A Neural Network Decomposition Framework for Multi-Robot Multi-Task Transfer Learning

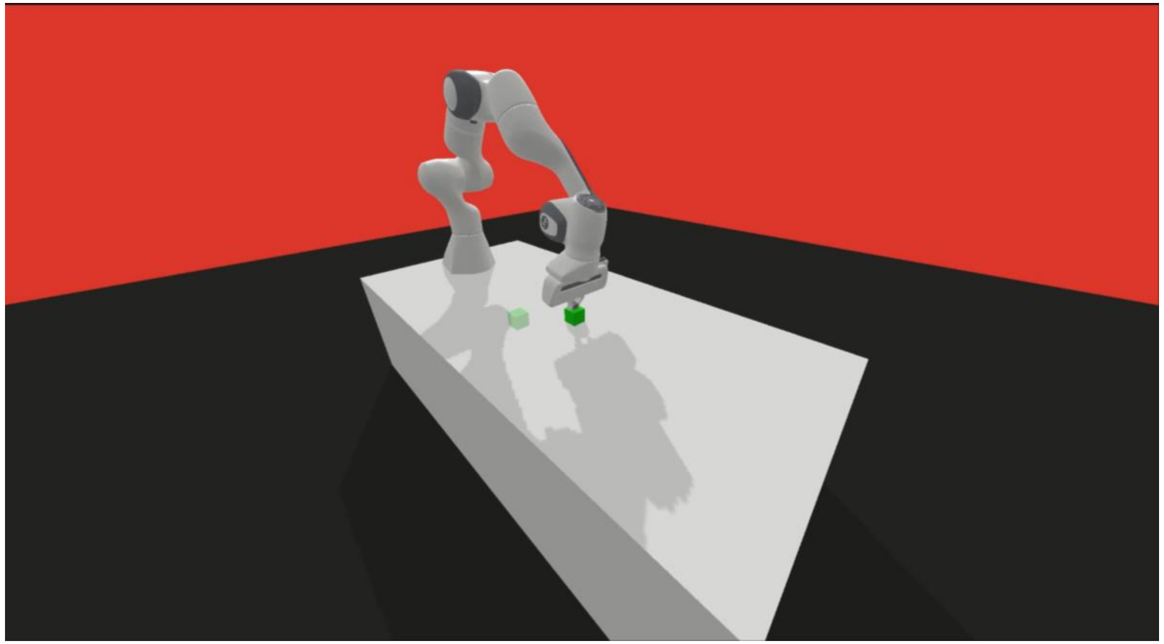
Pingcheng Jian
PhD candidate in ECE
Duke University

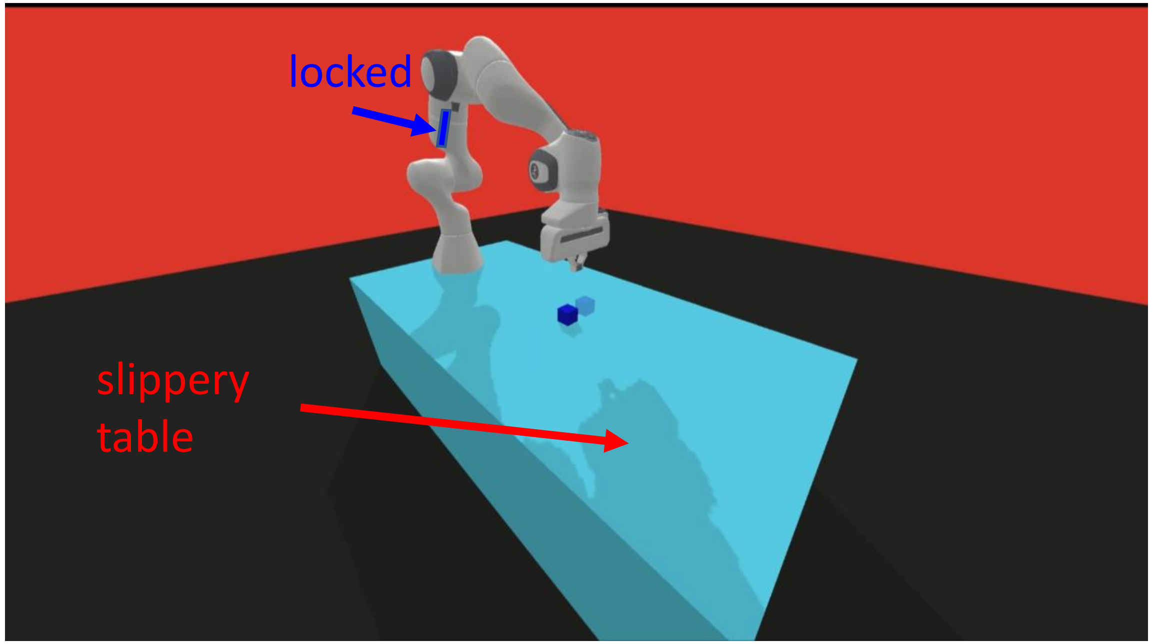
Zachary Bell
US Air Force
AFRL Division, USA

Boyuan Chen
Duke University

Michael M. Zavlanos
Duke University

Training robot with reinforcement learning

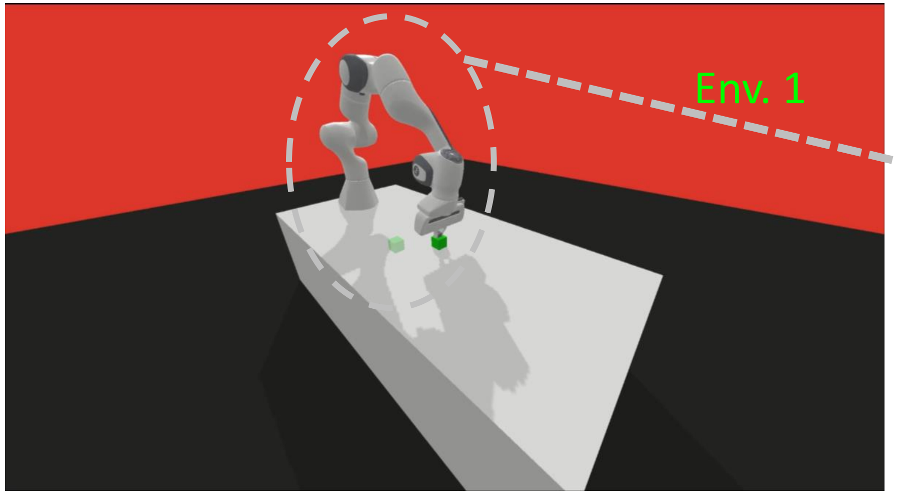




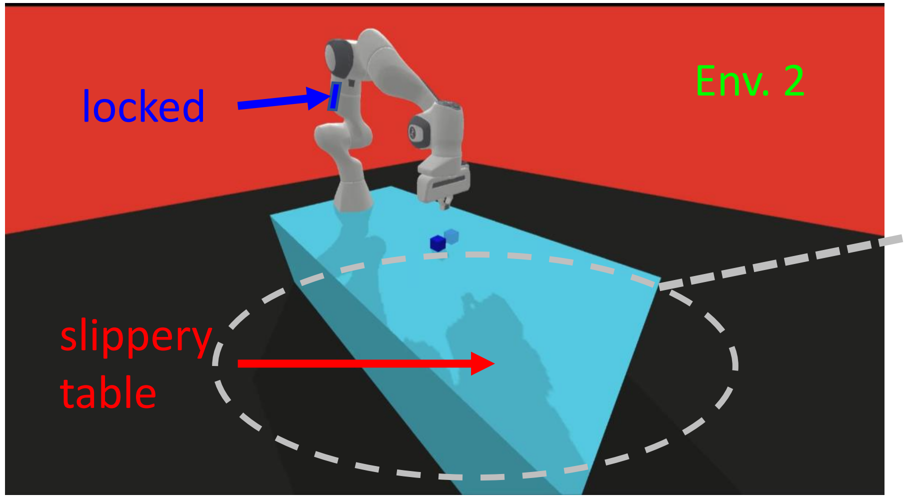
Pushing an object
Normal table: friction coefficient=1.0

The third joint of the robot is locked/damaged
The table is very slippery: friction coefficient=0.1

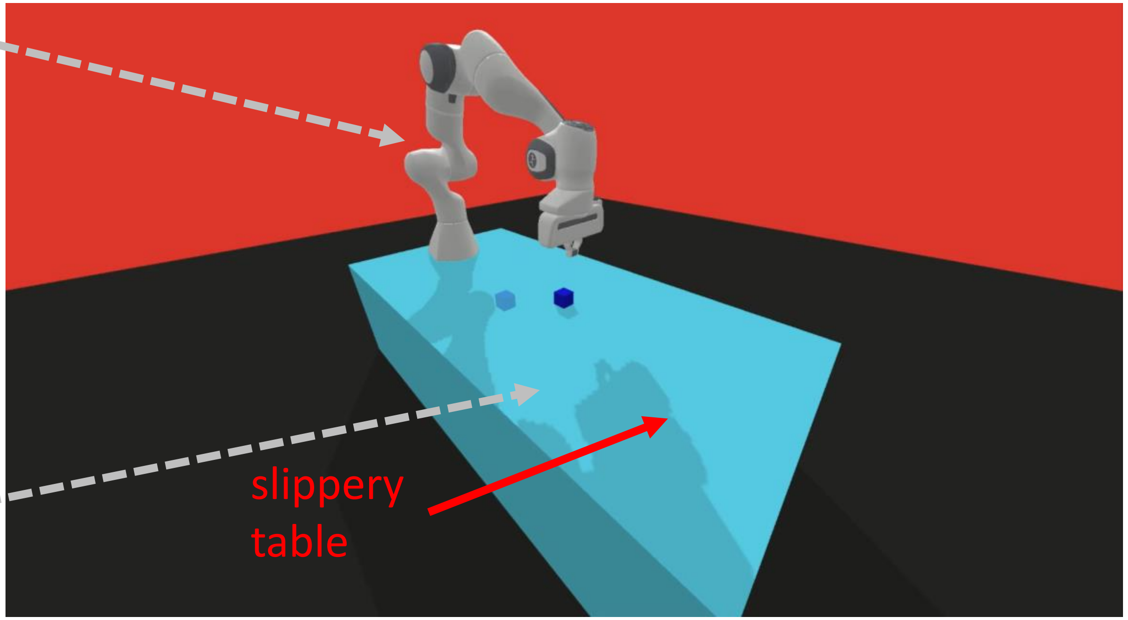
Transfer the trained policy between different environment



Pushing an object
Normal table: friction coefficient=1.0

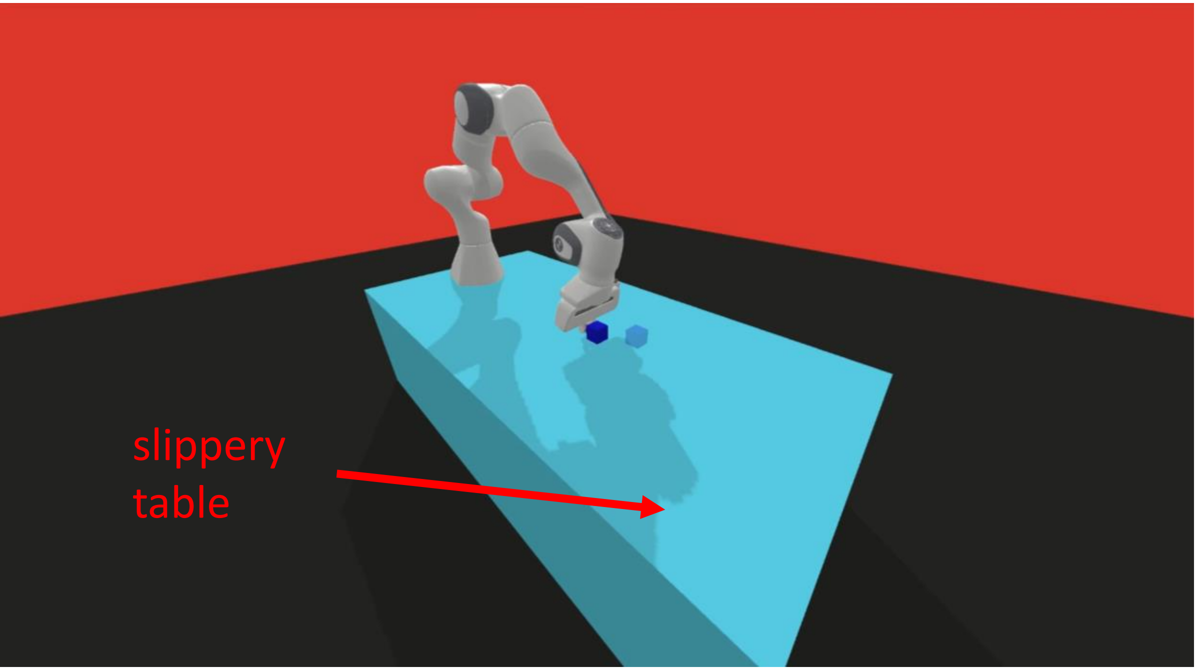


The third joint of the robot is locked/damaged
The table is very slippery: friction coefficient=0.1

