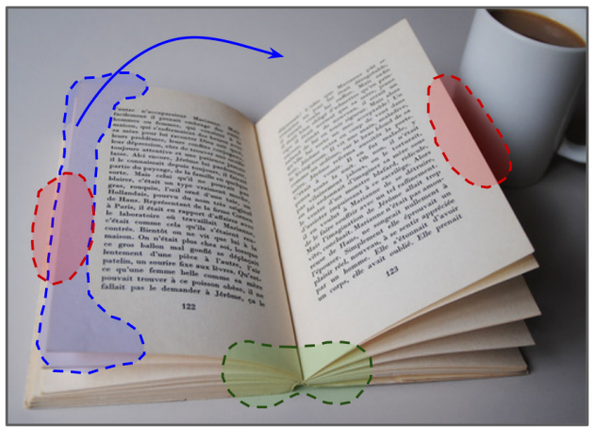
Learning Navigation Subroutines by Watching Egocentric Videos. A. Kumar, S. Gupta, J. Malik. CoRL 2019.

Social Learning

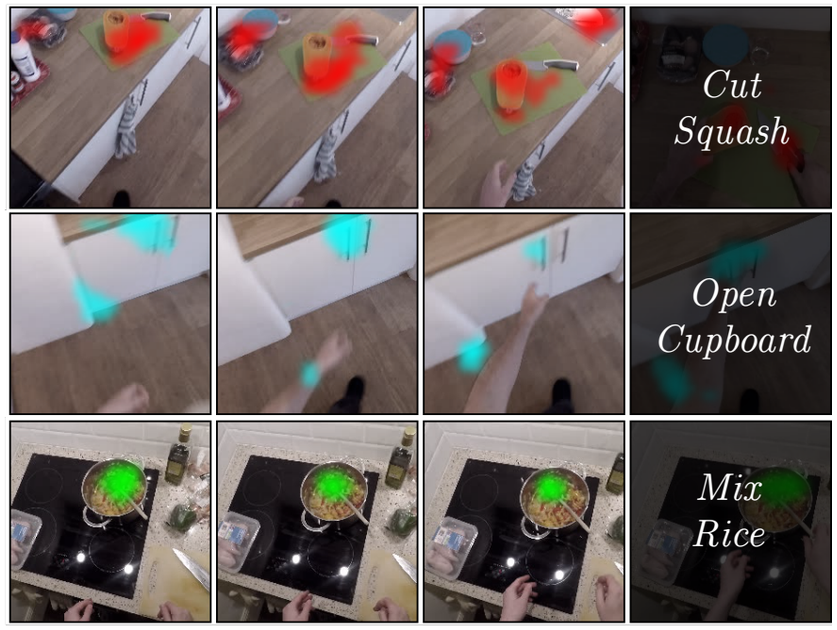
What all can we learn?

Priors for where to interact

Watch natural interactions



"Interaction hotspots"



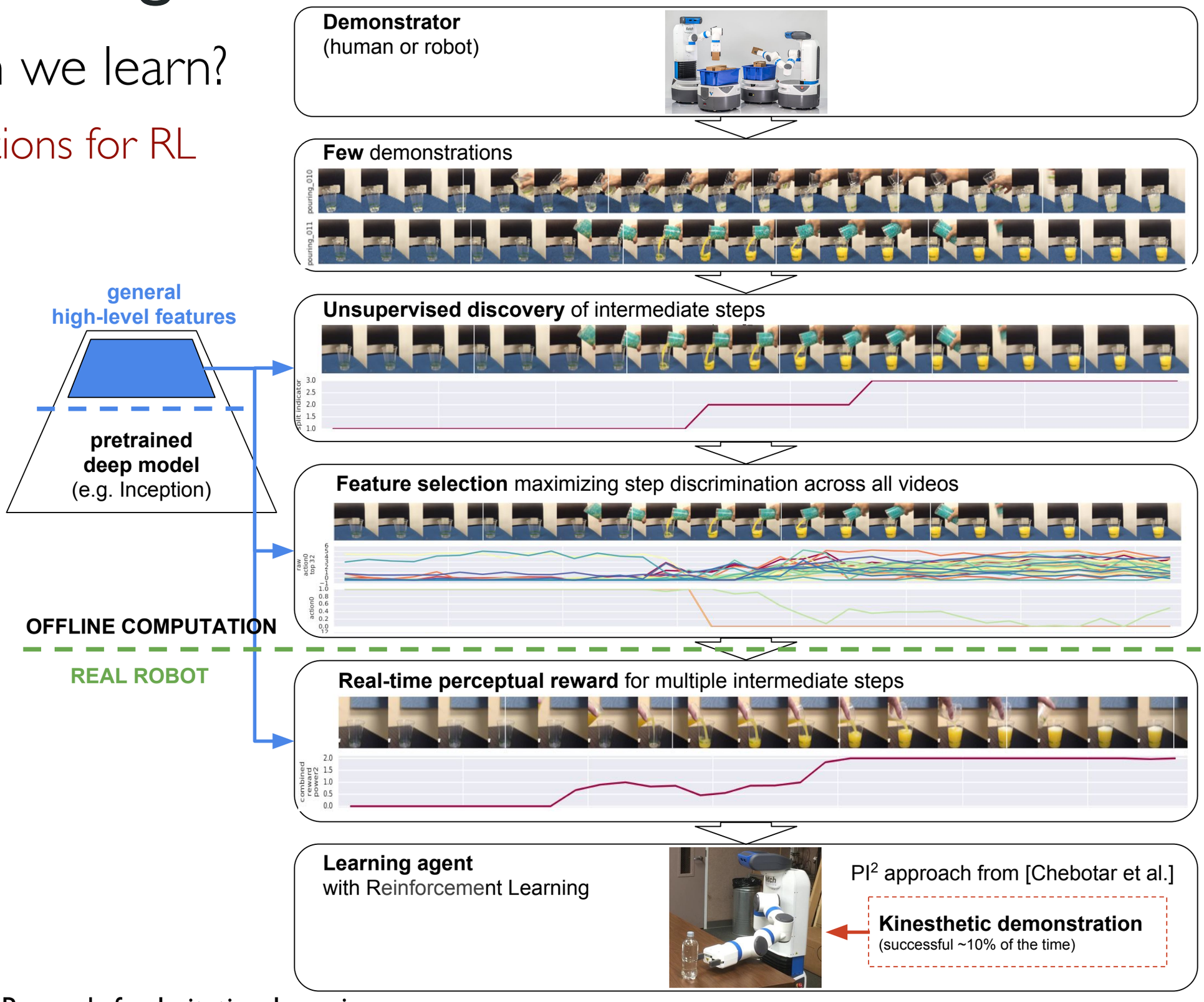
Pre-interaction (inactive object) — *t* → *Interaction occurs*

Grounded Human-Object Interaction Hotspots from Video
Tushar Nagarajan, Christoph Feichtenhofer, Kristen Grauman ICCV 2019.

Social Learning

What all can we learn?

Reward functions for RL



Unsupervised Perceptual Rewards for Imitation Learning

Pierre Sermanet, Kelvin Xu, Sergey Levine. R:SS 2017

Social Learning

Why is it hard?

- Embodiment gap
  - Sensors / actions / capabilities
- Missing action labels
- Only showcase positive data
- Depicted goals may not be known
- More than one way to solve a problem
- Can’t learn things beyond what is shown

Social Learning

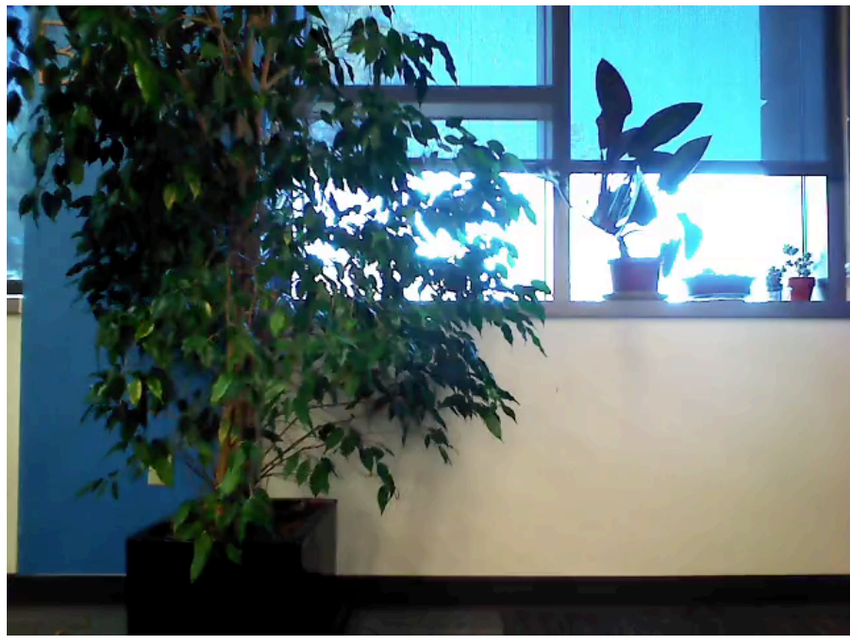
- Learning Navigation Subroutines from Egocentric Videos
- Semantic Visual Navigation by Watching YouTube Videos
- Grasping in the wild: Learning 6-DOF closed-loop grasping from low-cost demonstrations
- Unsupervised perceptual rewards for imitation learning

Learning Navigation Subroutines from Egocentric Videos

Ashish Kumar[1] **Saurabh Gupta**[3] **Jitendra Malik**[1,2]
[1]UC Berkeley [2]Facebook AI Research [3]UIUC
ashish_kumar@berkeley.edu, saurabhg@illinois.edu, malik@eecs.berkeley.edu

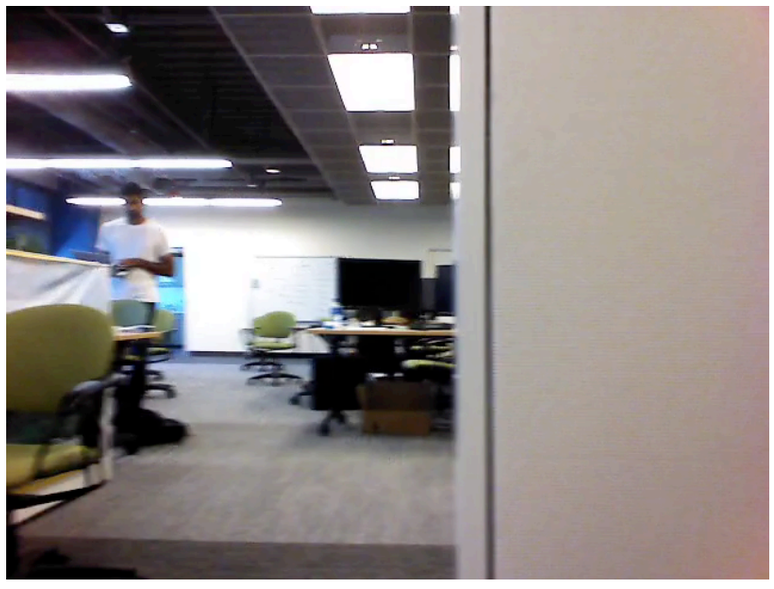
Learn Skills that enable a Robot to Move Around in Novel Environments