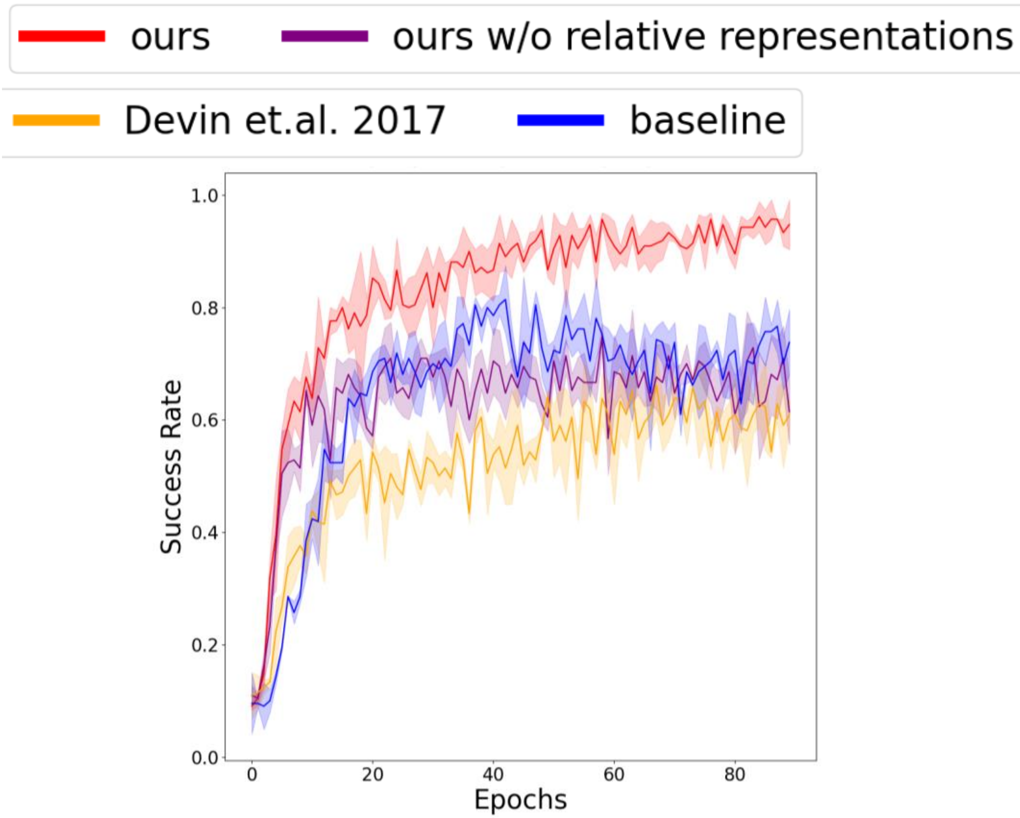


Transfer the trained policy from environment 1 to 2

With our proposed method



Few-shot transfer



Transfer learning success rates

our method with relative representation		our method w/o relative representation		Devin et. al. 2017		baseline method	
Success (%)	Touching(%)	Success(%)	Touching(%)	Success(%)	Touching(%)	Success(%)	Touching(%)
24.4 ± 1.5	99.9 ± 0.2	6.8 ± 0.7	93.5 ± 2.0	10.0 ± 0.9	78.3 ± 5.4	9.3 ± 2.1	17.9 ± 2.7

Zero-shot transfer

Research Motivation

- Efficient knowledge transfer method between different tasks

walk

run

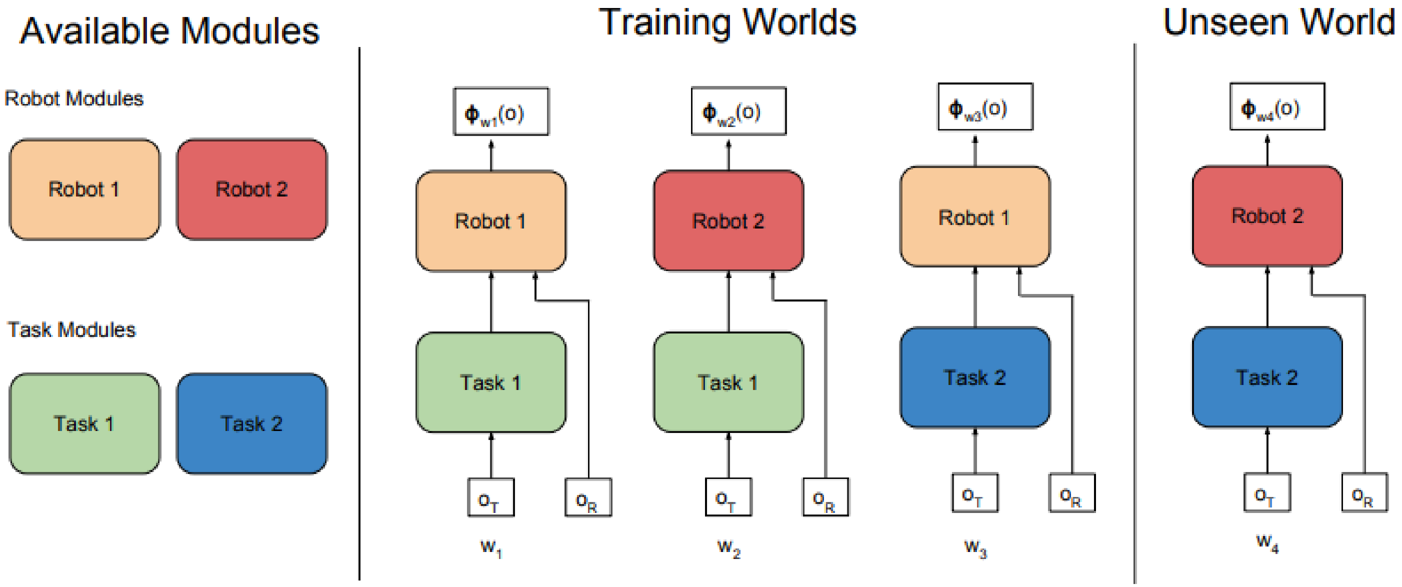
soccer

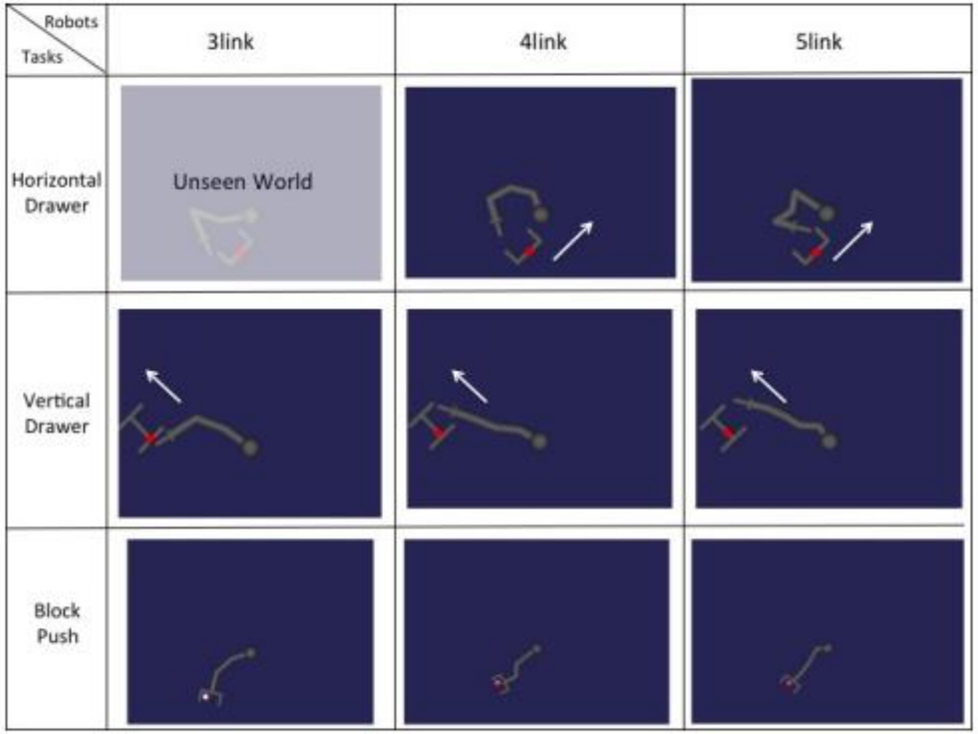
football

basketball

Different tasks share some common knowledge

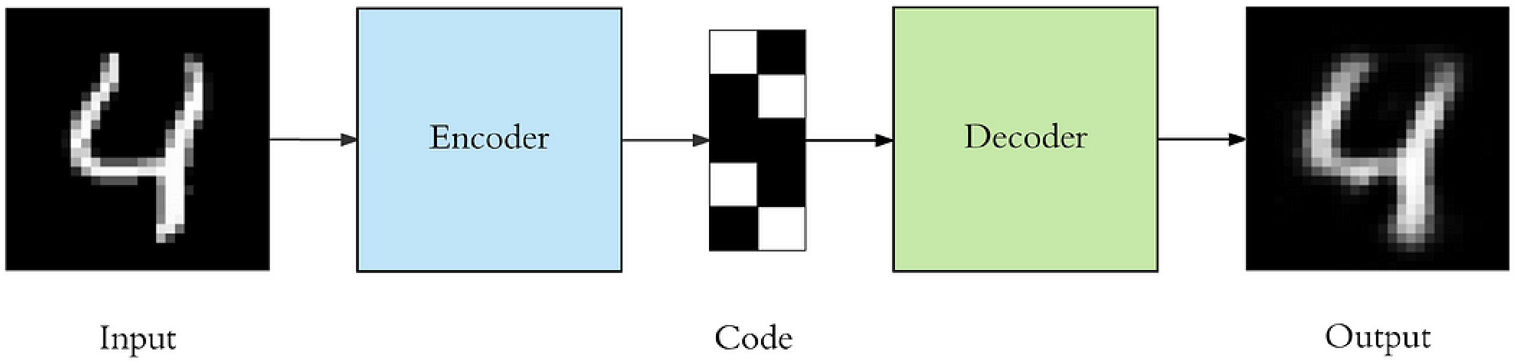
Related work

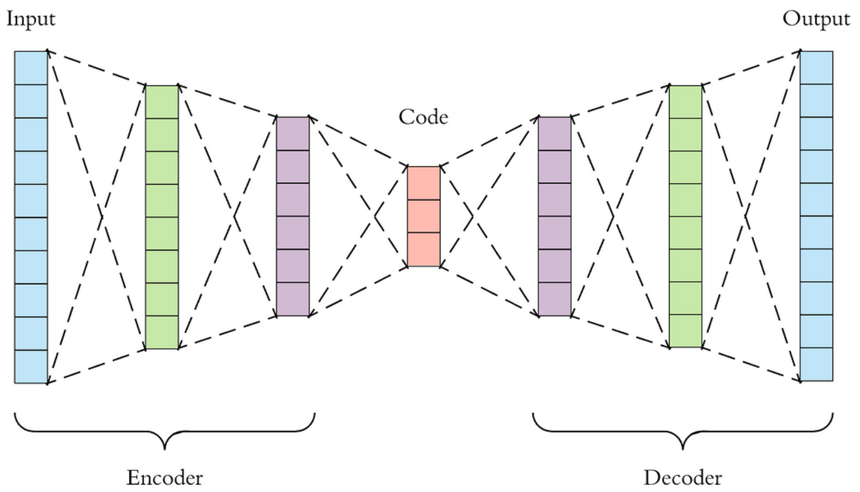


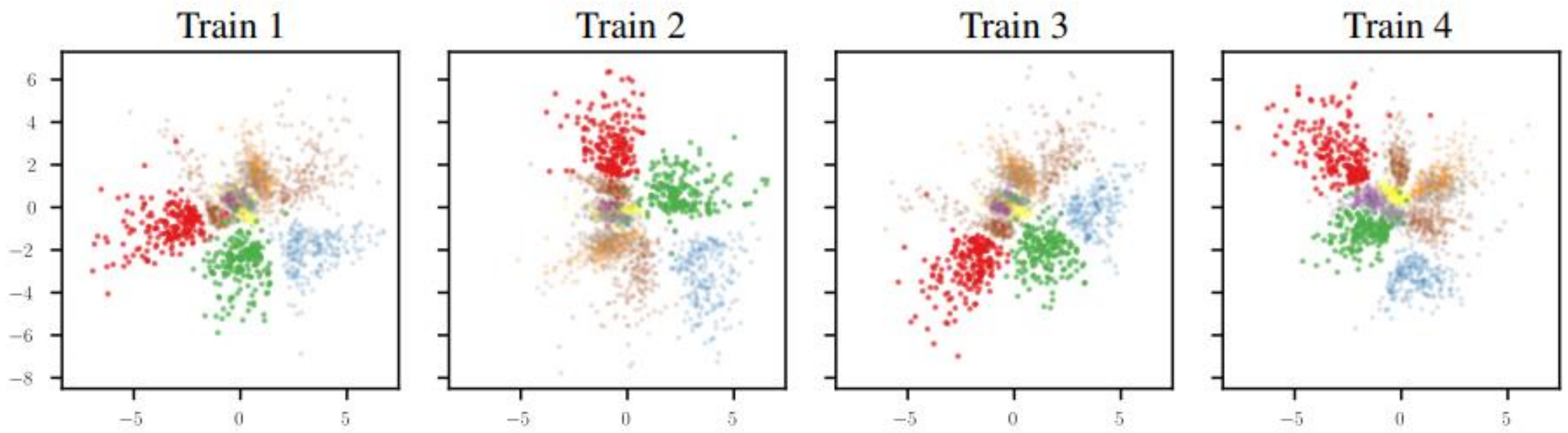


- *Learning Modular Neural Network Policies for Multi-Task and Multi-Robot Transfer (Devin et. al. 2017)*

Related work







- *Relative representations enable zero-shot latent space communication*
- isometric transformation relationship: rotations, reflections, rescaling, and translation

Proposed method

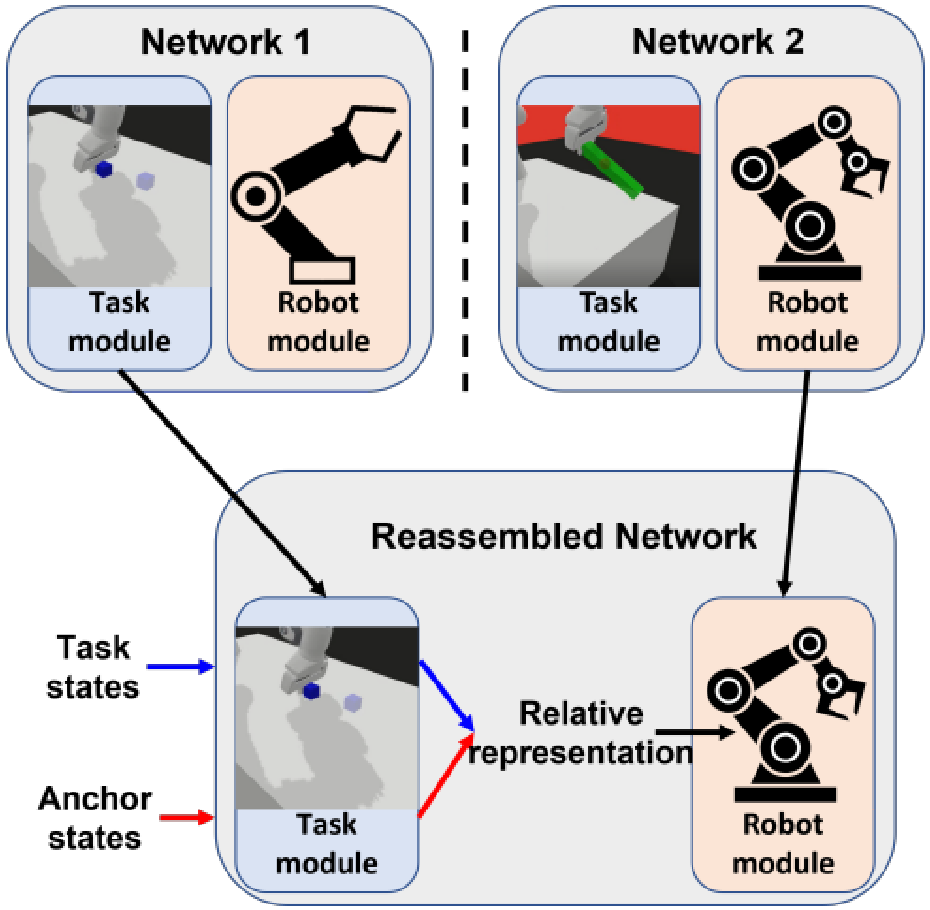
- Modular neural network with relative representation
- Select "anchor" states
- Calculate the cosine similarity

Cosine similarity: $S_C(\boldsymbol{a},\boldsymbol{b}) = \cos\theta = \frac{\boldsymbol{a}\boldsymbol{b}}{\|\boldsymbol{a}\|\|\boldsymbol{b}\|}$.

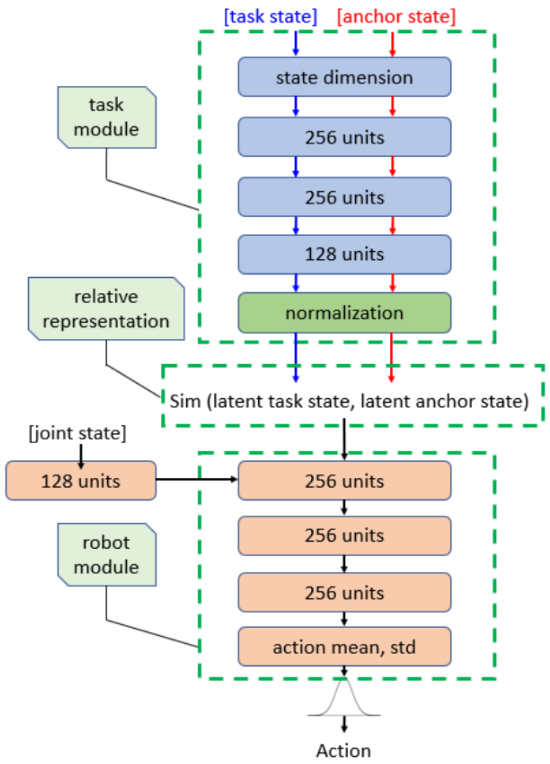
- Use relative representation to pass information

Relative representation:

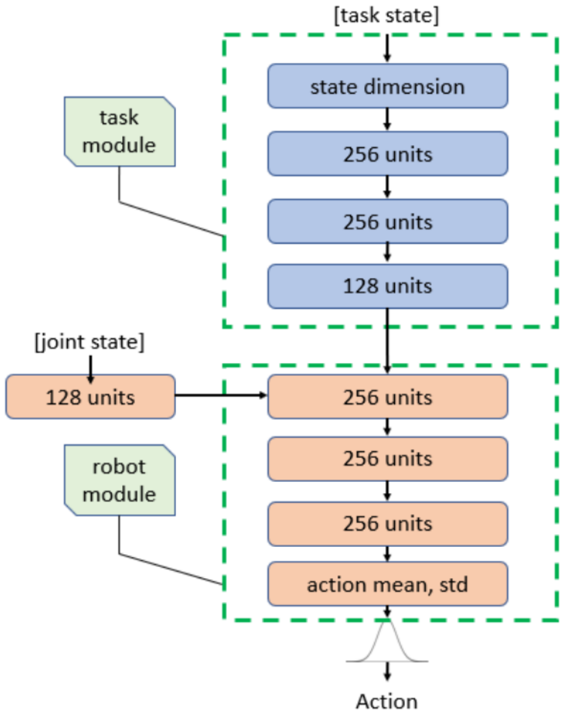
$$\boldsymbol{r}_{\boldsymbol{x}^{(i)}} = \left(\operatorname{sim}\left(\boldsymbol{e}_{\boldsymbol{s}^{(i)}}, \boldsymbol{e}_{\boldsymbol{a}^{(1)}}\right), \operatorname{sim}\left(\boldsymbol{e}_{\boldsymbol{s}^{(i)}}, \boldsymbol{e}_{\boldsymbol{a}^{(2)}}\right), \ldots, \operatorname{sim}\left(\boldsymbol{e}_{\boldsymbol{s}^{(i)}}, \boldsymbol{e}_{\boldsymbol{a}^{(|A|)}}\right)\right)$$



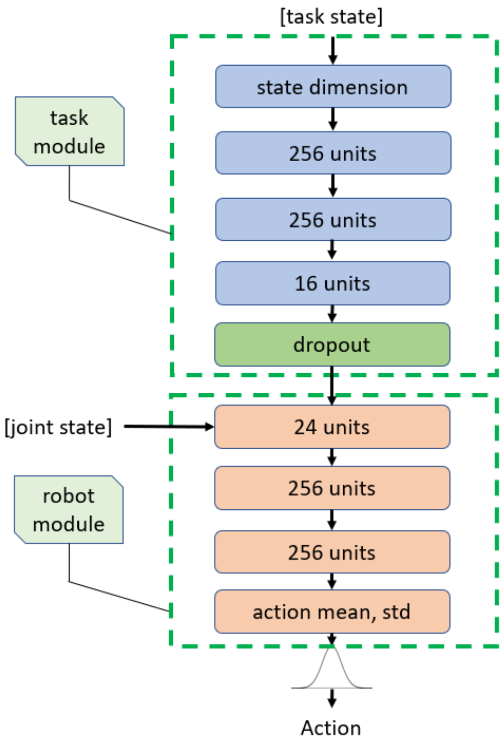
Proposed method



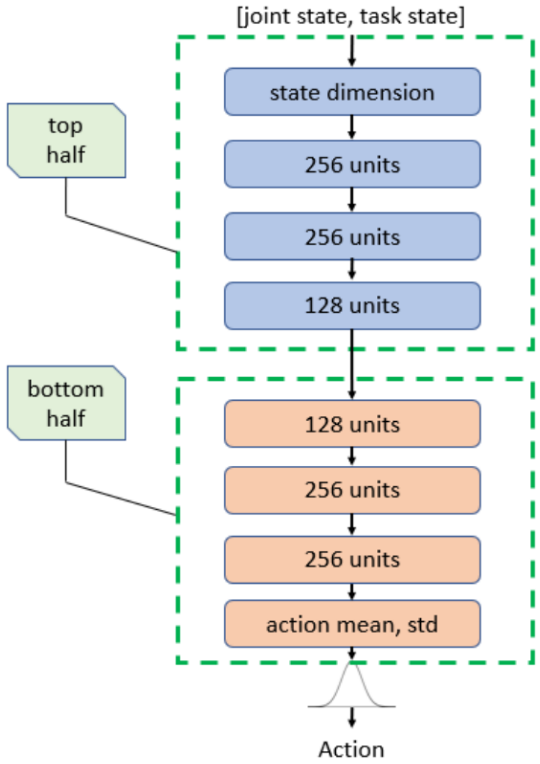
Our method: modular neural network with relative representation