Robot w/camera

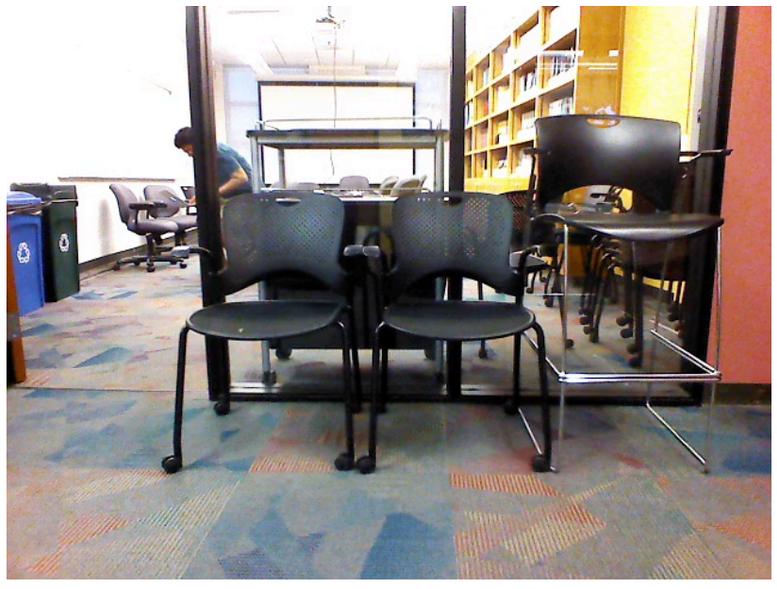


In novel environment

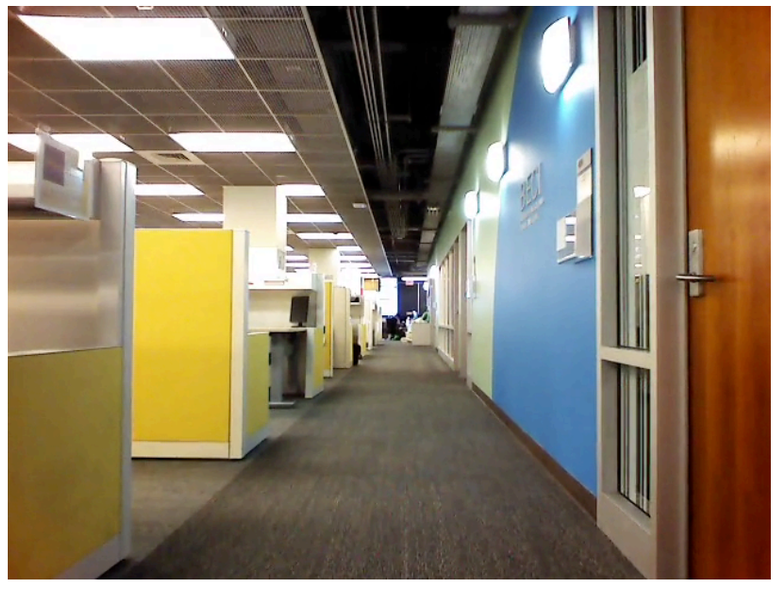
Skill: Go around obstacles



Skill: Come out of cubicles

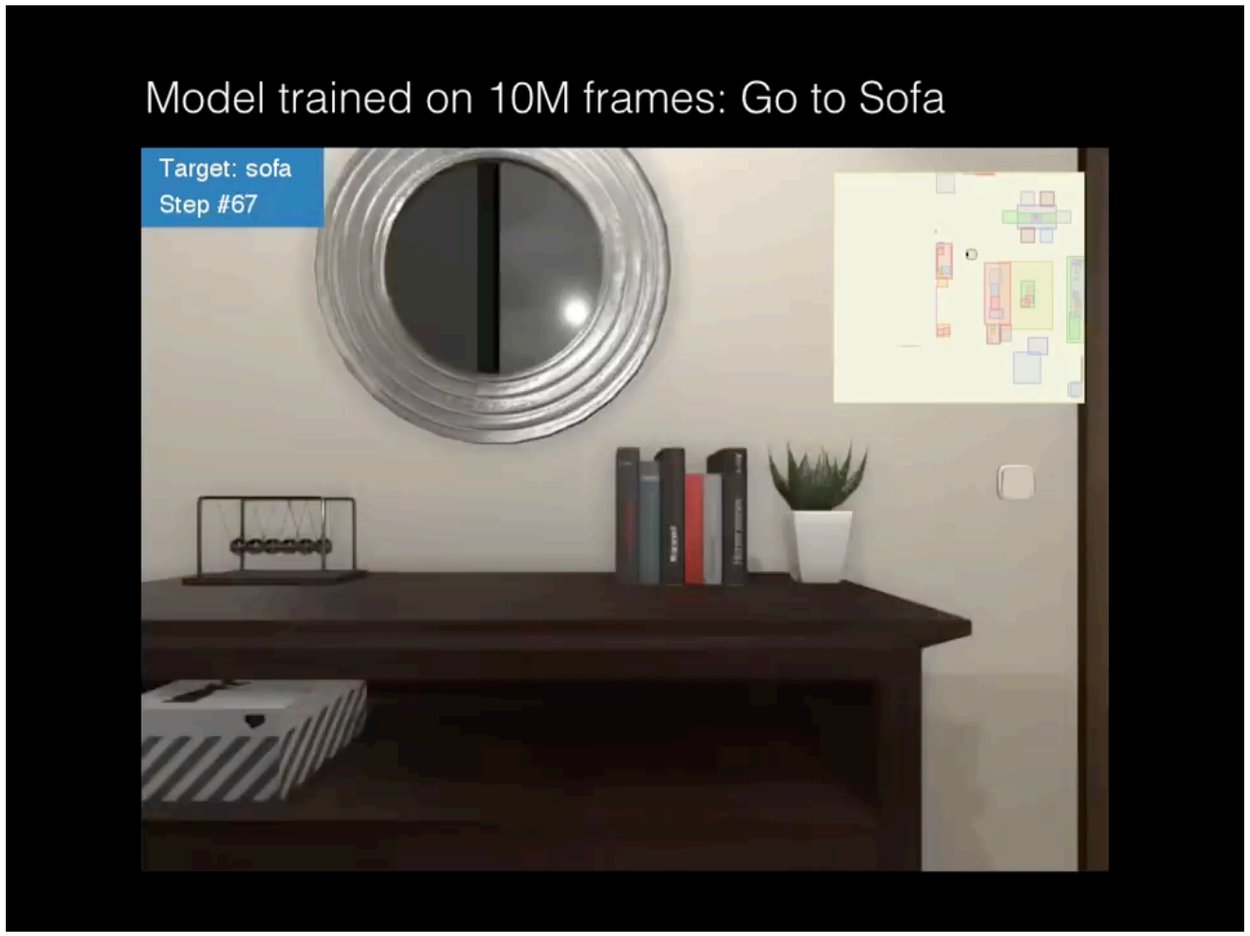


Skill: Go through door



Skill: Go down hallway

Existing Work: Learning by Interaction



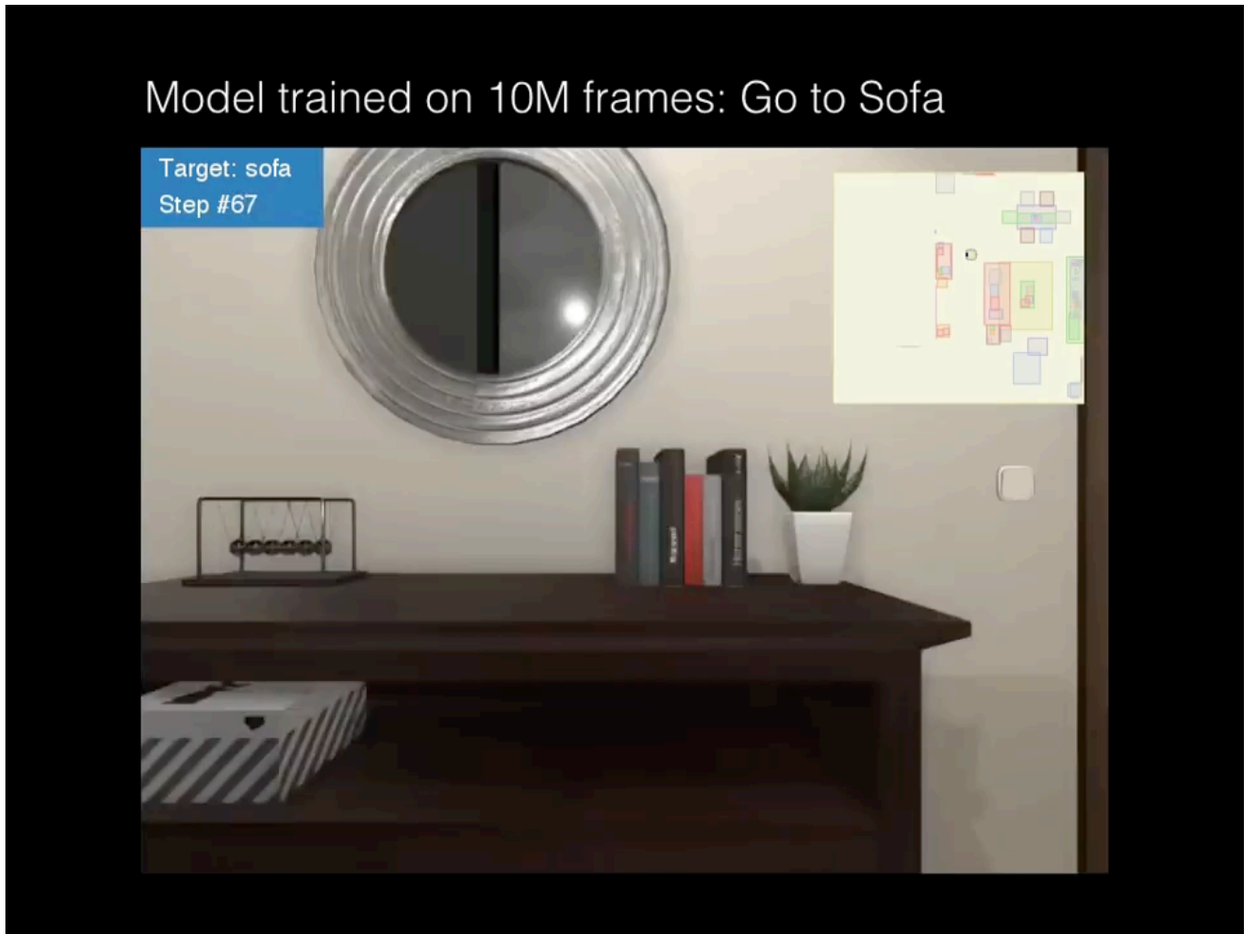
Learn using rewards

- Hard to define reward functions.
- Learned skills are hard to repurpose.
- *All training signal comes from direct interaction with environment.*

In Contrast, Learning in Computer Vision:

Big passive data gathered on the Internet

Best of Both Worlds?

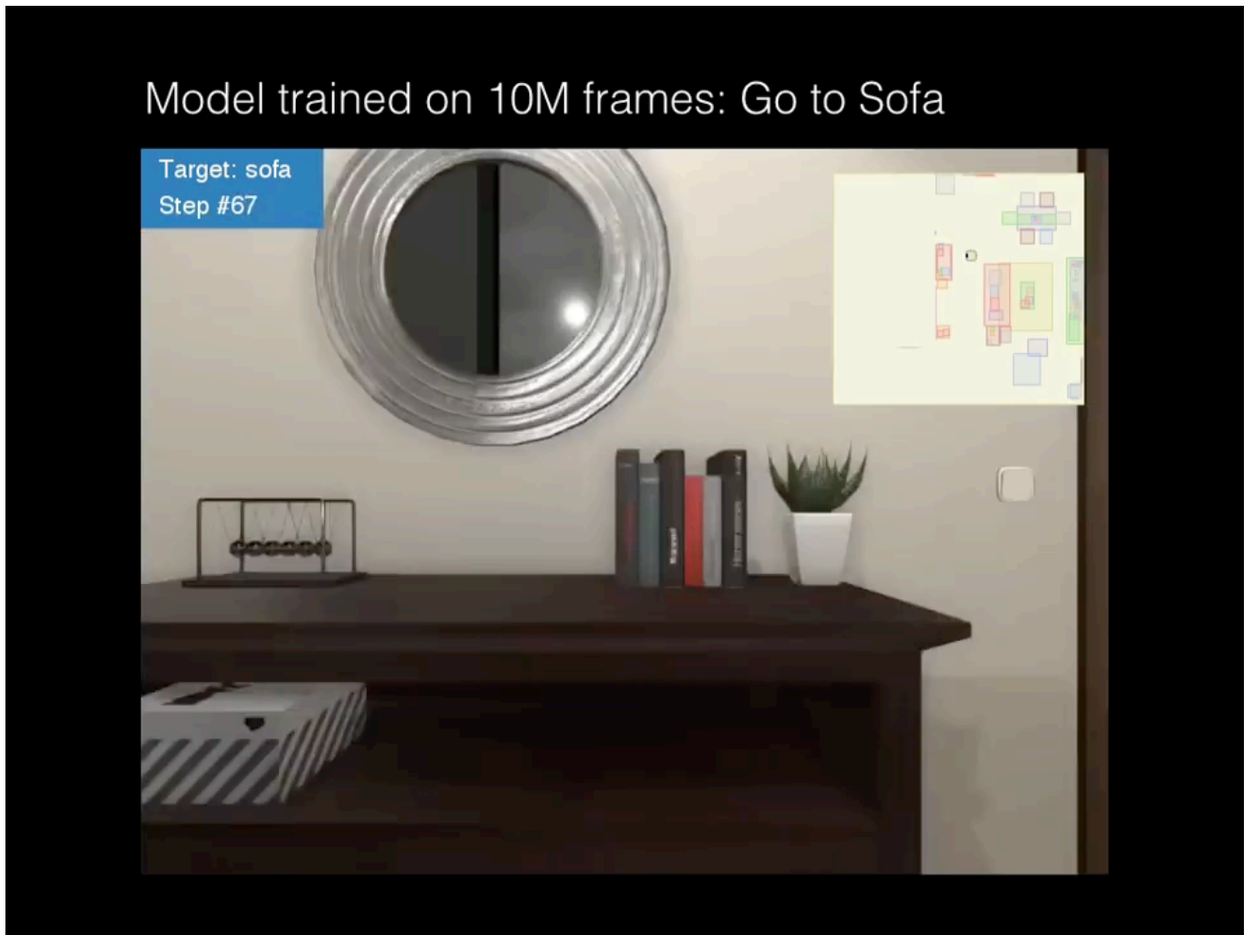


Expensive Interaction Data

+

Cheap Internet Data

Best of Both Worlds?

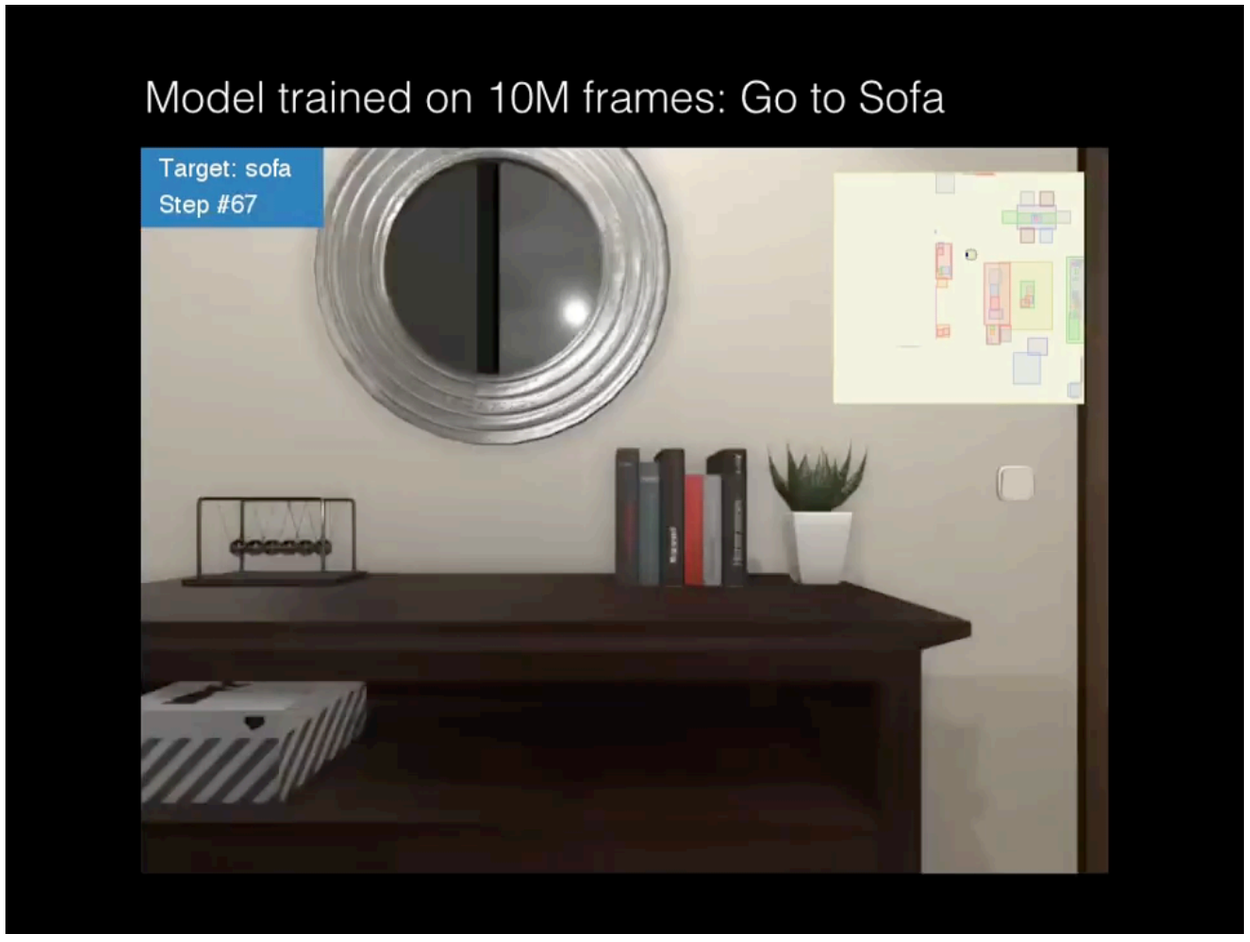


Interaction data: Random Interaction Data

+

Cheap Internet Data: First-person Videos from Youtube

Best of Both Worlds?

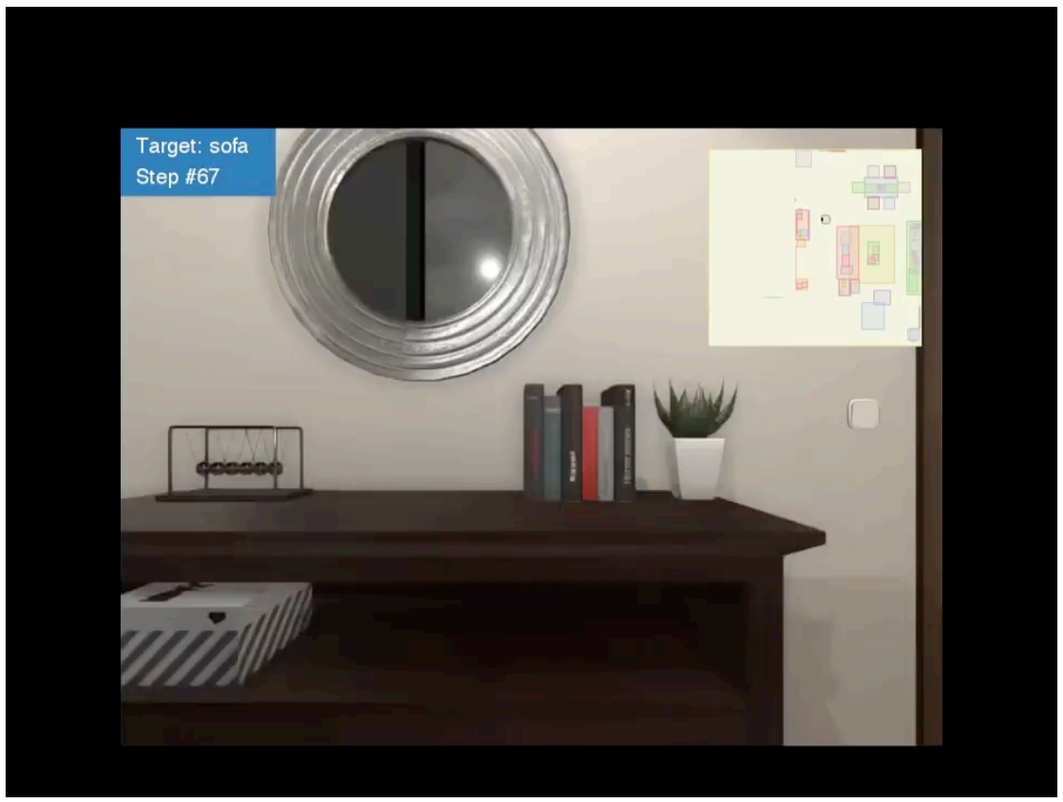


Interaction data: Random Interaction in Simulators

+

Cheap Internet Data: First-person renderings from Simulators

Use Internet Data to Scale-up Policy Learning



Active Interaction

+

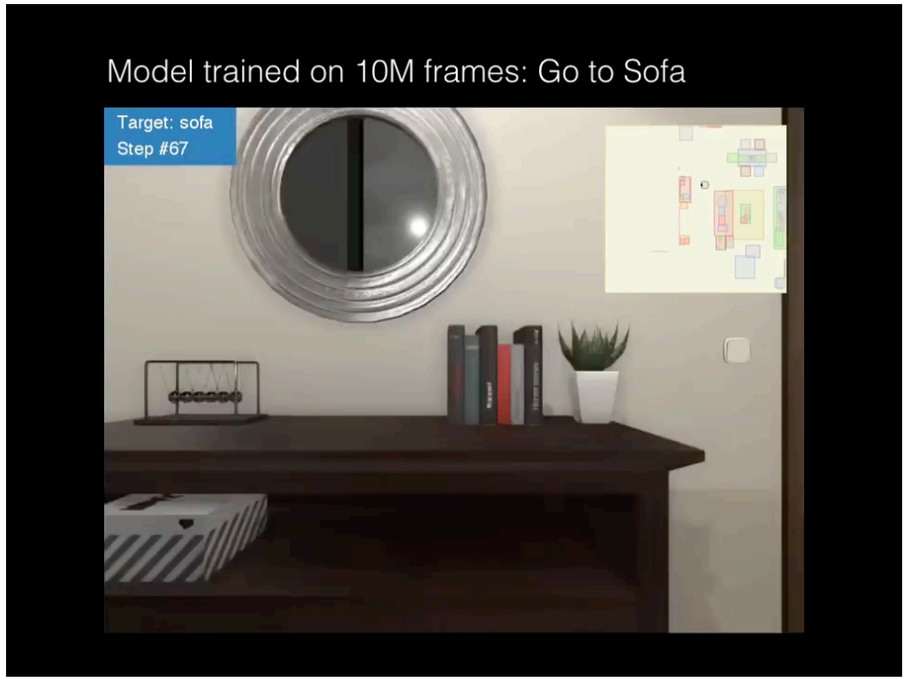
Egocentric Videos

1. Learn Model to Interpret Videos

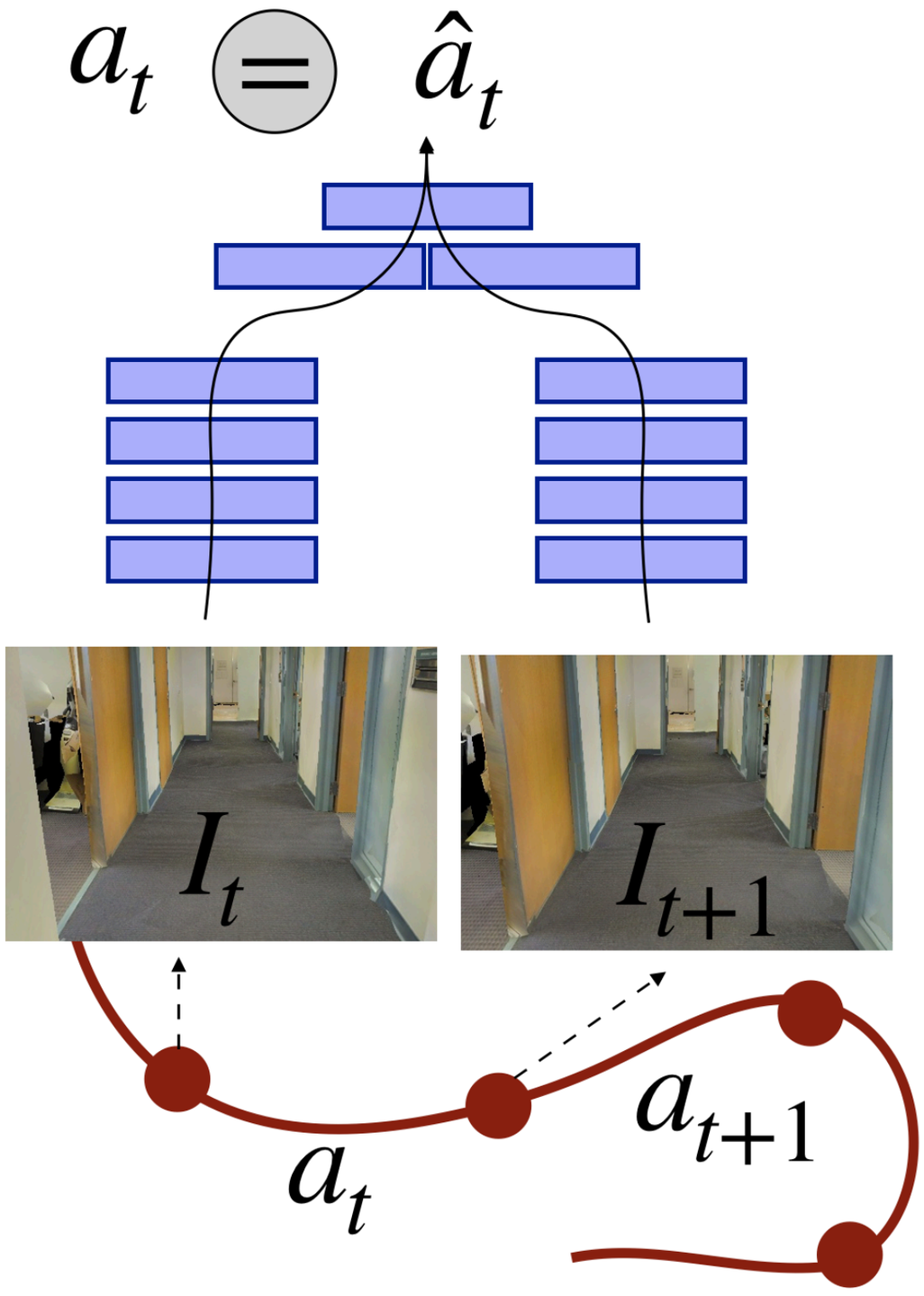
2. Use Model to Pseudo-Label Egocentric Videos with Actions

3. Learn Skills via Supervised Learning

Learning Navigation Subroutines by Watching Egocentric Videos. CoRL 2019. A. Kumar, S. Gupta, J. Malik



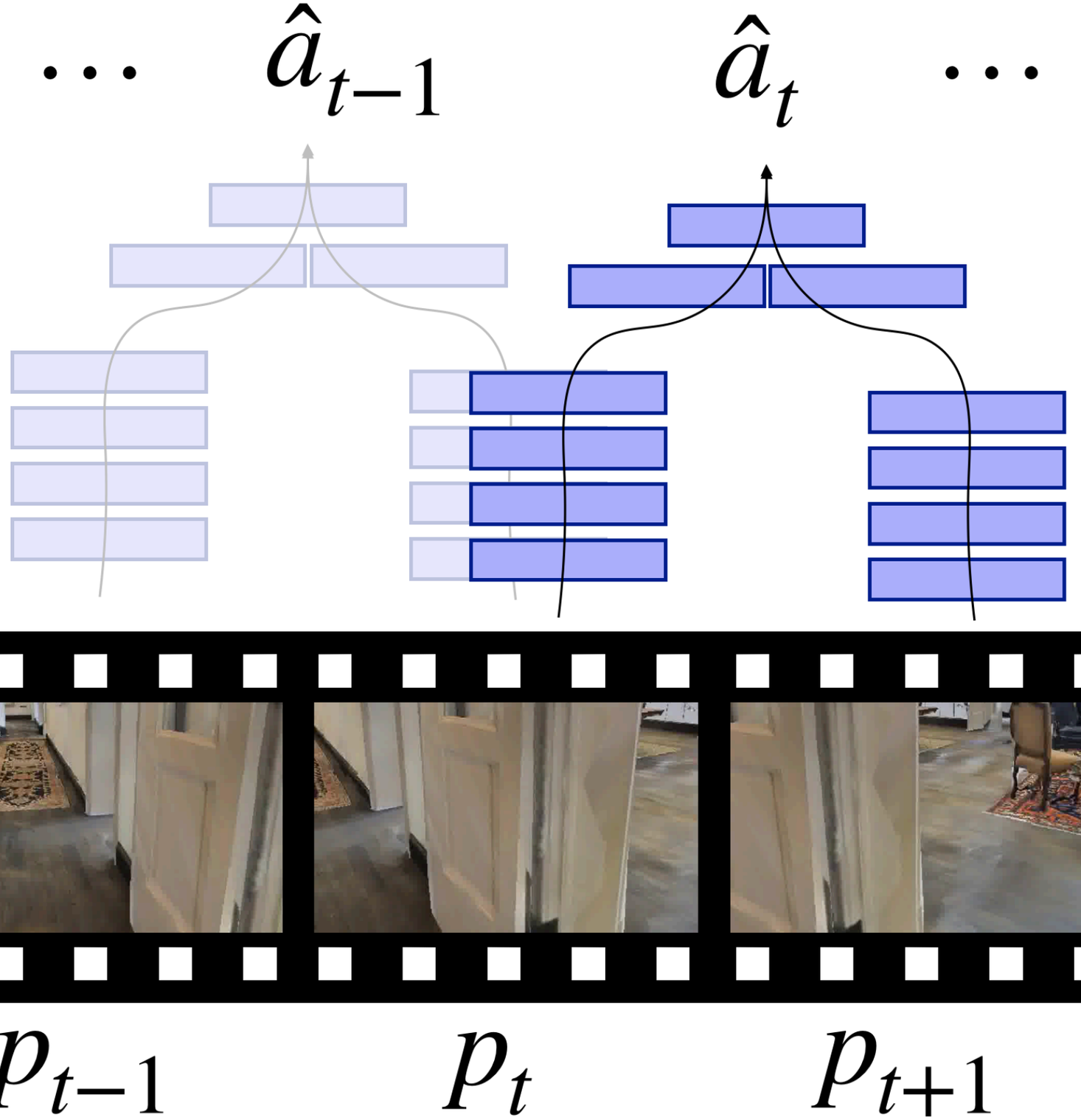
Active Interaction



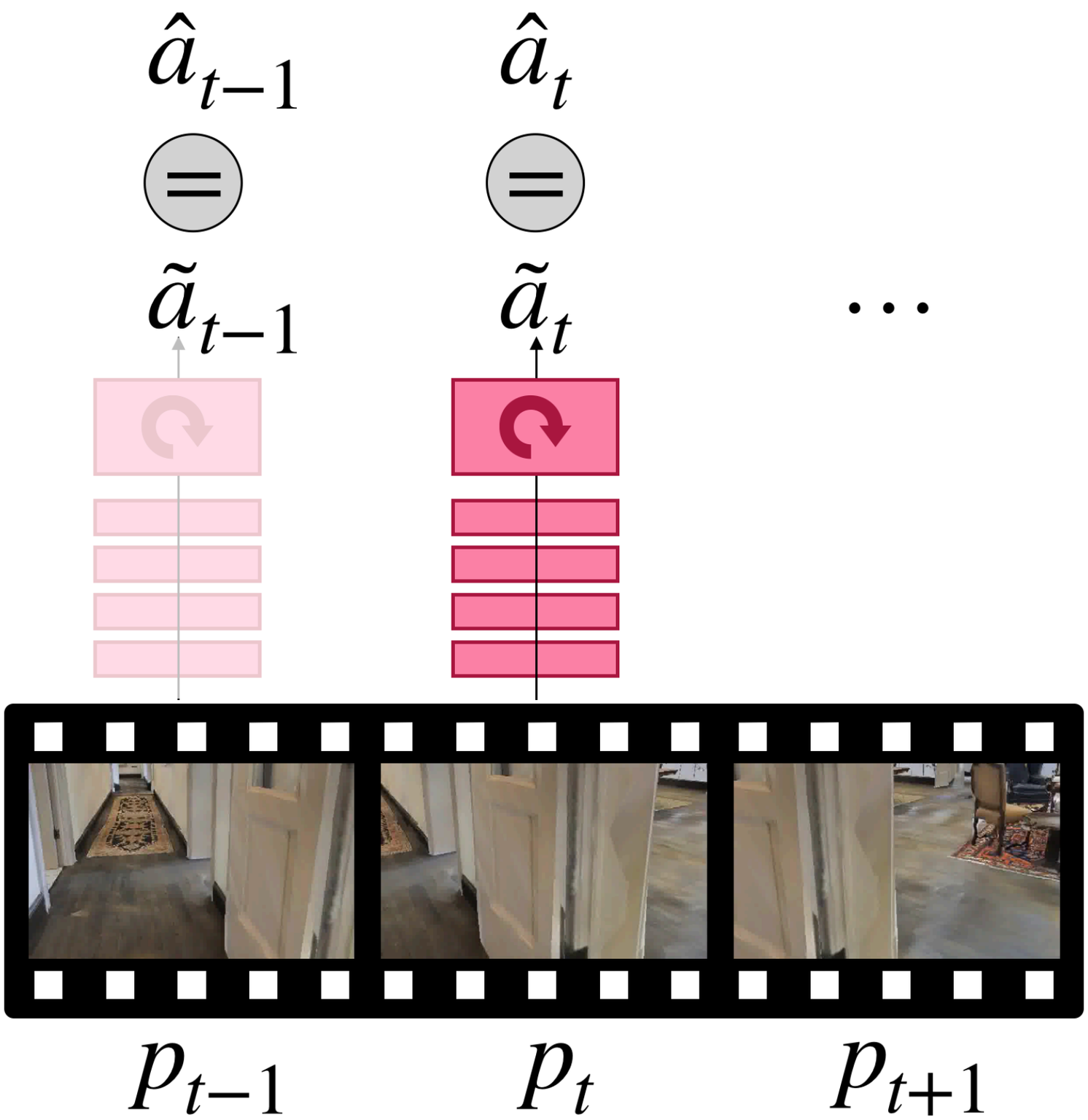
(a) Learning Inverse Model from Random Interaction