Ablation method: modular neural network without relative representation



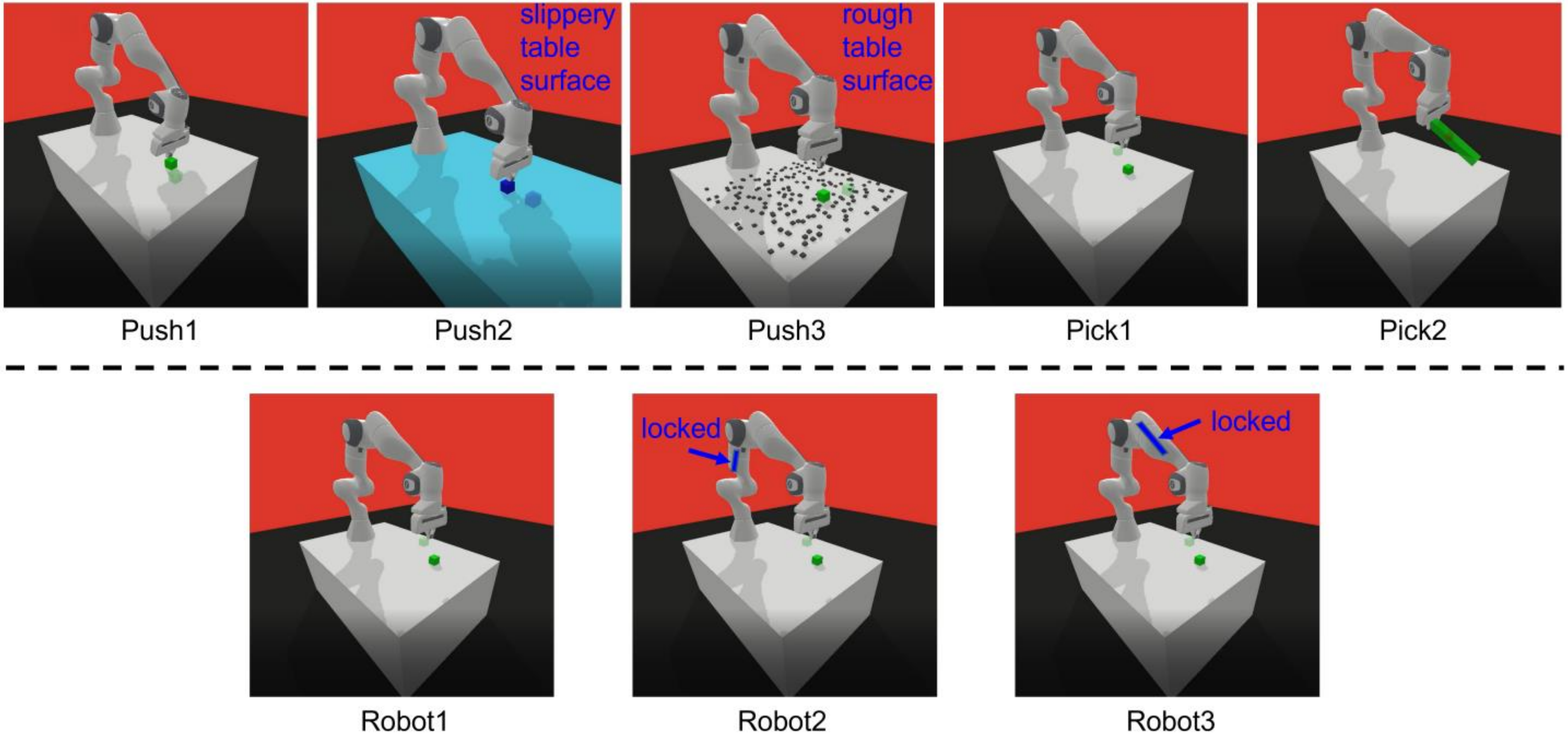
Devin et. al. 2017: modular neural network with small interface and dropout



Baseline method: fully connected neural network

- Use soft actor-critic for the training

Experiments setup



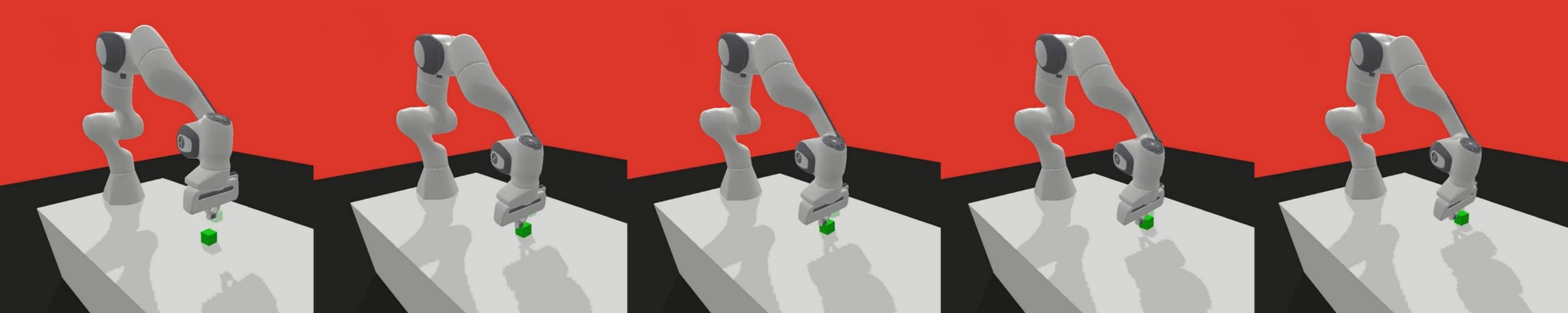
Experiments Results: zero-shot transfer

		our method with relative representation		our method w/o relative representation		Devin et. al. 2017		baseline method	
		Success (%)	Touching(%)	Success(%)	Touching(%)	Success(%)	Touching(%)	Success(%)	Touching(%)
E. 1	task: Pu1-R1 robot: Pu2-R2	26.9 ± 3.6	73.0 ± 1.4	13.9 ± 3.5	75.8 ± 4.4	11.7 ± 2.6	44.8 ± 3.1	6.5 ± 2.5	0.0 ± 0.0
E. 2	task: Pu2-R1 robot: Pu1-R2	24.4 ± 1.5	99.9 ± 0.2	6.8 ± 0.7	93.5 ± 2.0	10.0 ± 0.9	78.3 ± 5.4	9.3 ± 2.1	17.9 ± 2.7
E. 3	task: Pu1-R3 robot: Pu3-R1	16.5 ± 1.9	90.8 ± 2.3	7.2 ± 1.5	95.6 ± 0.8	9.8 ± 2.5	82.6 ± 1.6	8.4 ± 3.7	0.0 ± 0.0
E. 4	task: Pu3-R1 robot: Pu1-R2	13.2 ± 1.1	95.4 ± 1.9	14.9 ± 1.0	88.7 ± 1.1	11.8 ± 2.4	49.6 ± 1.1	9.0 ± 1.4	1.6 ± 1.0
E. 5	task: Pi1-R1 robot: Pu1-R2	2.1 ± 0.4	14.7 ± 2.7	4.0 ± 1.9	9.6 ± 1.6	2.9 ± 1.0	8.6 ± 0.3	3.8 ± 1.6	2.8 ± 0.9
E. 6	task: Pu1-R3 robot: Pi1-R2	$8.8 + 1.9$	55.6 ± 2.4	11.6 ± 2.0	80.8 ± 3.4	10.5 ± 2.1	30.5 ± 4.2	9.0 ± 0.7	0.1 ± 0.2
E. 7	task: Pi2-R1 robot: Pi1-R3	4.7 ± 1.6	41.8 ± 1.7	3.6 ± 0.8	13.1 ± 2.9	3.0 ± 1.6	9.8 ± 1.3	3.2 ± 1.4	0.0 ± 0.0
E. 8	task: Pi1-R3 robot: Pi2-R1	5.3 ± 1.7	18.4 ± 1.3	3.0 ± 0.7	18.8 ± 2.6	3.1 ± 0.5	13.6 ± 1.1	3.0 ± 0.7	0.0 ± 0.0

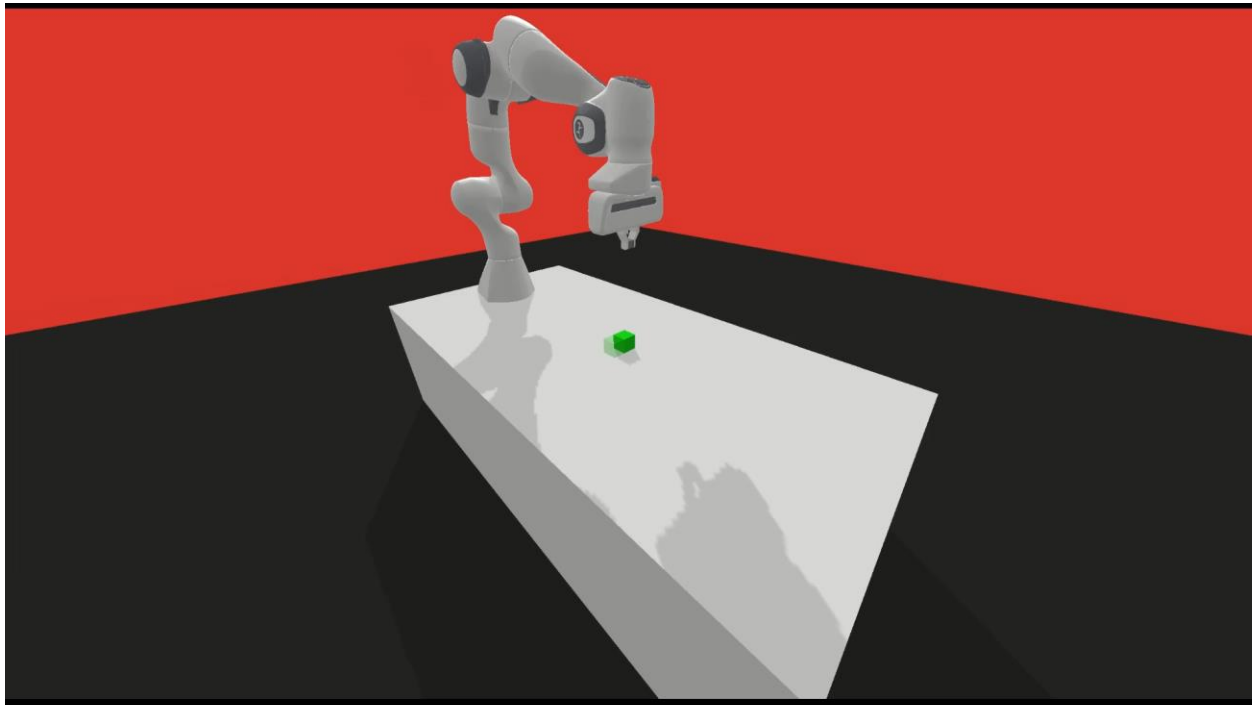
- “task: Pu1-R3” means the task module is trained in Push1-Robot3 world
- “robot: Pi1-R2” means the robot module is trained in Pick1-Robot2 world
- Reassemble these two modules and directly apply this stitched policy network in the Push1-Robot2 environment without any finetuning.

Visualize the robot arm operation process of the three method

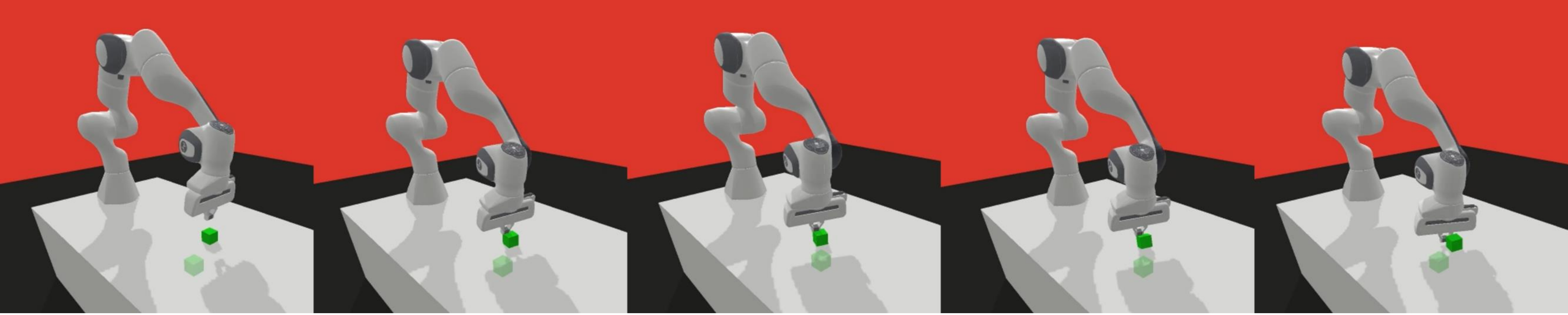
- The zero-shot behavior of the stitched policy in Experiment 1



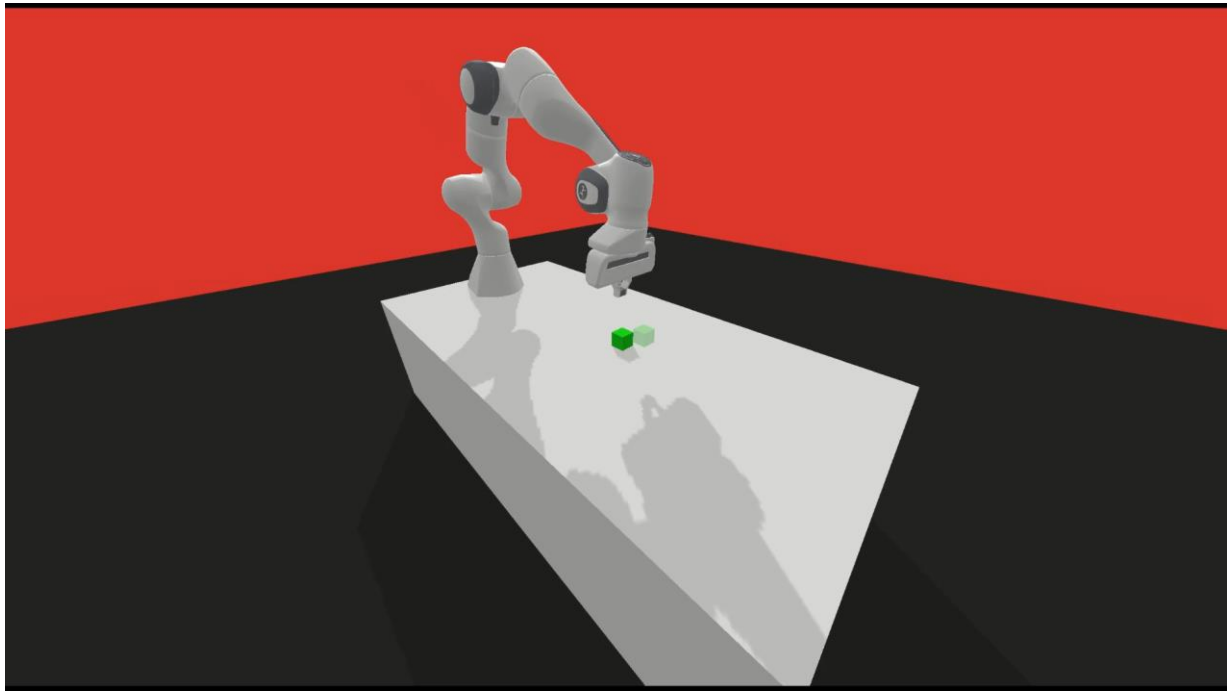
Ours method



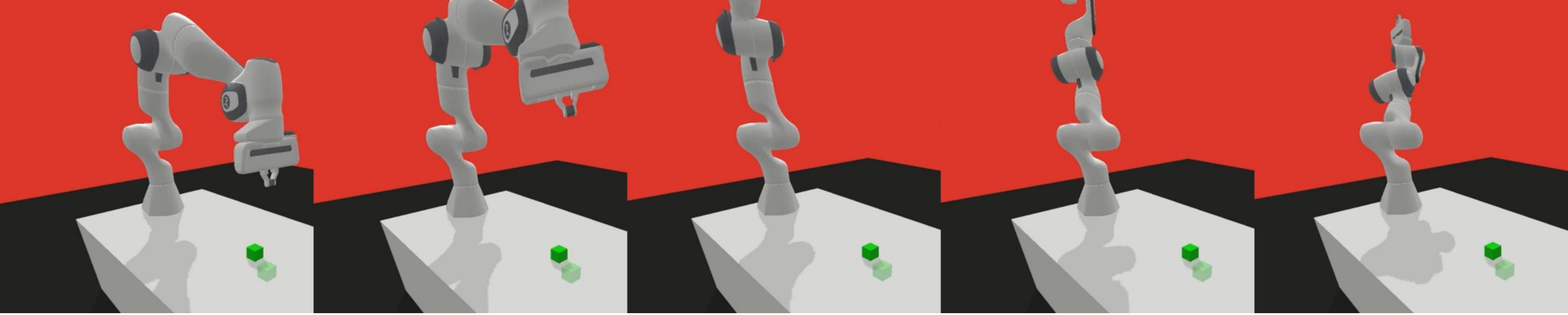
- Successfully push the object to the goal in some cases.



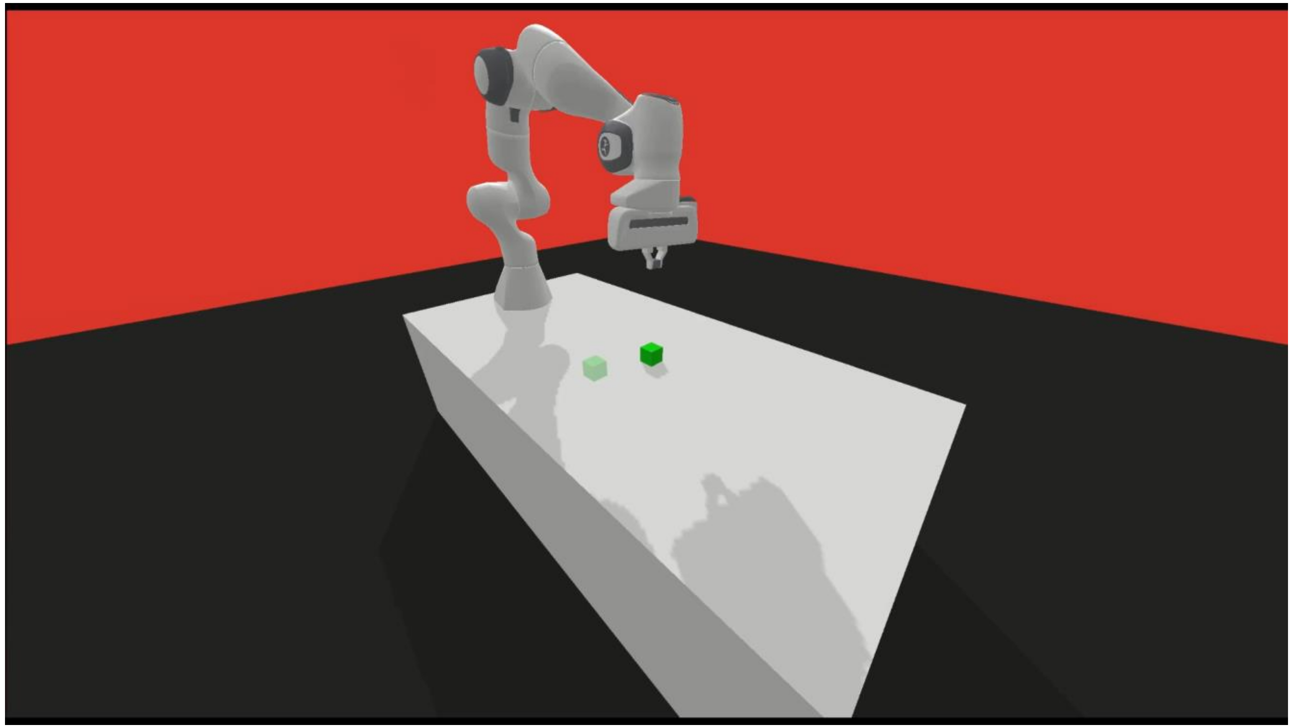
Ablation method



- Touch the object but cannot push it to the goal.

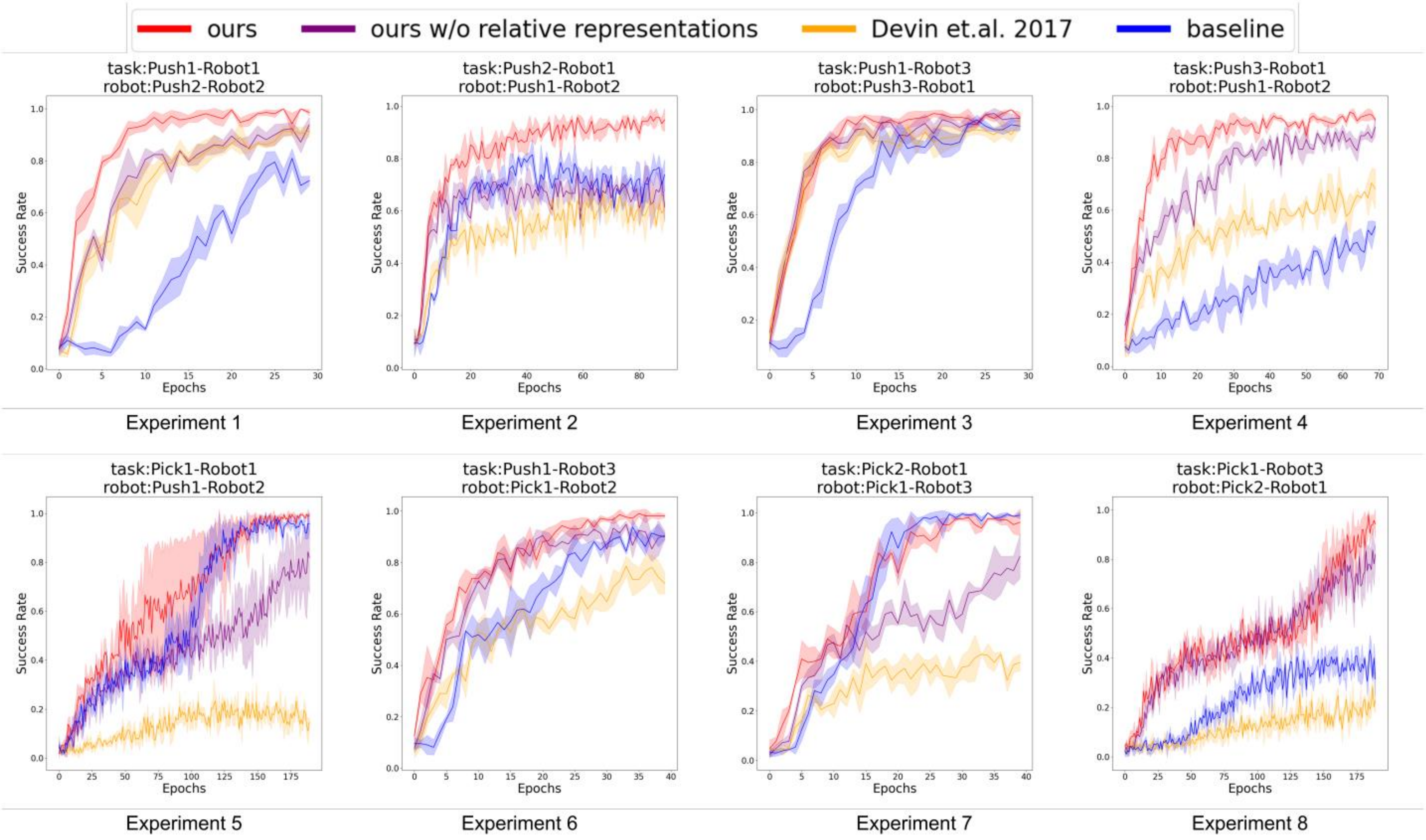


Baseline method



- Swing its arm in the air randomly.