Learning Navigation Subroutines by Watching Egocentric Videos. CoRL 2019. A. Kumar, S. Gupta, J. Malik

Egocentric Videos



(b) Pseudo-labeling Egocentric Videos



(c) Operator Learning

Only used during training

Cross Entropy Loss

Cross Entropy Loss

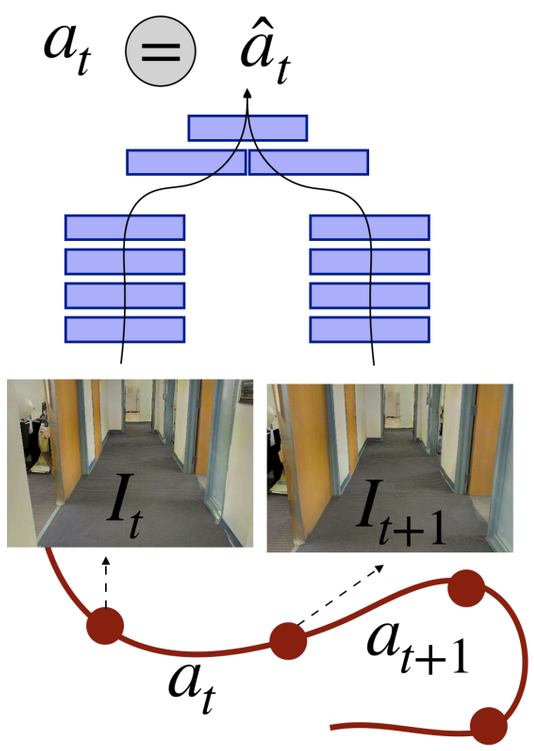
Subroutines

1D CNN

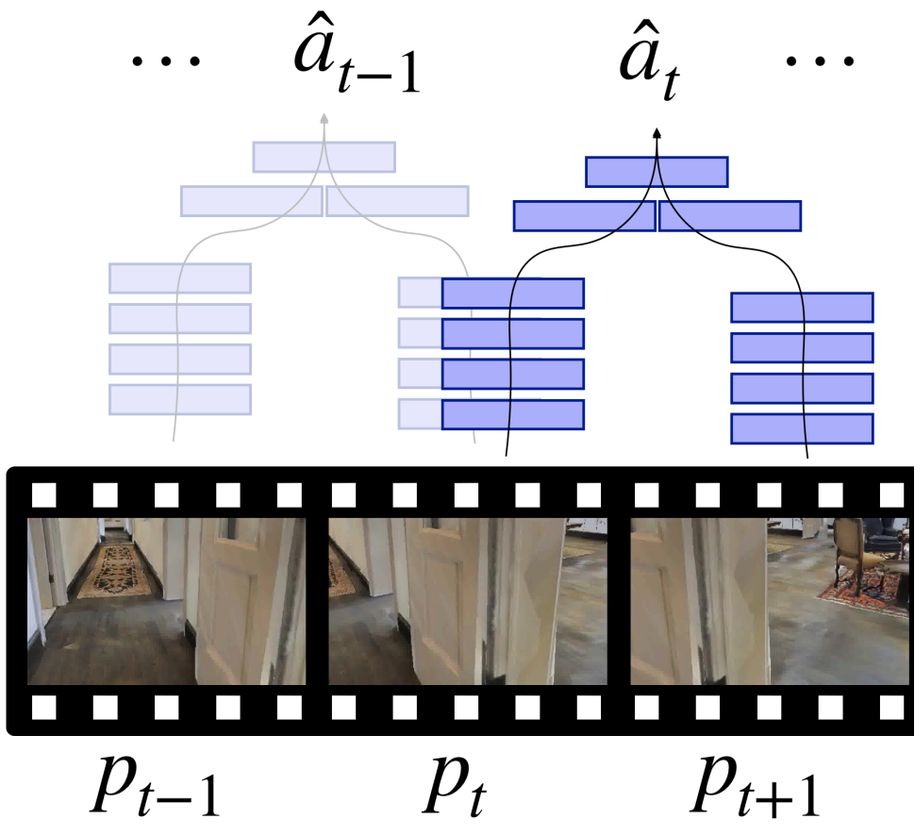
Affordance Model

Pseudo-action Sequence

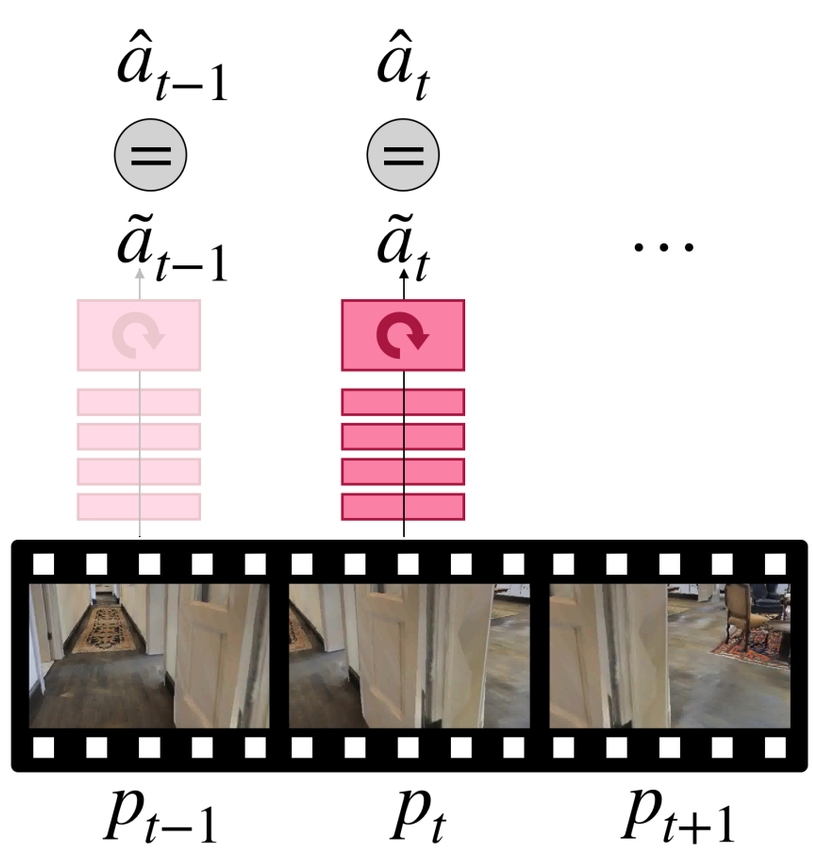
Video



(a) Learning Inverse Model from Random Interaction



(b) Pseudo-labeling Egocentric Videos

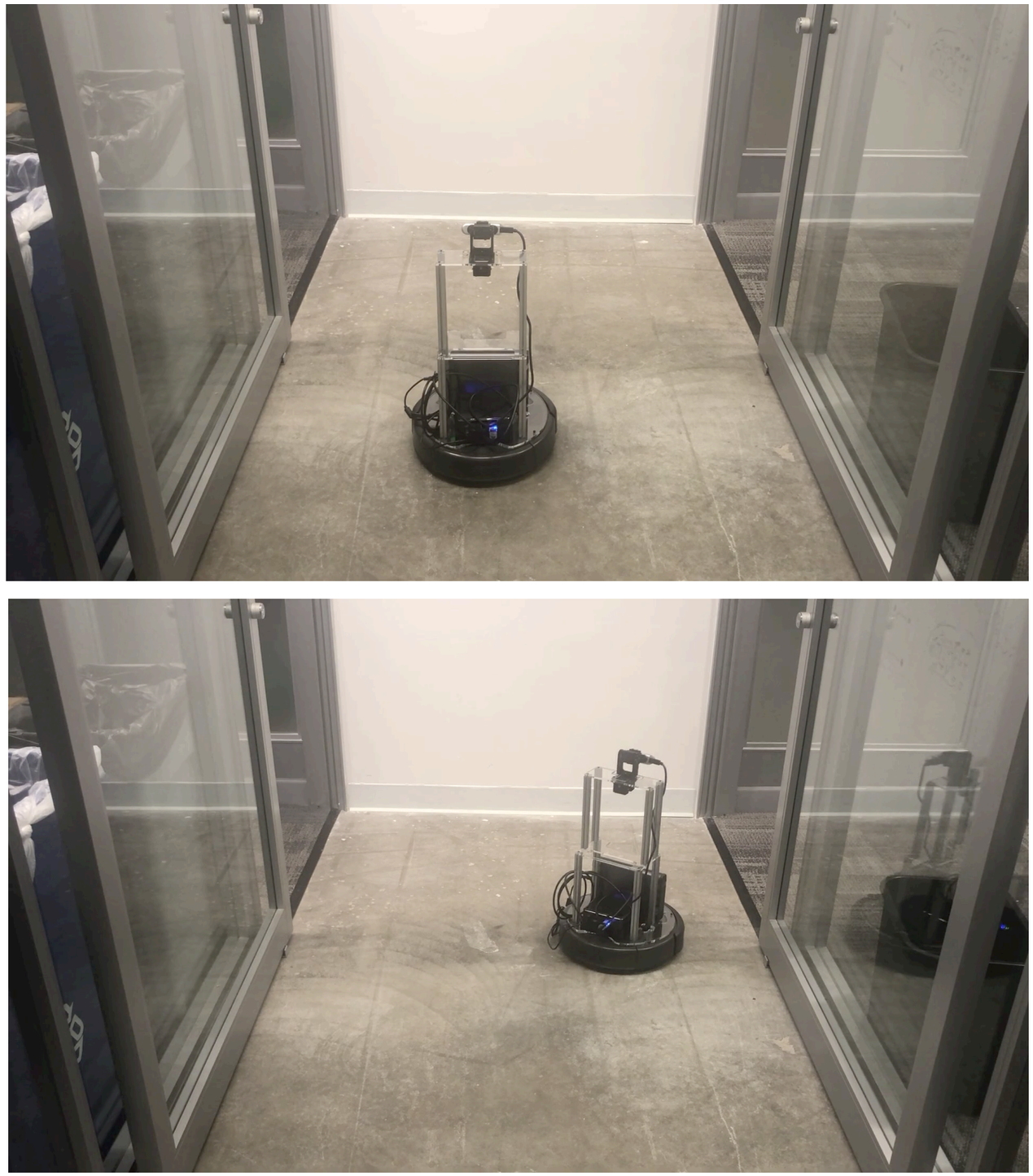


(c) Operator Learning

Learned Skills

SubR3 - Turns Left

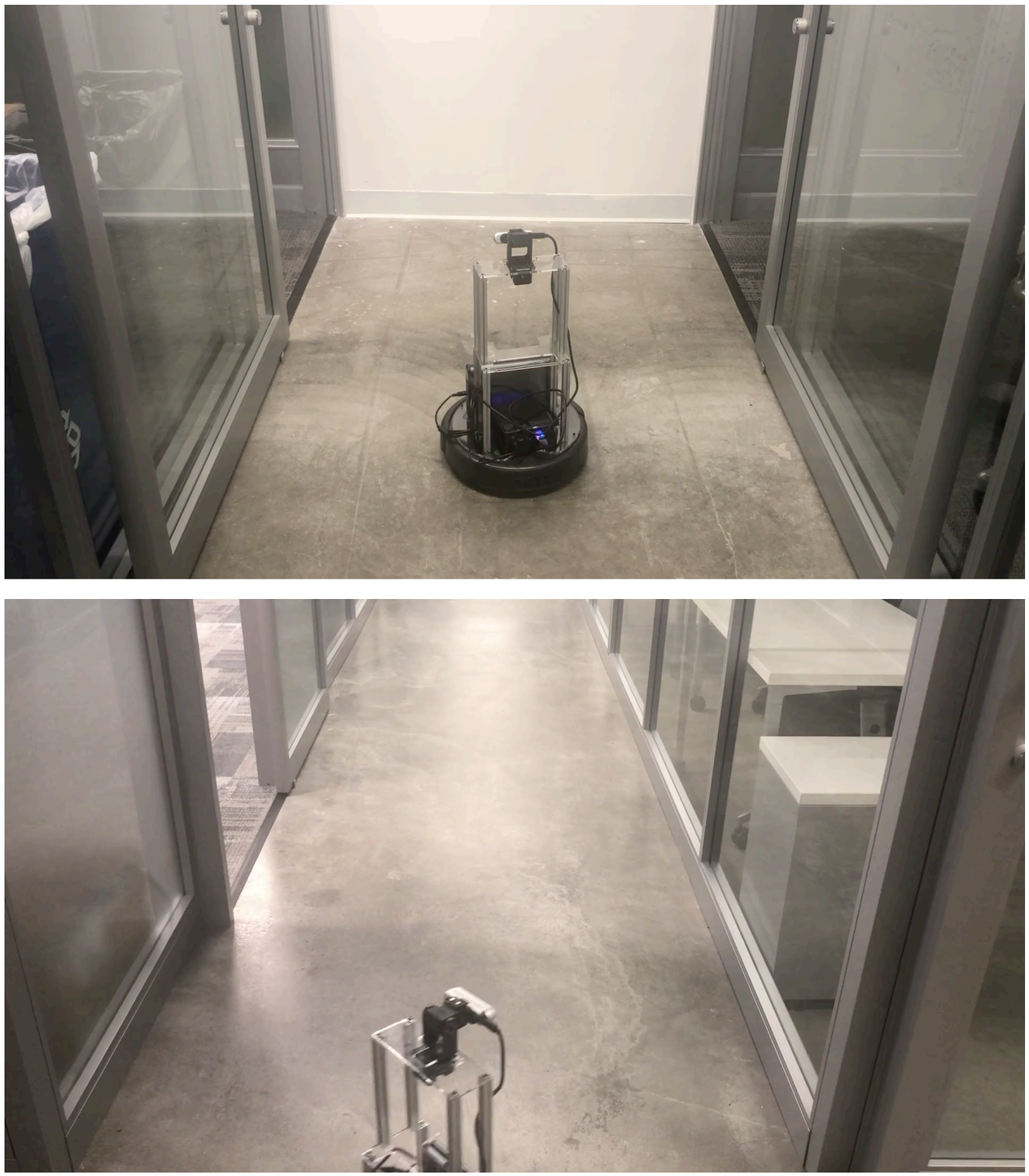
SubR1 - Turns Right

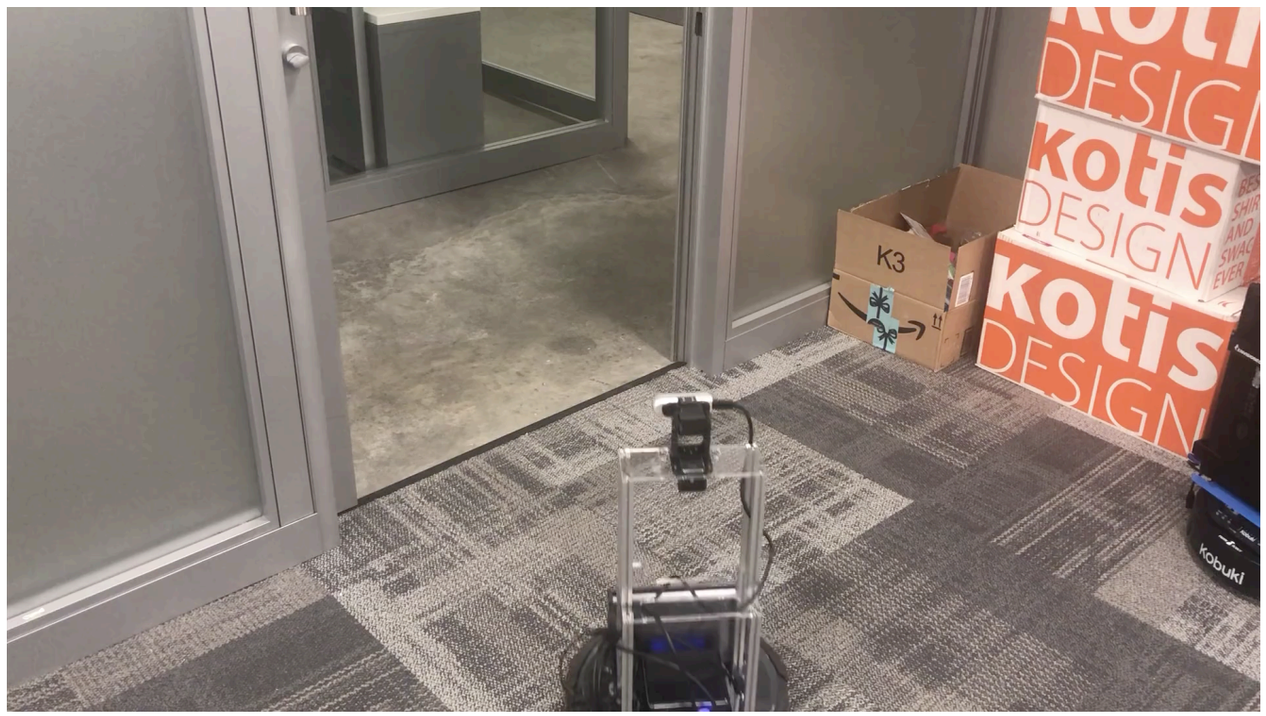


Learning Navigation Subroutines by Watching Egocentric Videos. CoRL 2019. A. Kumar, S. Gupta, J. Malik

Learned Skills

SubR3 - Turns Left





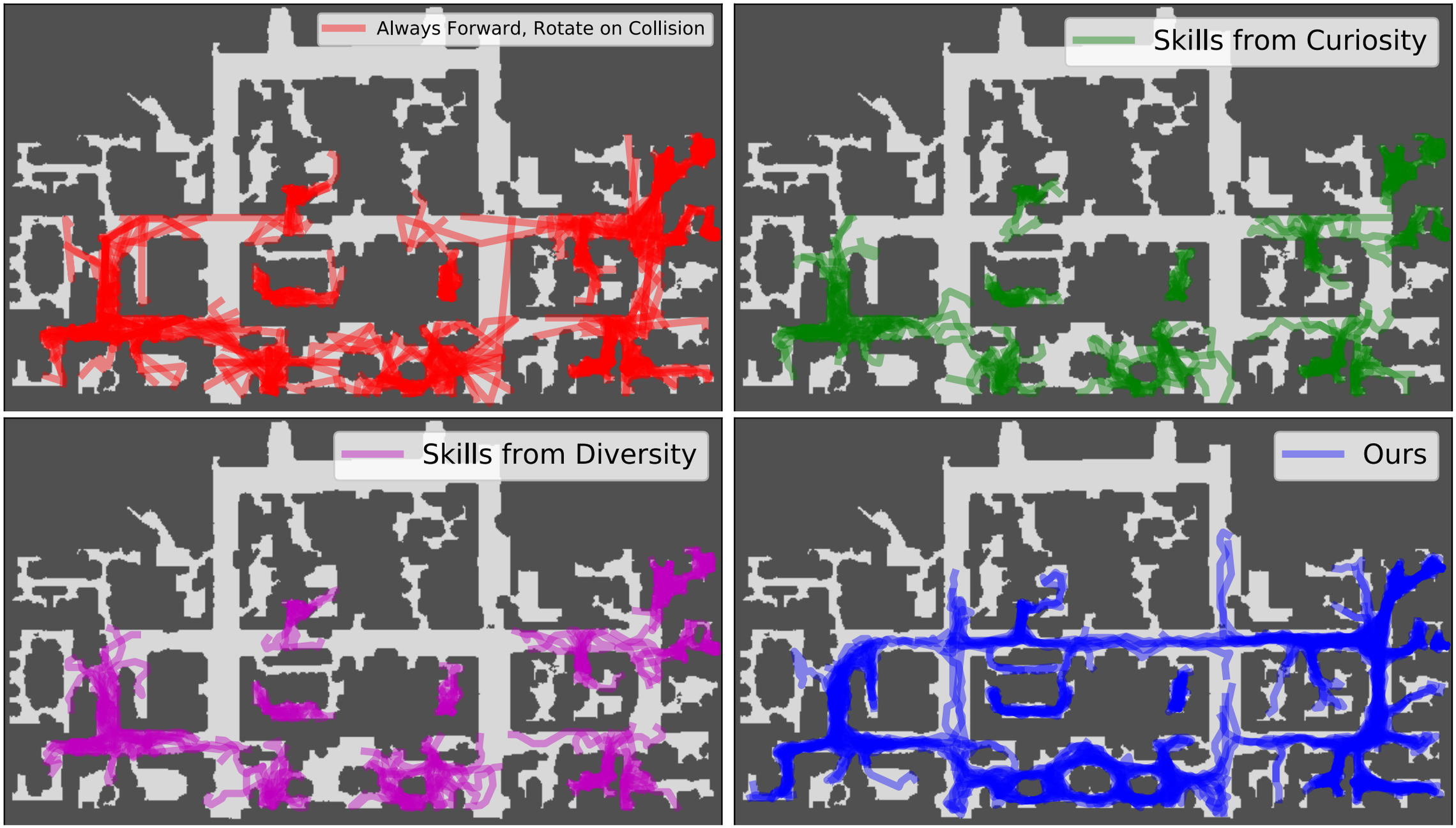
Learning Navigation Subroutines by Watching Egocentric Videos. CoRL 2019. A. Kumar, S. Gupta, J. Malik