Experiments Results: few-shot transfer learning



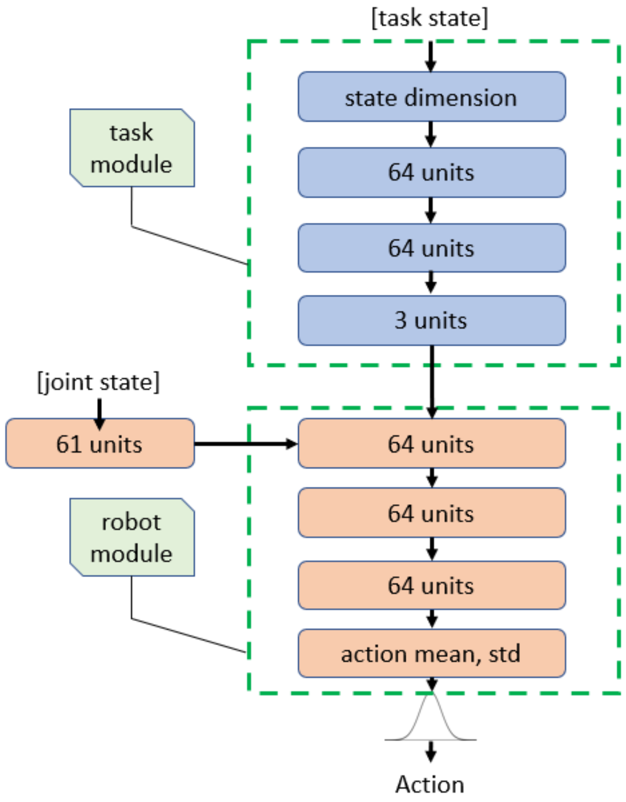
Analysis of the modules interface

- Our method is based on an assumption: The interfaces of different trainings have an isometric transformation relationship.

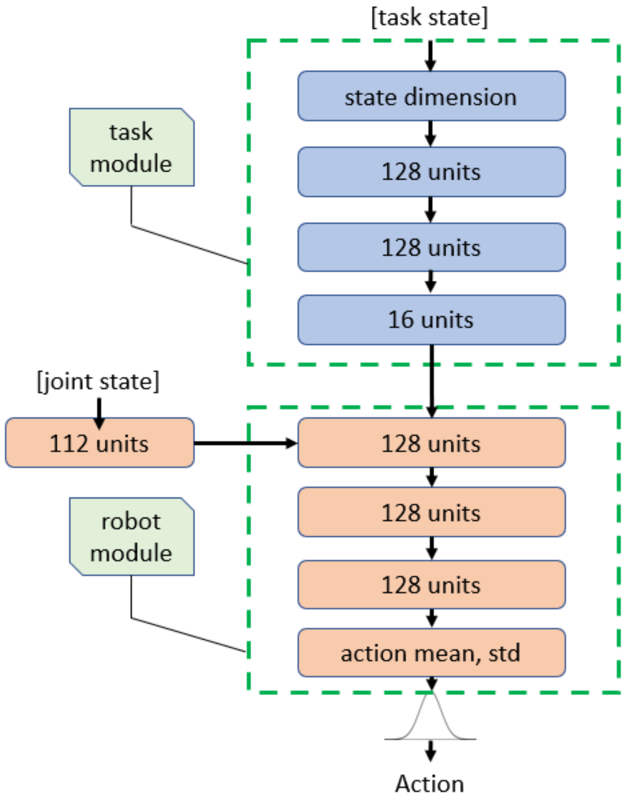
- This assumption is empirically proved in the supervised learning in previous work.

- Does it also stand in the reinforcement learning?

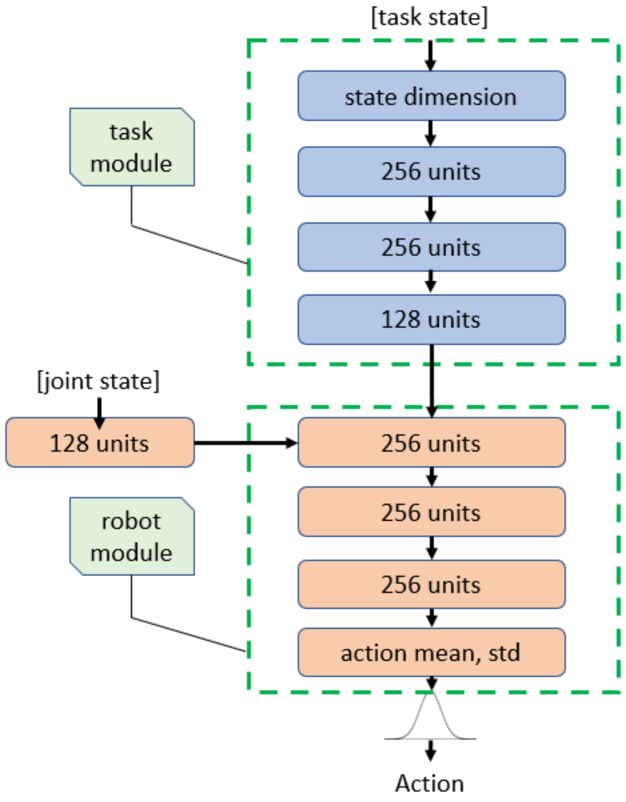
Networks for analyzing the interfaces



Small network



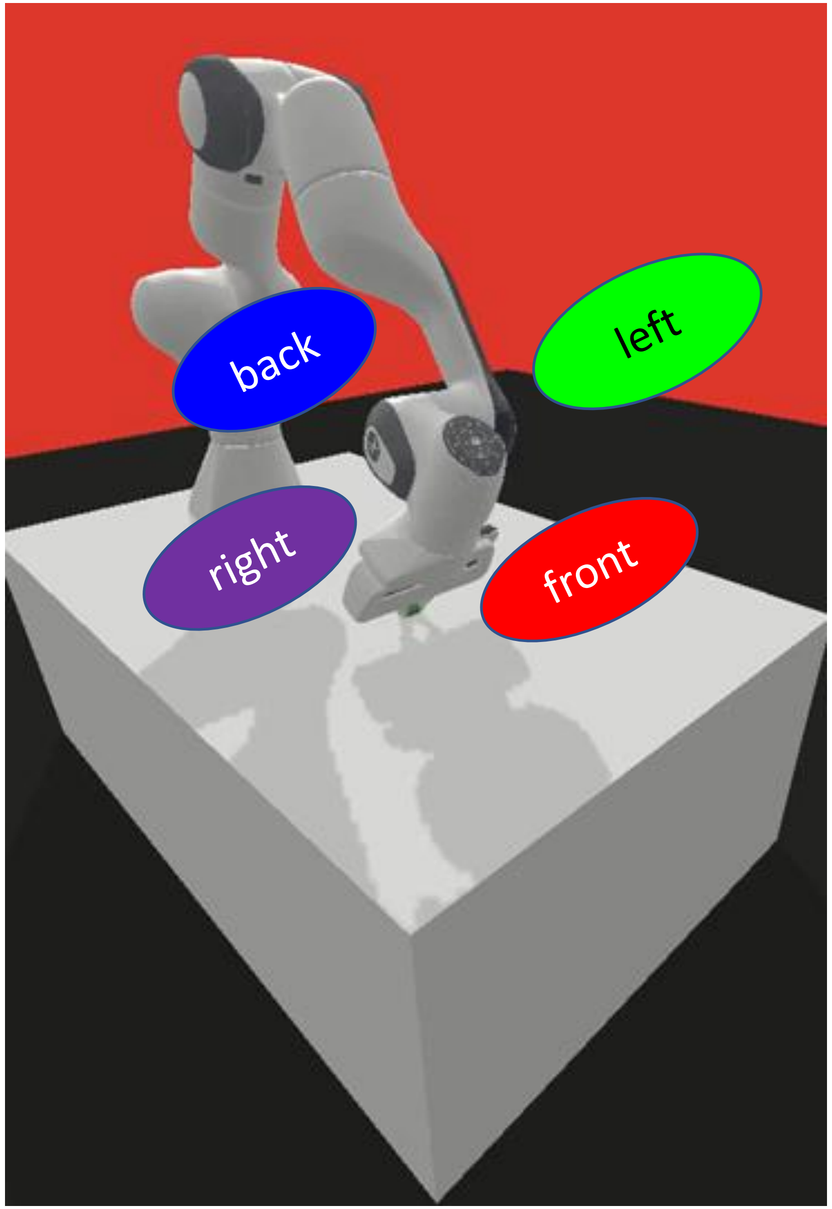
Medium network



Large network

Task state sampled from Reaching task in four directions

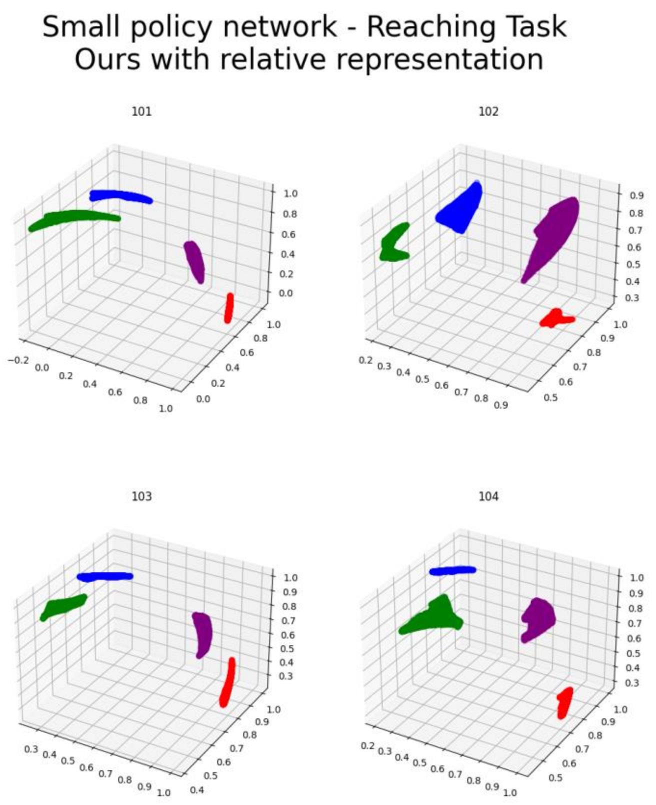
- Need labels for the RL data.
- Reaching task.
- Calculate the average pairwise distance of the same class of data at different network interfaces.
- Cosine distance and L2 distance.



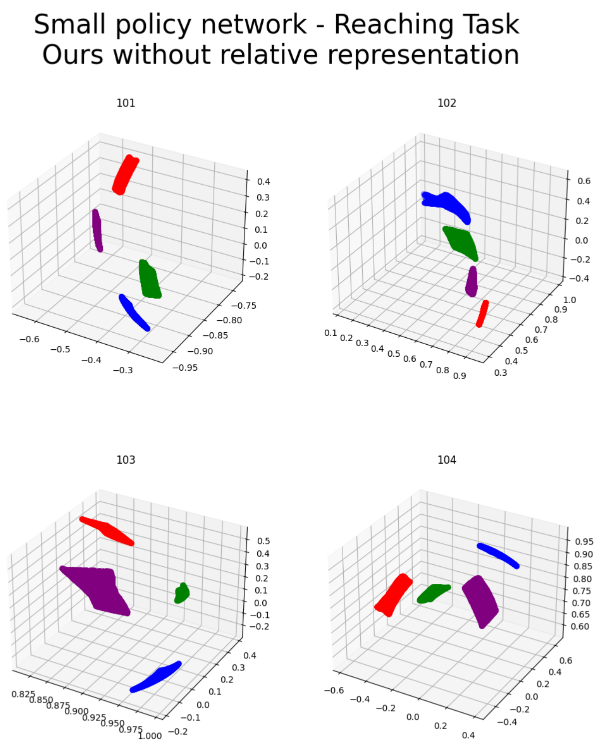
Visualizing interfaces from different training random seed

Using four different random seed: 101 - 104

Small networks: visualize interface directly



Ours with relative representation

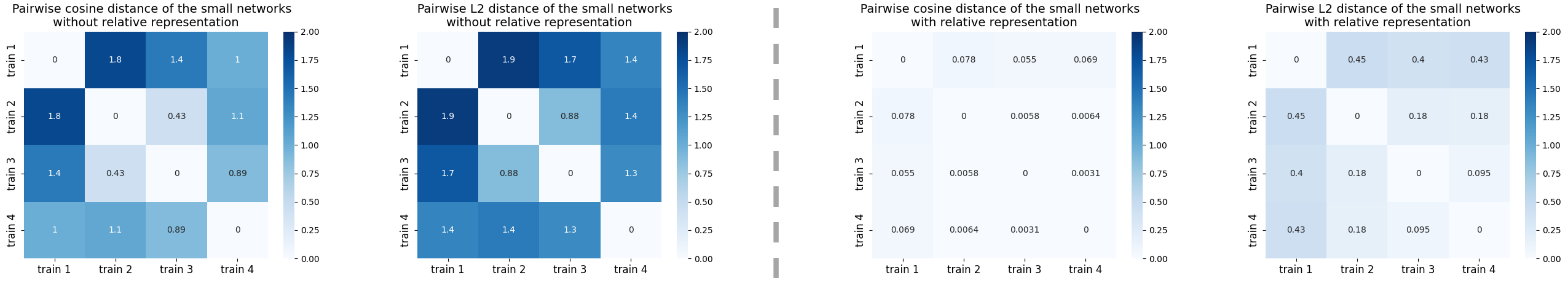


Ablation method without relative representation

Ours interfaces are basically identical. Without relative representation, there is an isometric transformation relationship.

Small networks: pairwise distance

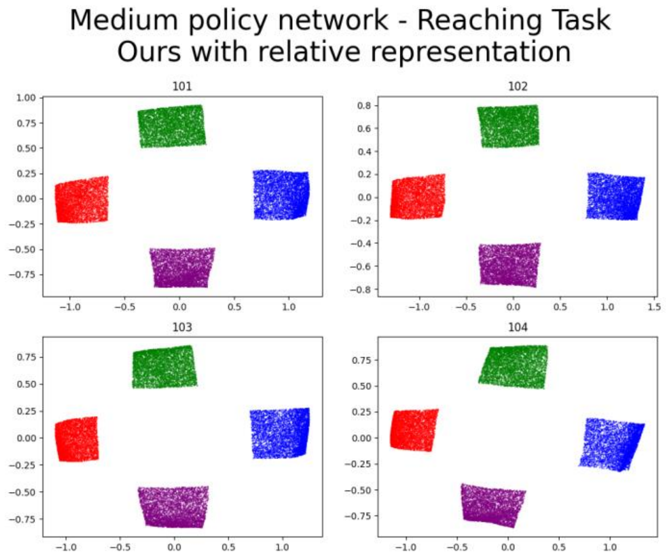
	Cosine distance	L2 distance
Without relative representation	1.106 ± 0.434	1.426 ± 0.322
With relative representation	0.0363 ± 0.0319	0.289 ± 0.141



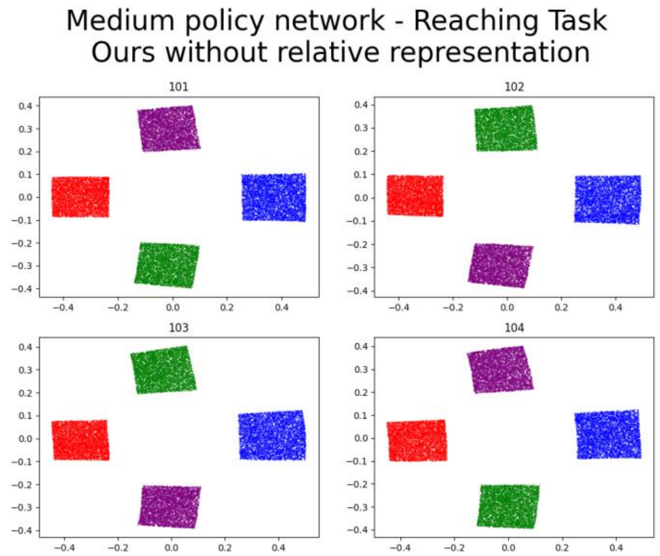
Reaching task:

When using the relative representation, the pairwise distance (cosine, L2) of the same task state at the latent space is significantly smaller.

Medium networks: visualize interface with PCA



Ours with relative representation



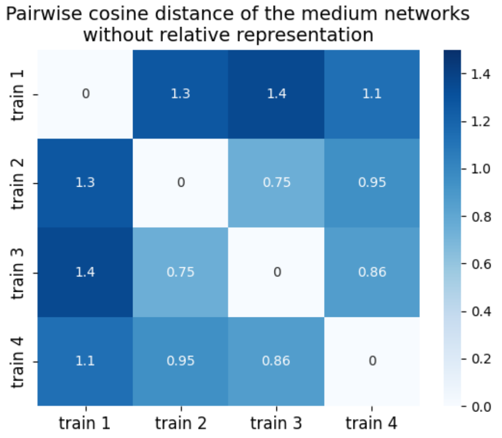
Ablation method without relative representation

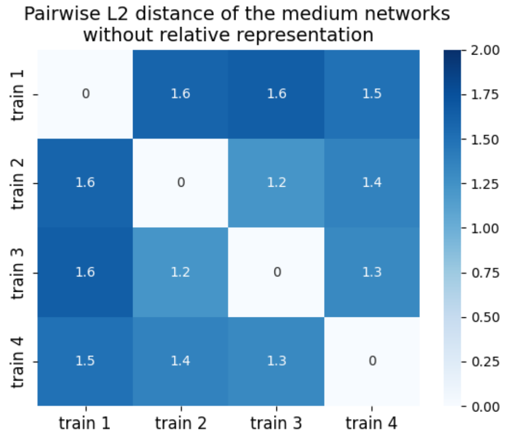
front back left right

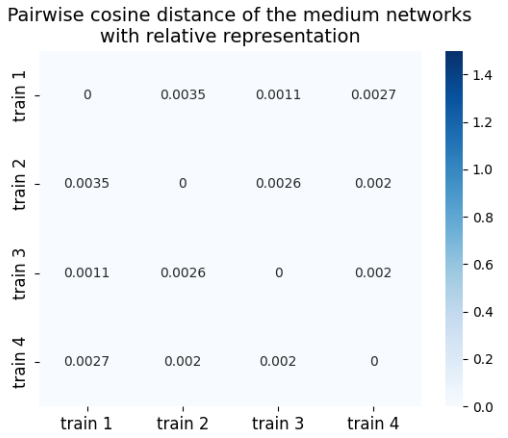
Ours interfaces are basically identical. Without relative representation, there is an isometric transformation relationship.

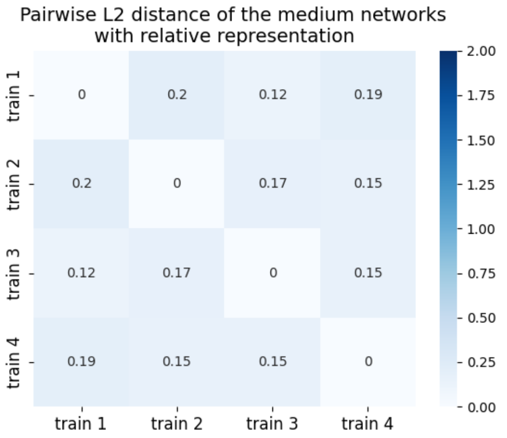
Medium networks: pairwise distance

	Cosine distance	L2 distance
Without relative representation	1.054±0.218	1.442±0.151
With relative representation	0.00231±0.00073	0.165±0.0





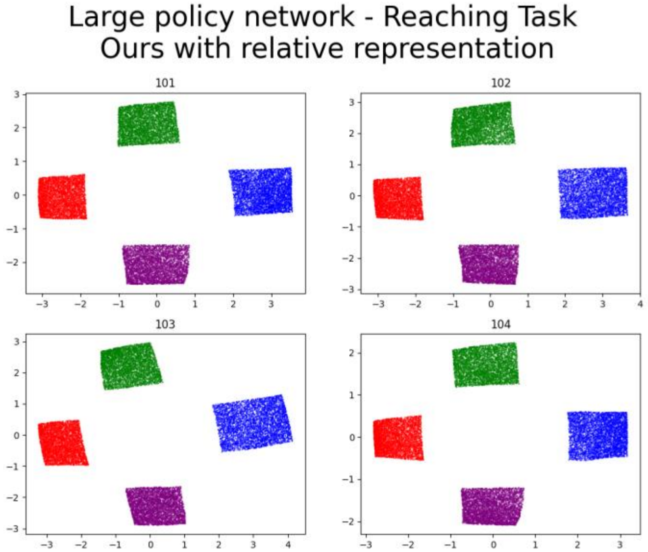




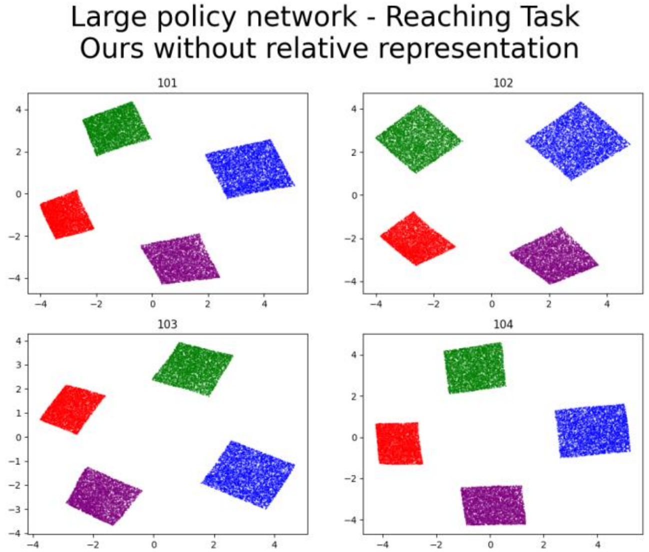
Reaching task:

When using the relative representation, the pairwise distance (cosine, L2) of the same task state at the latent space is significantly smaller.

Large networks: visualize interface with PCA



Ours with relative representation

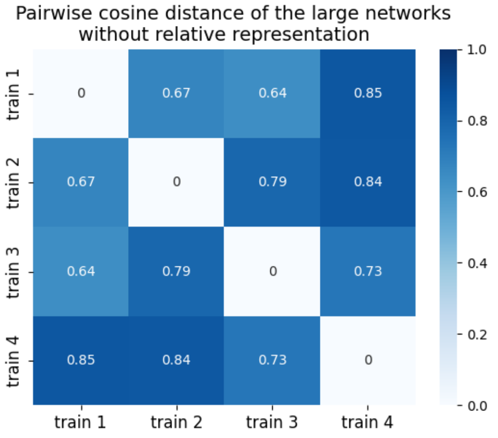


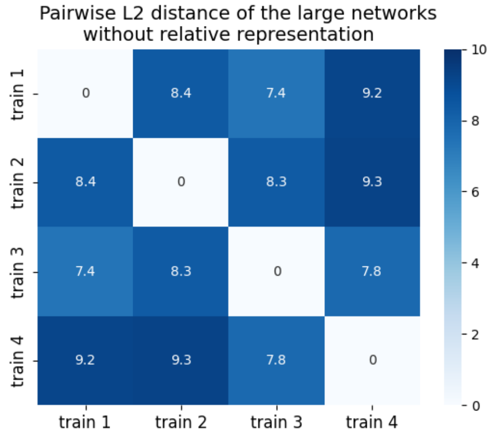
Ablation method without relative representation

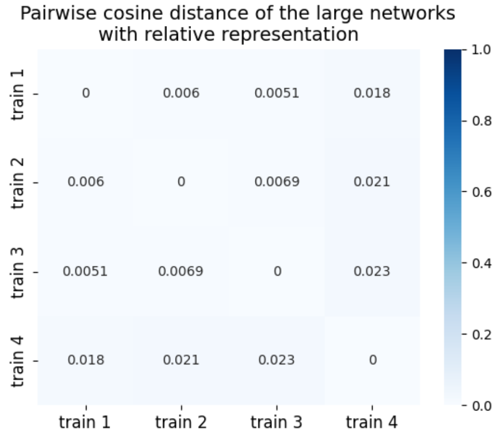
This network structure is exactly what we used for all the few shot transfer learning experiments.