Using Subroutines and Affordances

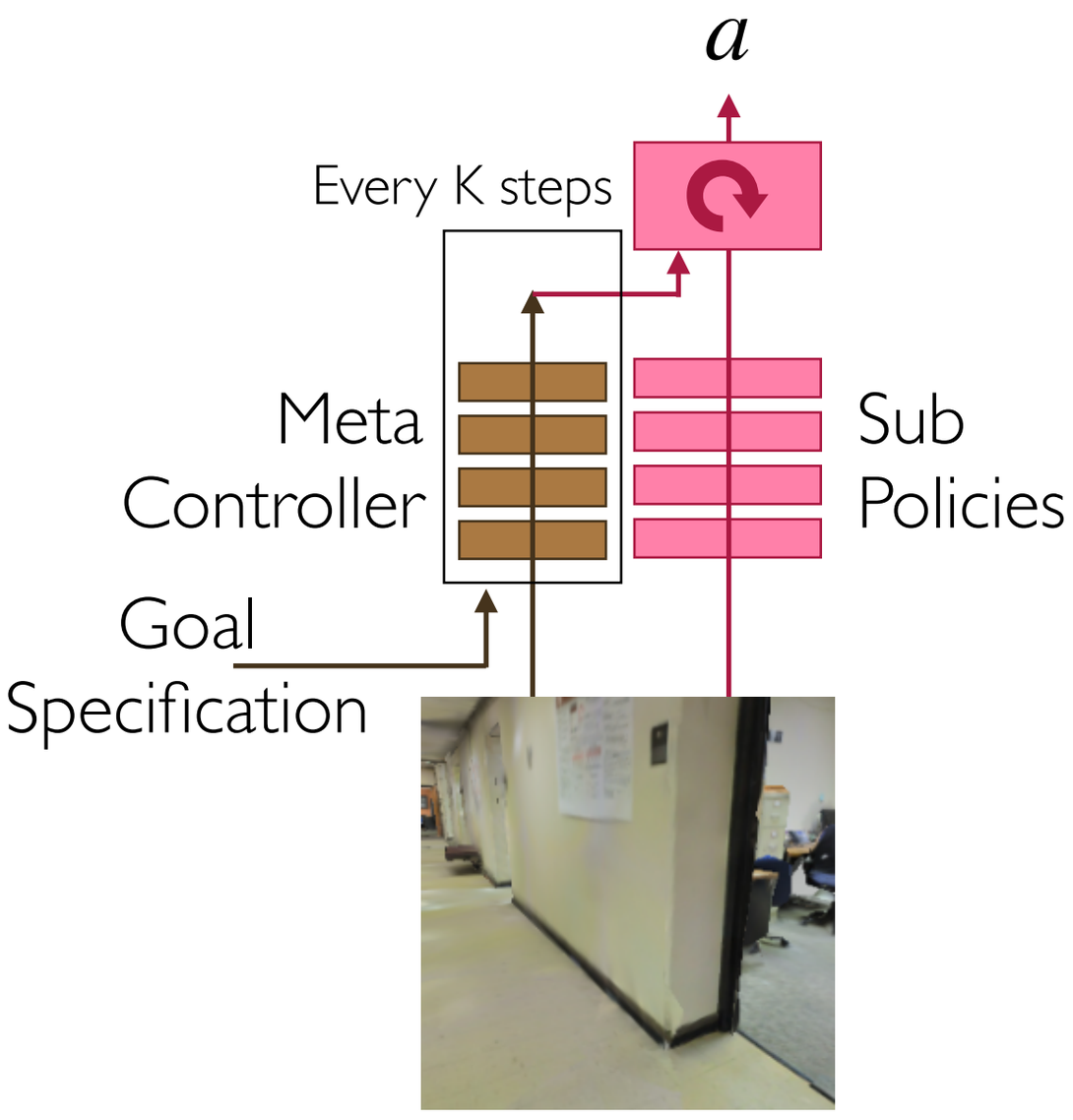
Method	# Samples	ADT	Max. Dist.	Collision Rate (%)
Random	0	0.96	4.34	62.5
Forward Bias Policy	0	0.66	7.19	80.2
Always Forward, Rotate on Collision	0	0.62	8.20	66.3
Skills from Diversity [13]	$10M$	0.79	4.90	64.0
Skills from Curiosity [27]	$10M$	0.83	4.36	61.3
Our (Exploration via Subroutines)	$45K$	**0.34**	**11.06**	**12.0**

B. Eysenbach, A. Gupta, J. Ibarz, and S. Levine. ICL 2019. **Diversity is all you need: Learning skills without a reward function.**
D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell. ICML 2017. **Curiosity-driven exploration by self-supervised prediction.**

Exploration Comparisons



Using Subroutines and Affordances for Hierarchical RL

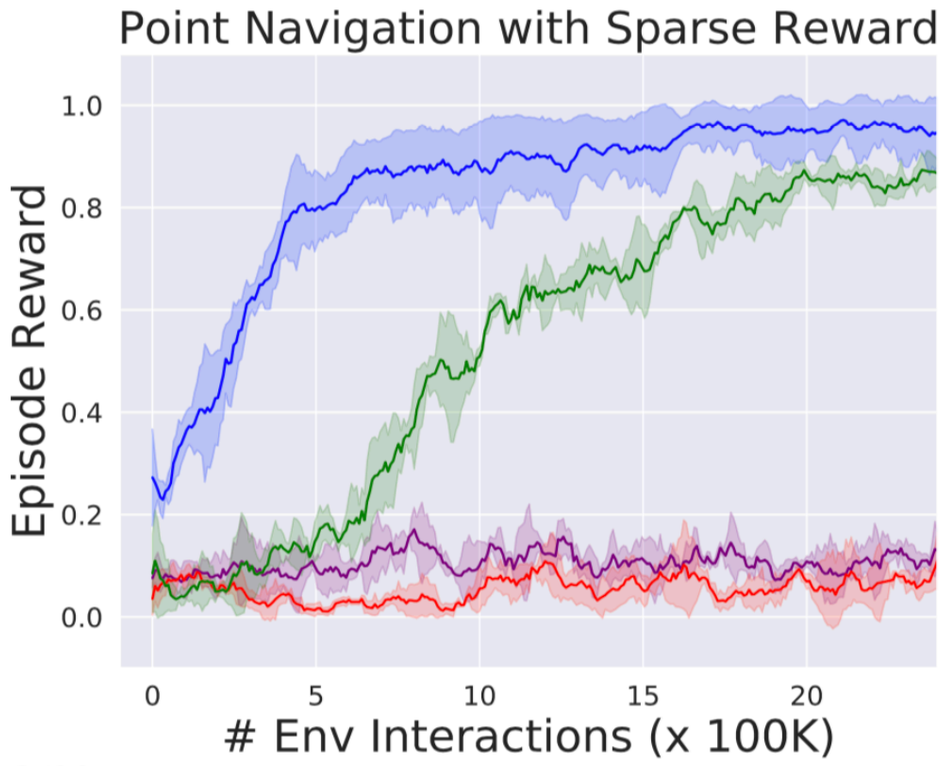


1. Use Subroutines as sub-policies.
2. Use Affordance Model to initialize meta-controller and guide meta-controller towards feasible sub-policies.

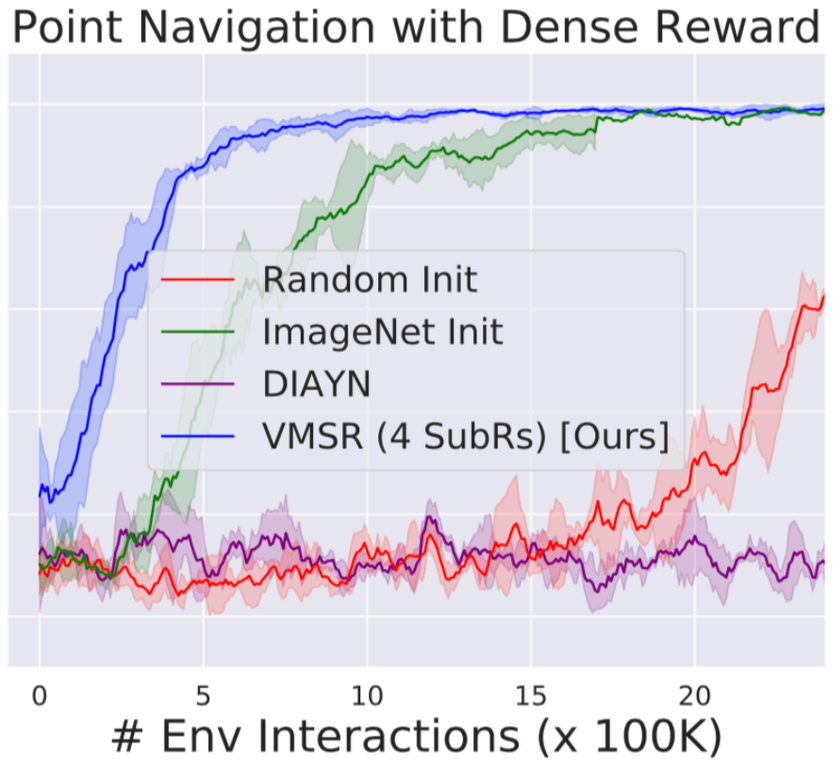
Using Subroutines and Affordances for Hierarchical RL

PointGoal Go To (x,y)

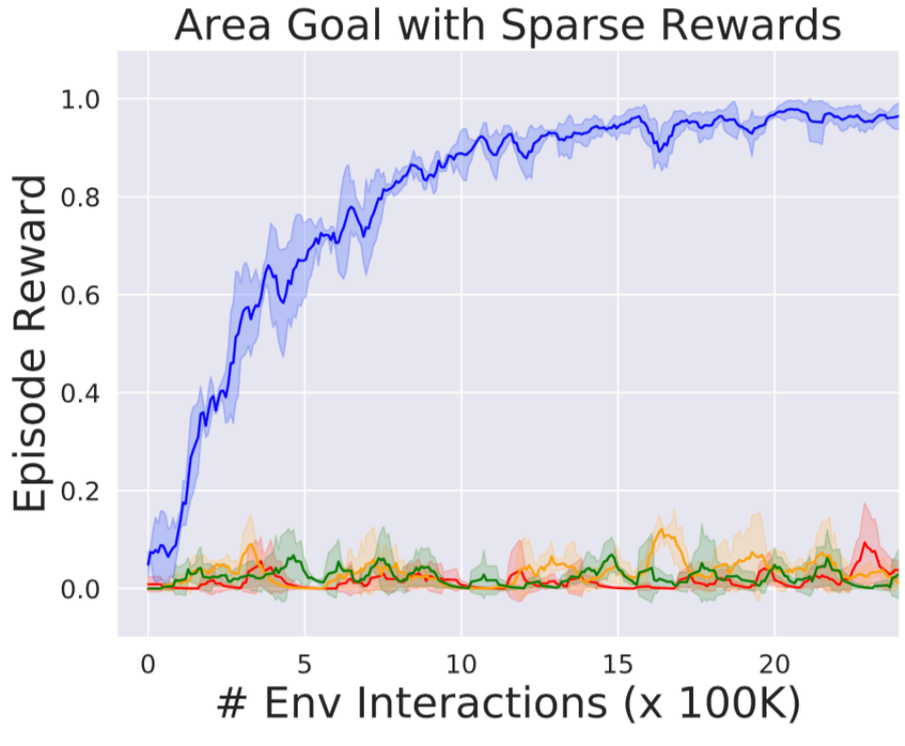
Sparse

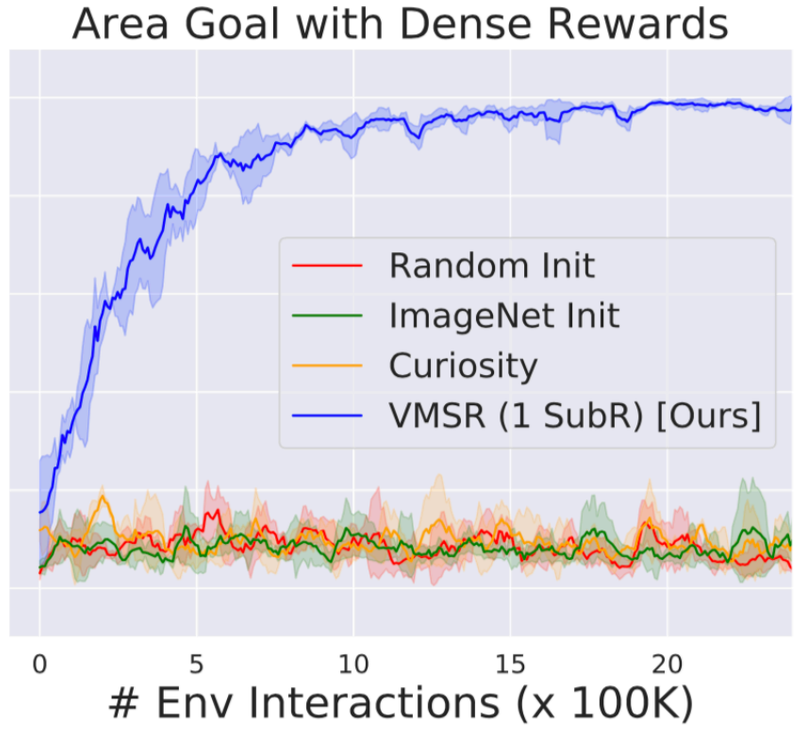


Dense



AreaGoal - Find Bathroom



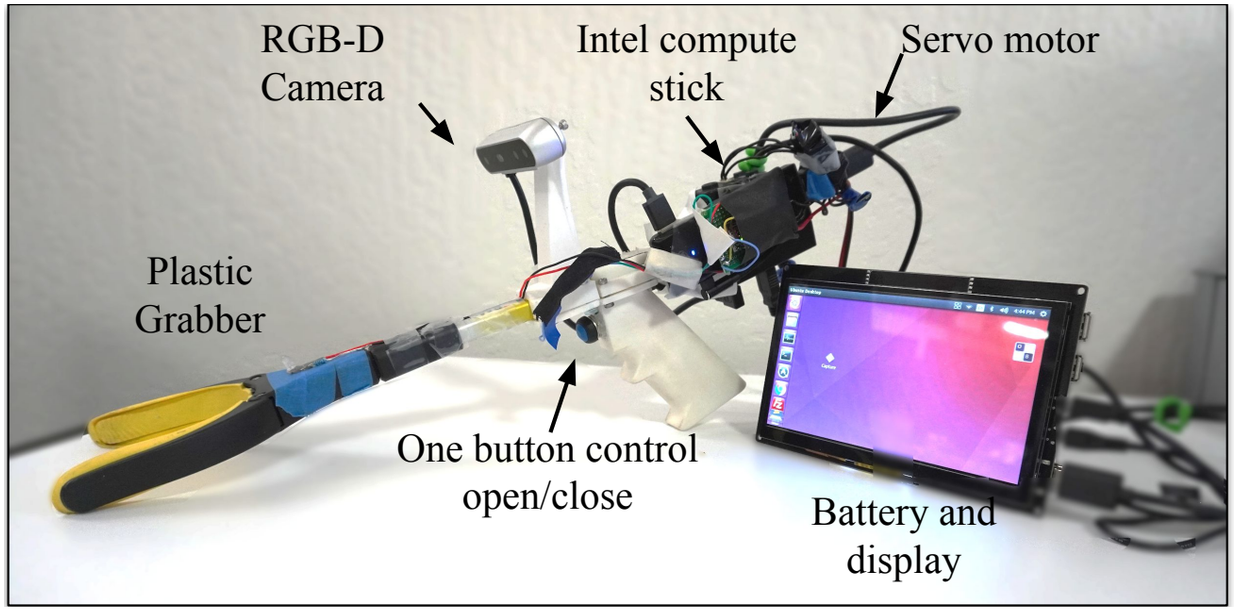


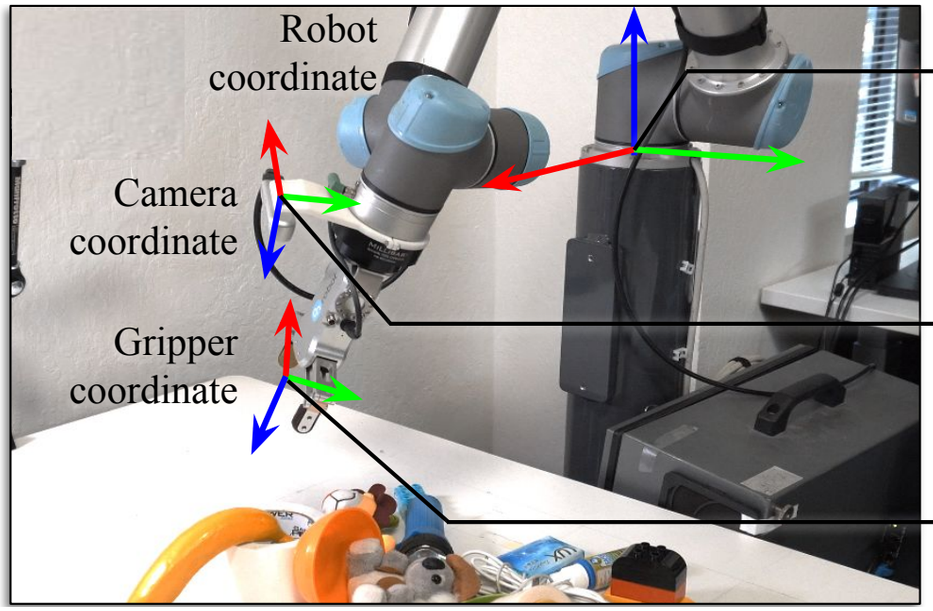
Grasping in the Wild: Learning 6DoF Closed-Loop Grasping from Low-Cost Demonstrations

Shuran Song[1,2] Andy Zeng[2] Johnny Lee[2] Thomas Funkhouser[2]

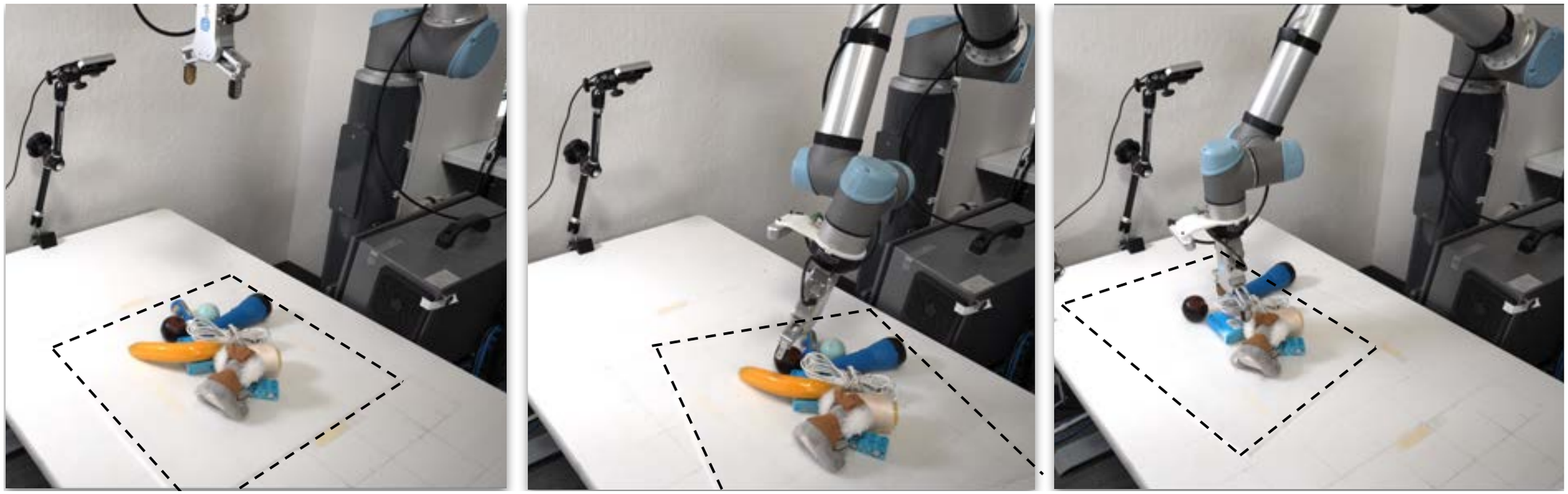
Grasping in the Wild: Learning 6DoF Closed-Loop Grasping from Low-Cost Demonstrations

Shuran Song[1,2] Andy Zeng[2] Johnny Lee[2] Thomas Funkhouser[2]

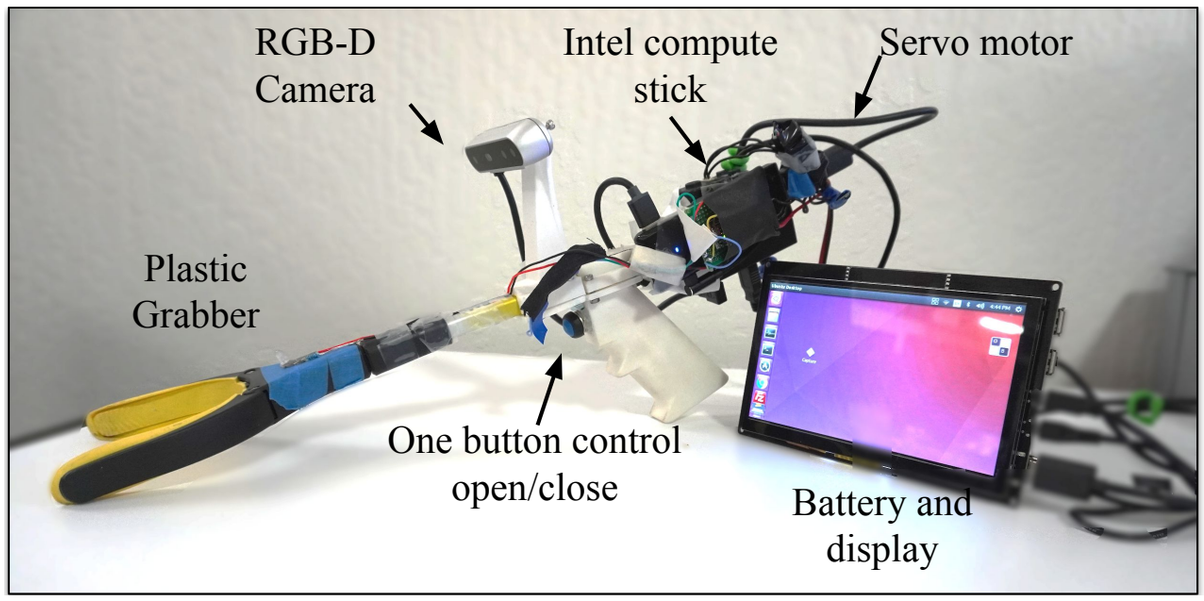


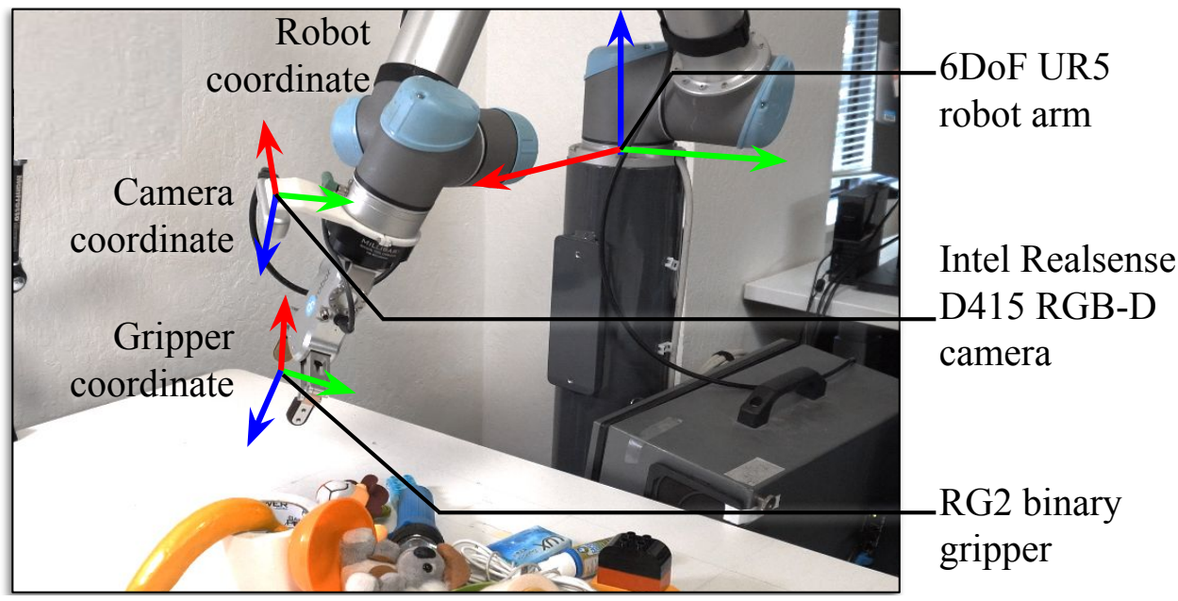


- Tackle the problem of *closed-loop* 6DOF grasping



- Design a capture tool that simplifies gathering data for manipulation:





- Use collected demonstrations to finetune policy on real robot

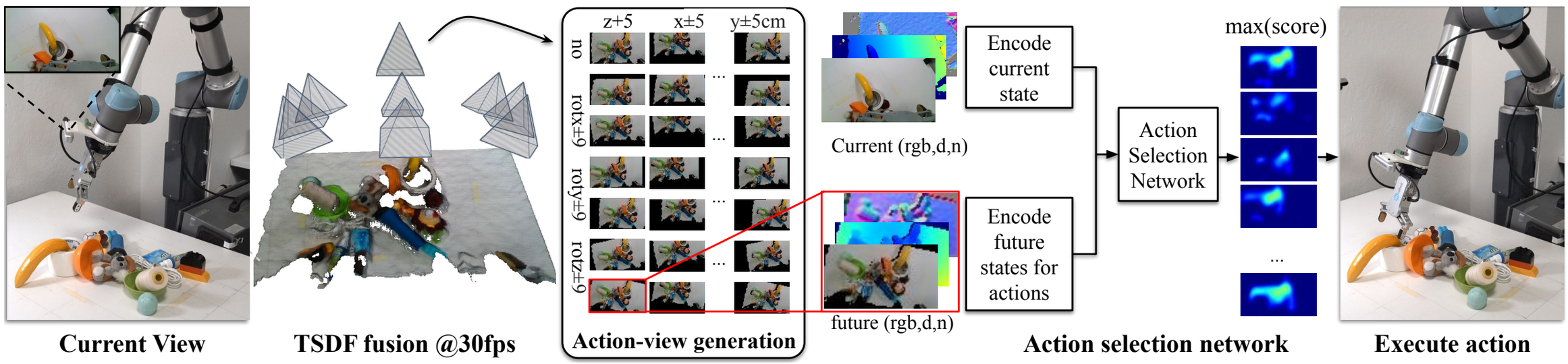
Collected Data

8000 Trajectories, 12 hours of data

Approaching Trajectories

Other Grasping Examples

Inference and Learning



Current View **TSDF fusion @30fps** **Action selection network** **Execute action**

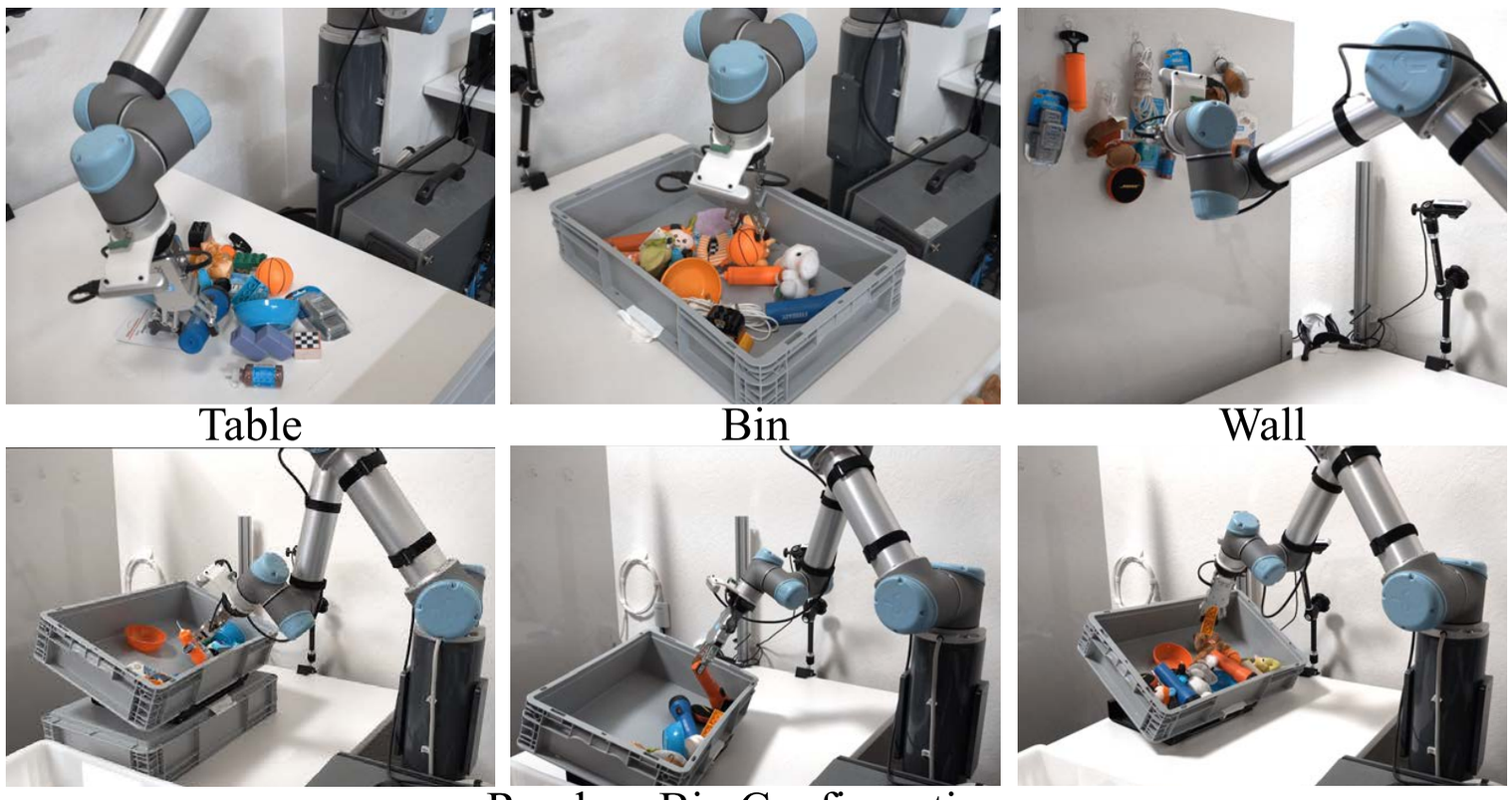
- Action space?
- (state, next state) value function (obtained via Q-learning)
- Generate action views to generate next state (via TSDF fusion + render)
- Convolutional Q-functions

- Q-learning
- Bootstrap using demonstrations
- Positive data only: synthesize negative trajectories

Experiments

TABLE II
TESTING ON DIFFERENT SCENE CONFIGURATIONS (MEAN %).

	Tabletop	Bin	Wall	Random
pretrain only	76	66	78	62
+finetune	92	82	89	76



Experiments

COMPARISON TO STATE-OF-THE-ART METHODS (MEAN %).

Method and Setup	Static Scenes	Dynamic Scenes	Time
GG-CNN [18]	87 ± 7	81 ± 8	19ms
Viereck et al. [17]	89	77	0.2s
Zeng et al. [2]	90 ± 6	-	-
Ours	92±5	88 ± 8	0.18s

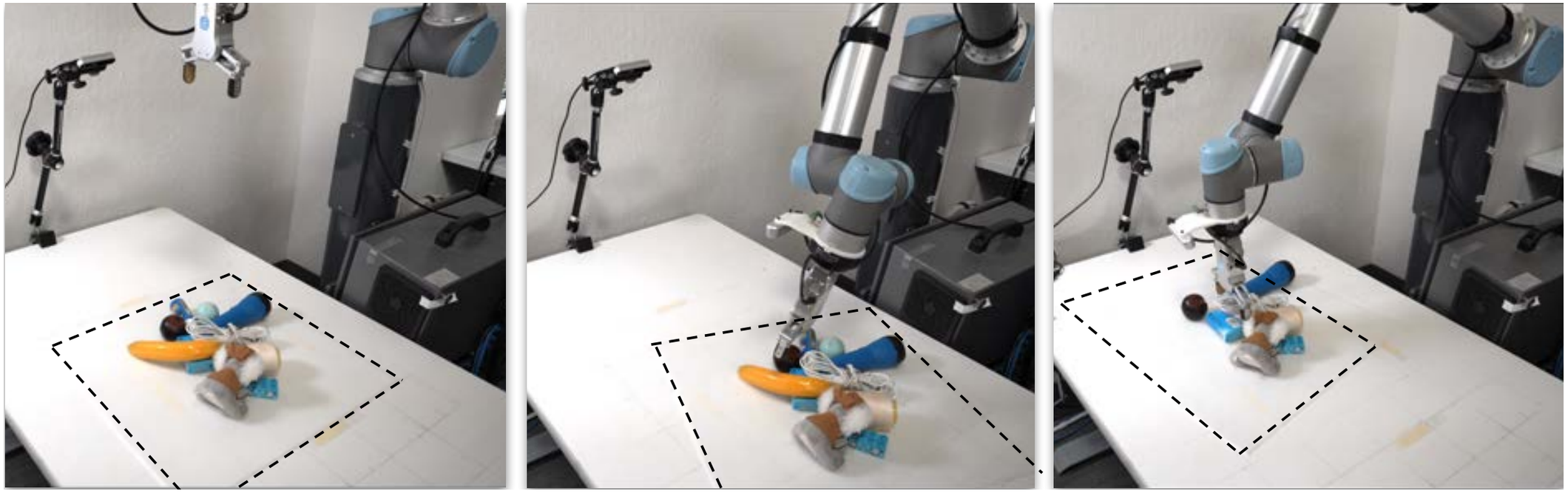
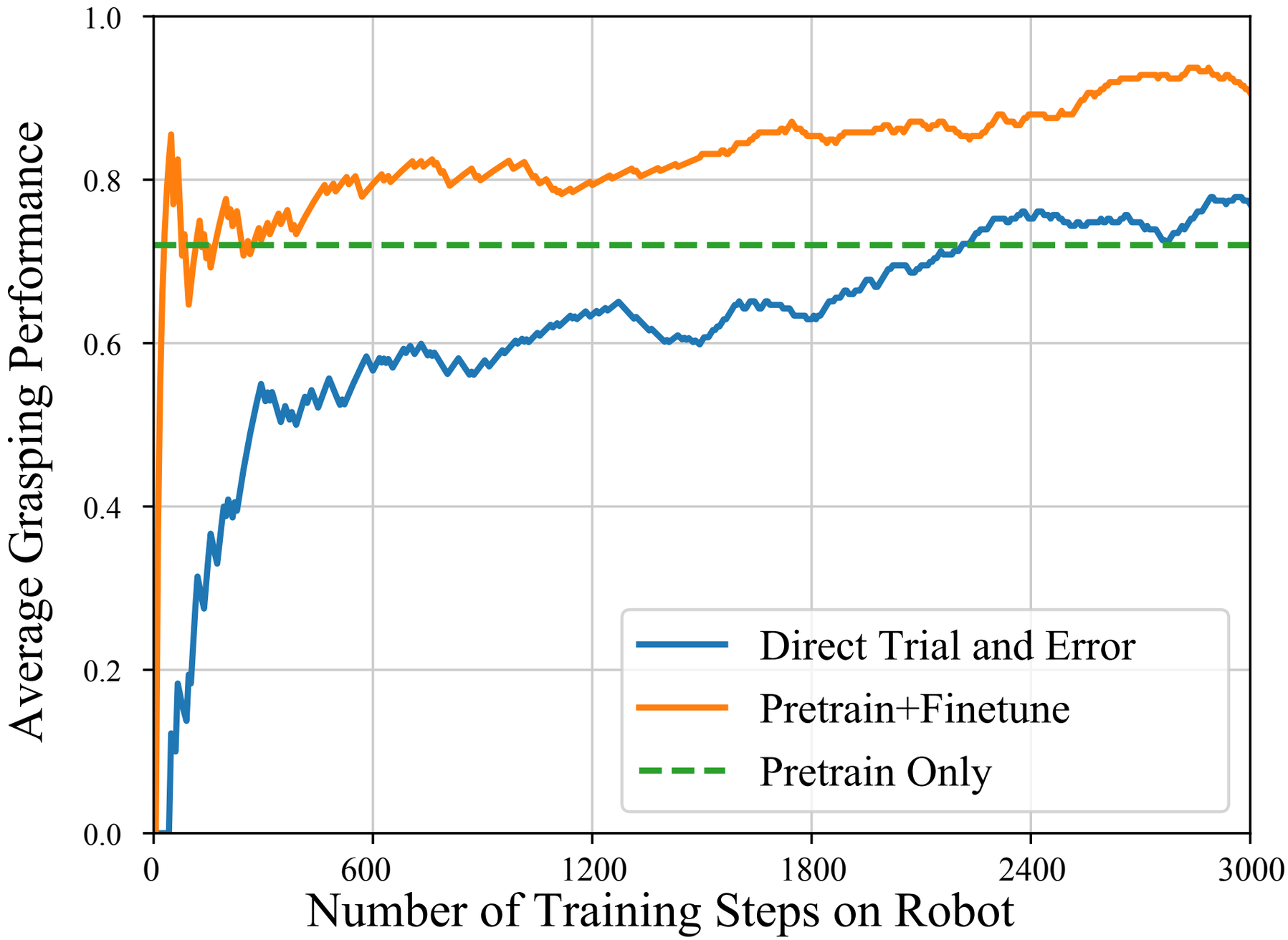


Fig. 6. In dynamic scene experiments, the entire pile of objects is randomly shifted around while the gripper approaches an object.

Experiments

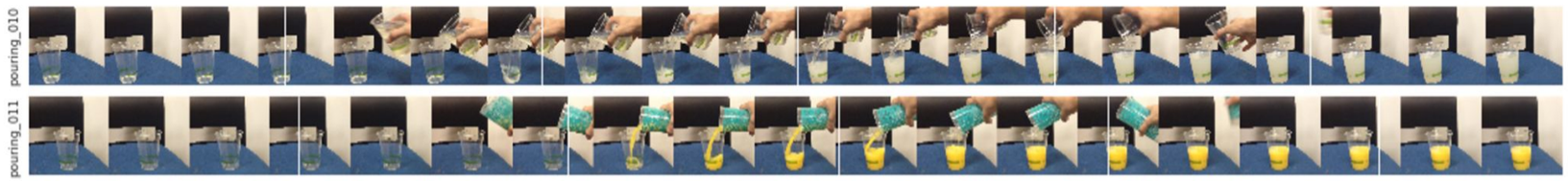


Unsupervised Perceptual Rewards for Imitation Learning

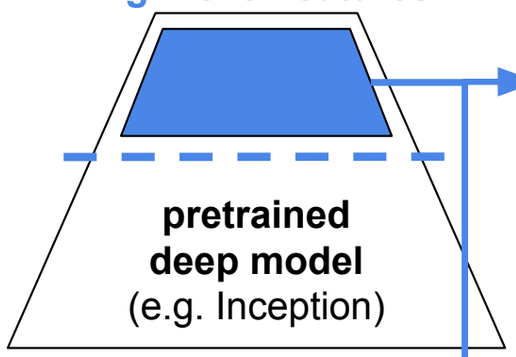
Pierre Sermanet* Kelvin Xu*† Sergey Levine
sermanet,kelvinxx,slevine@google.com
Google Brain

Demonstrator
(human or robot)

Few demonstrations

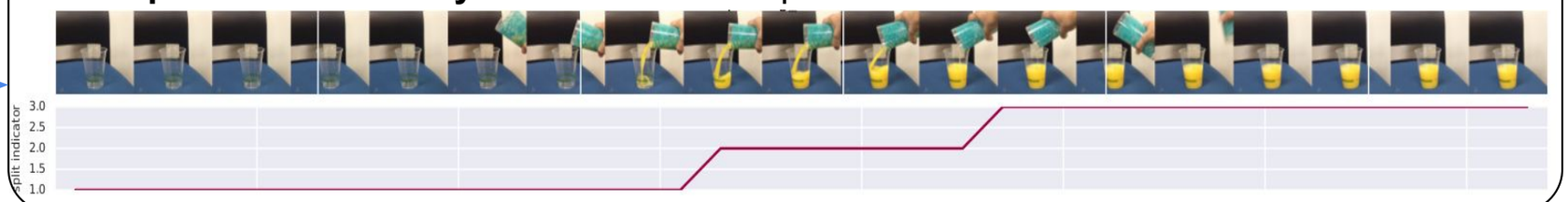


general
high-level features

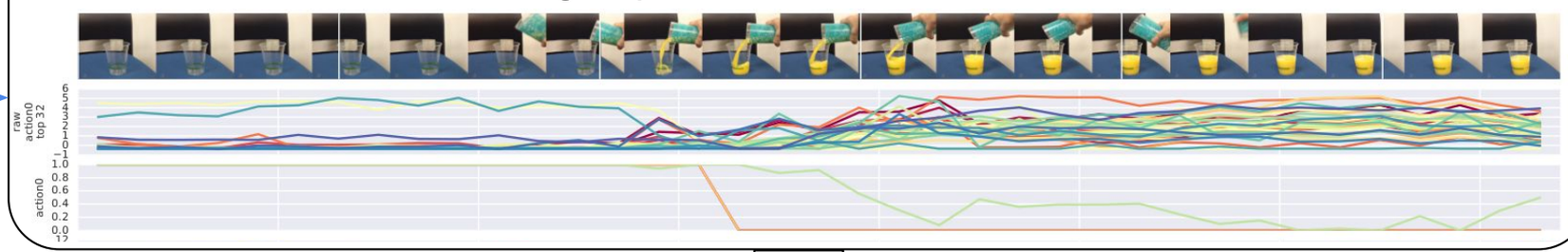


pretrained
deep model
(e.g. Inception)

Unsupervised discovery of intermediate steps



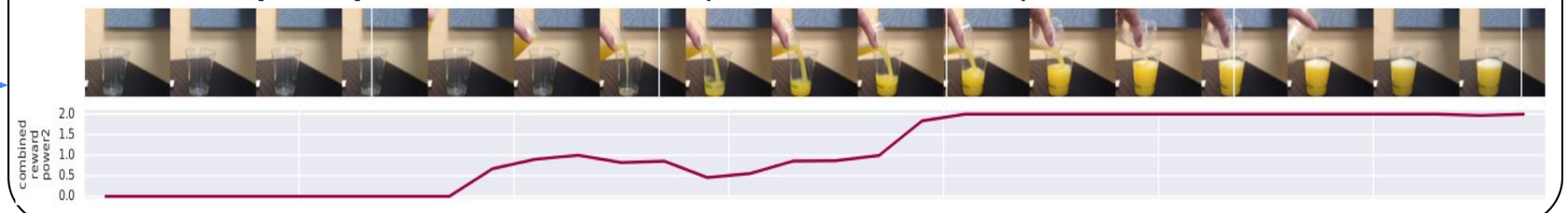
Feature selection maximizing step discrimination across all videos



OFFLINE COMPUTATION

REAL ROBOT

Real-time perceptual reward for multiple intermediate steps



Learning agent
with Reinforcement Learning

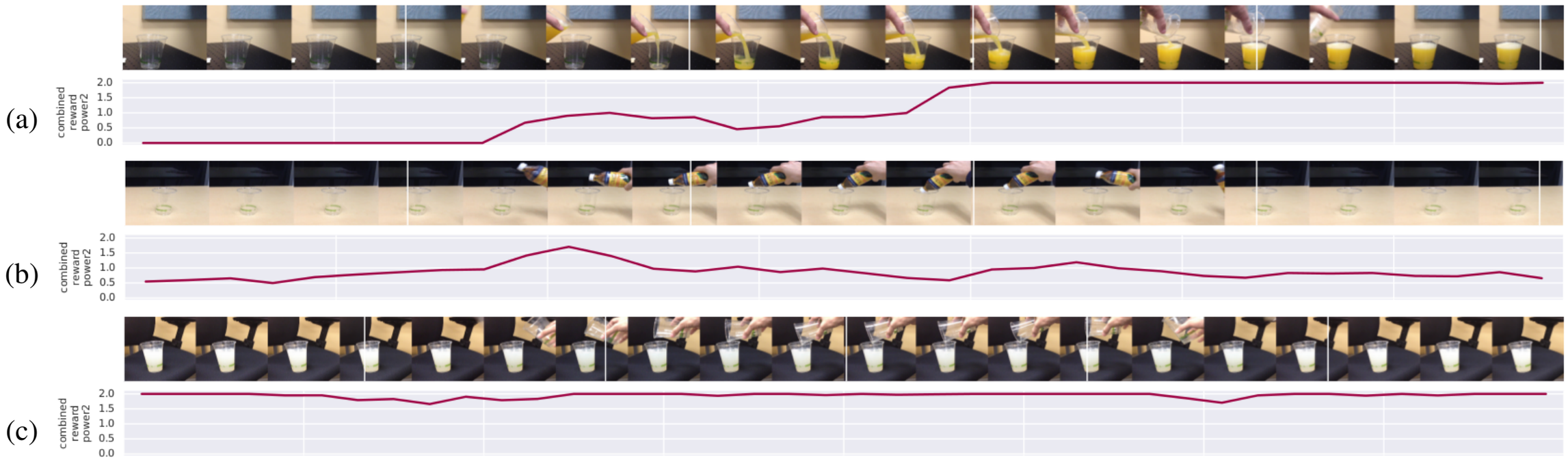
PI^2 approach from [Chebotar et al.]

Kinesthetic demonstration
(successful ~10% of the time)

Method

- Unsupervised discovery of steps:
  - find segments such that variance of features within is minimized
- Learn step classifiers (linear classifiers to classify discovered steps)
- Use step classifiers as reward functions: $\sum_{i=2}^{n} R_i(a) \times 2^{i-1}$
- Use reward functions to train policy using PI^2 RL algorithm

Results



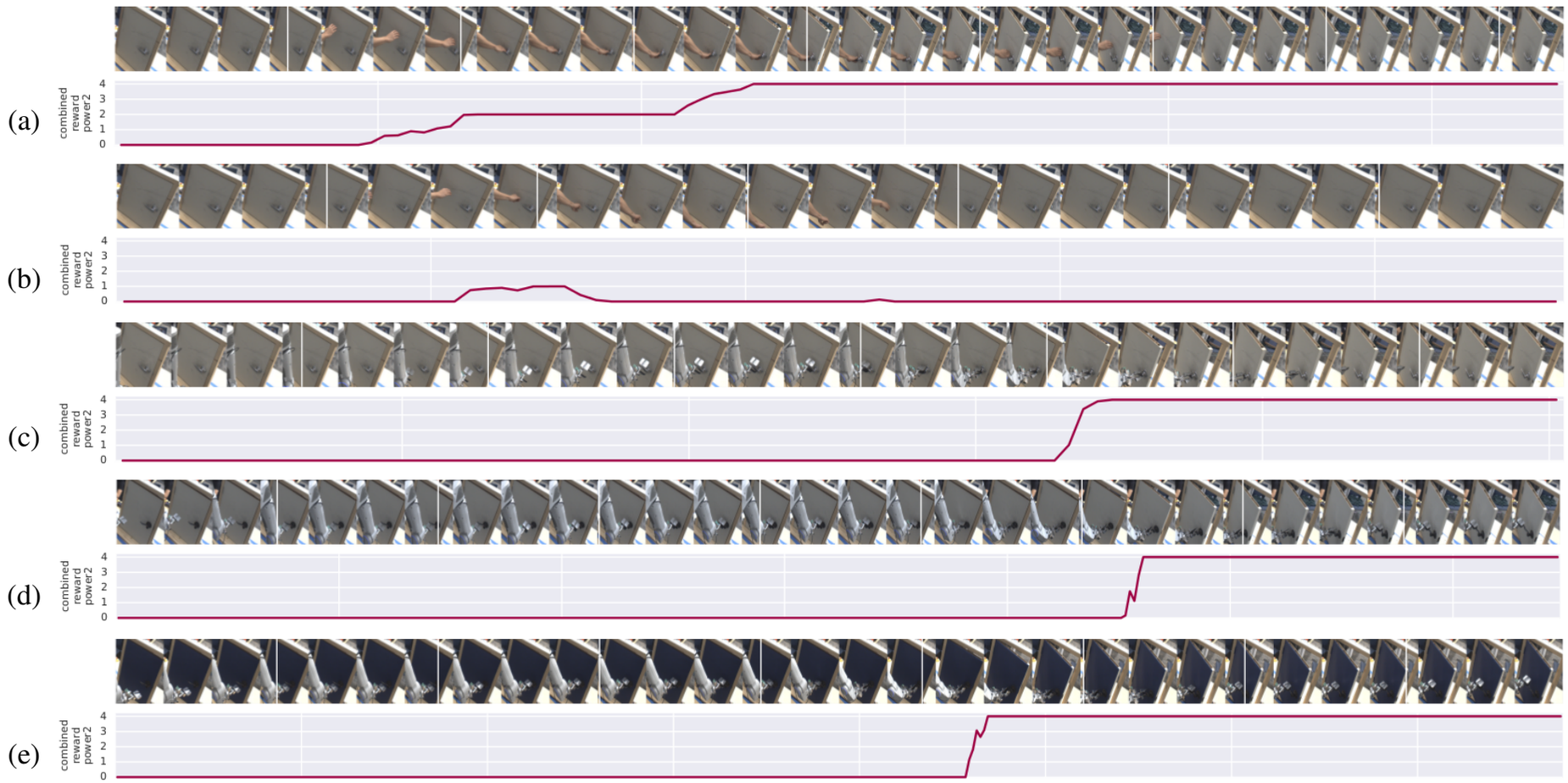
- State discovery accuracy:

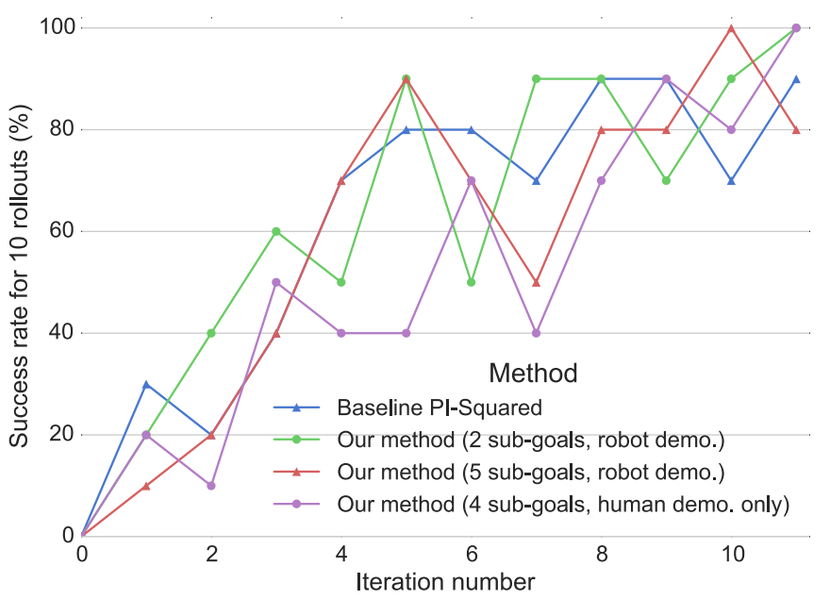
dataset (training)	method	2 steps average	3 steps average
door	ordered random steps	52.5%	55.4%
	unsupervised steps	**76.1%**	**66.9%**
pouring	ordered random steps	65.9%	52.9%
	unsupervised steps	**91.6%**	**58.8%**

- State classification accuracy:

dataset (testing)	classification method	2 steps average	3 steps average
door	random baseline	$33.6\% \pm 1.6$	$25.5\% \pm 1.6$
	feature selection	$72.4\% \pm 0.0$	$52.9\% \pm 0.0$
	linear classifier	$\mathbf{75.0\% \pm 5.5}$	$\mathbf{53.6\% \pm 4.7}$
pouring	random baseline	$31.1\% \pm 3.4$	$25.1\% \pm 0.1$
	feature selection	$65.4\% \pm 0.0$	$40.0\% \pm 0.0$
	linear classifier	$\mathbf{69.2\% \pm 2.0}$	$\mathbf{49.6\% \pm 8.0}$

Results


(a)
(b)
(c)
(d)
(e)
combined
reward
power2
0
1
2
3
4


Success rate for 10 rollouts (%)
Iteration number
Method
Baseline PI-Squared
Our method (2 sub-goals, robot demo.)
Our method (5 sub-goals, robot demo.)
Our method (4 sub-goals, human demo. only)
0
20
40
60
80
100
0
2
4
6
8
10

Thank you