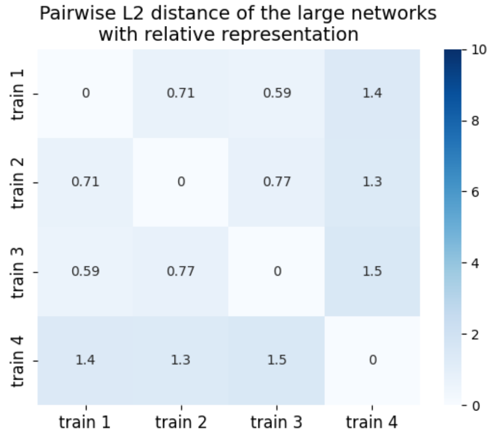
Large networks: pairwise distance on reaching task

	Cosine distance	L2 distance
Without relative representation	0.753 ± 0.081	8.386 ± 0.687
With relative representation	0.013 ± 0.007	1.051 ± 0.368









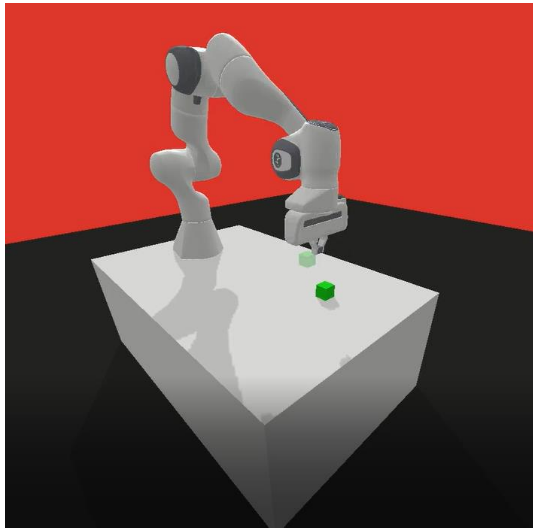
Reaching task:

When using the relative representation, the pairwise distance (cosine, L2) of the same task state at the latent space is significantly smaller.

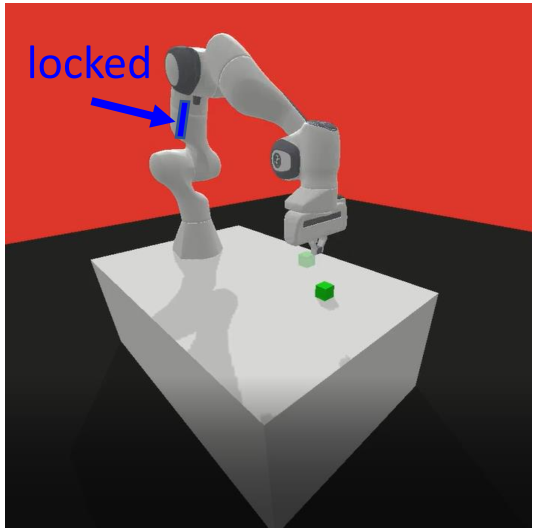
Visualizing interfaces from different task-robot pairs

Using three different pairs:

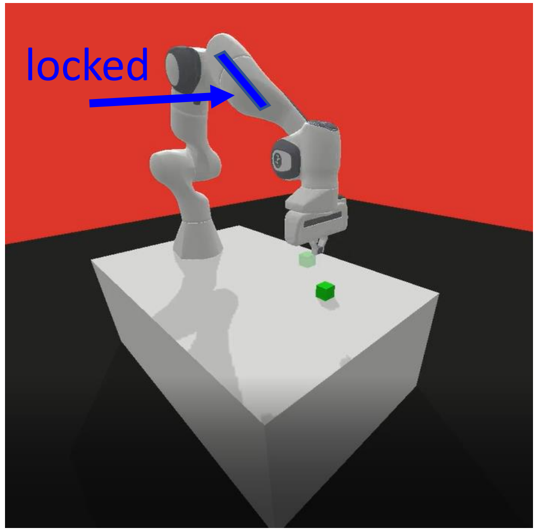
- Reach-robot1
- Reach-robot2
- Reach-robot3



Robot1



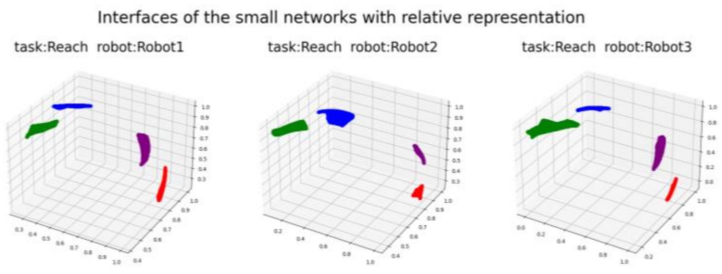
Robot2



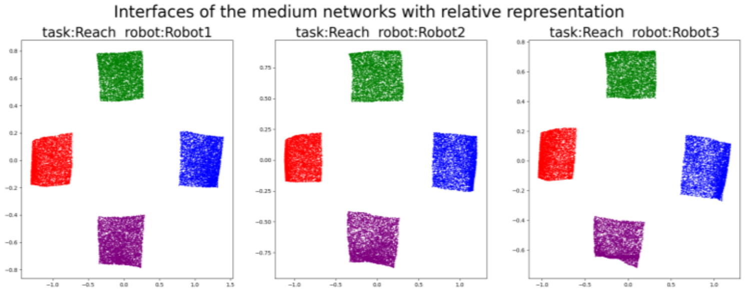
Robot3

Visualize interfaces

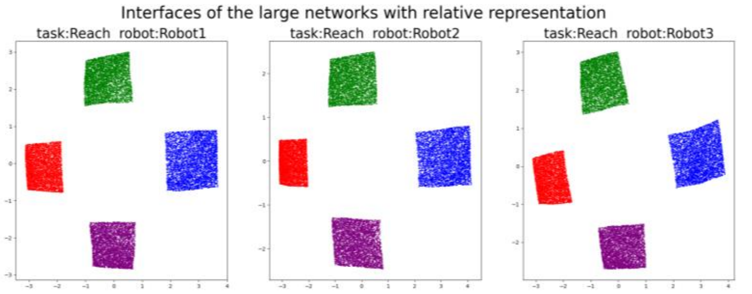
front back left right



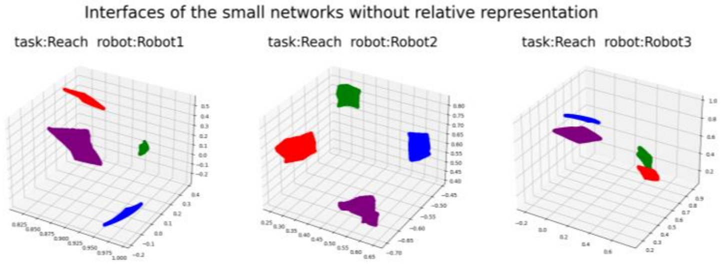
Ours with relative representation



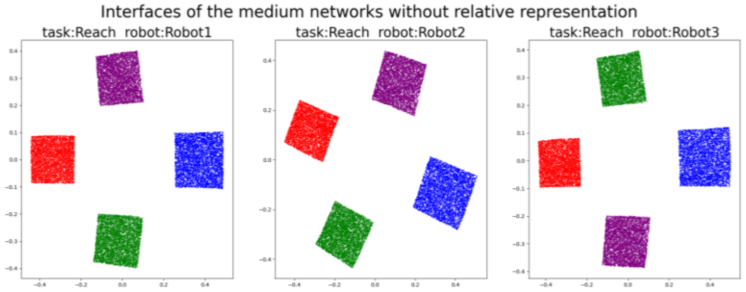
Ours with relative representation



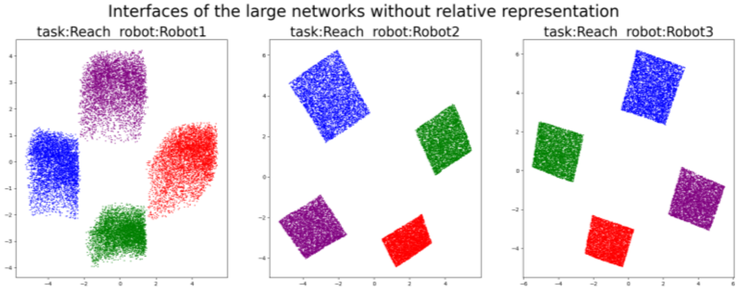
Ours with relative representation



Ablation method without relative representation



Ablation method without relative representation



Ablation method without relative representation

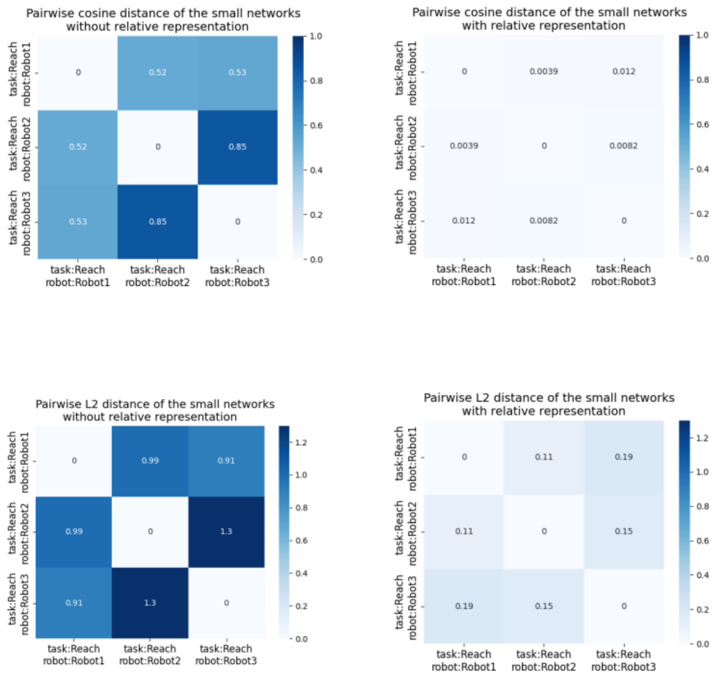
Small networks

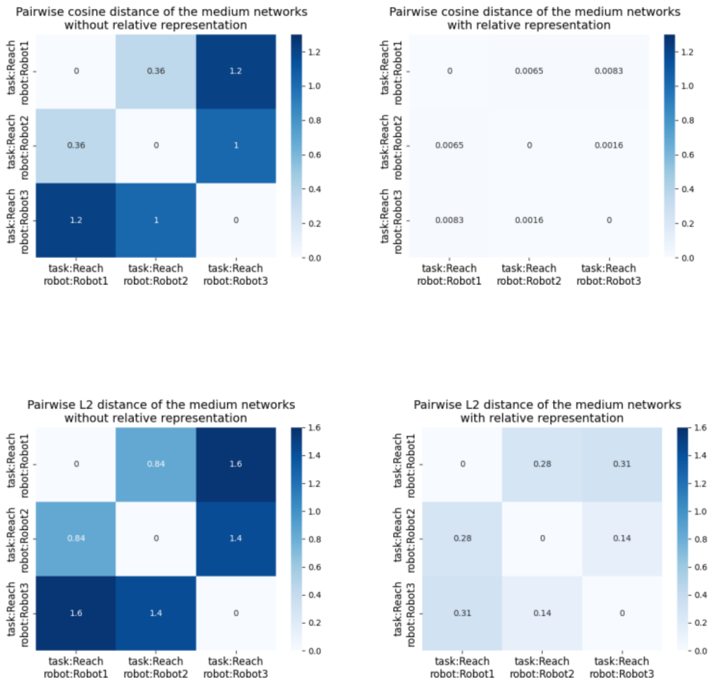
Medium networks

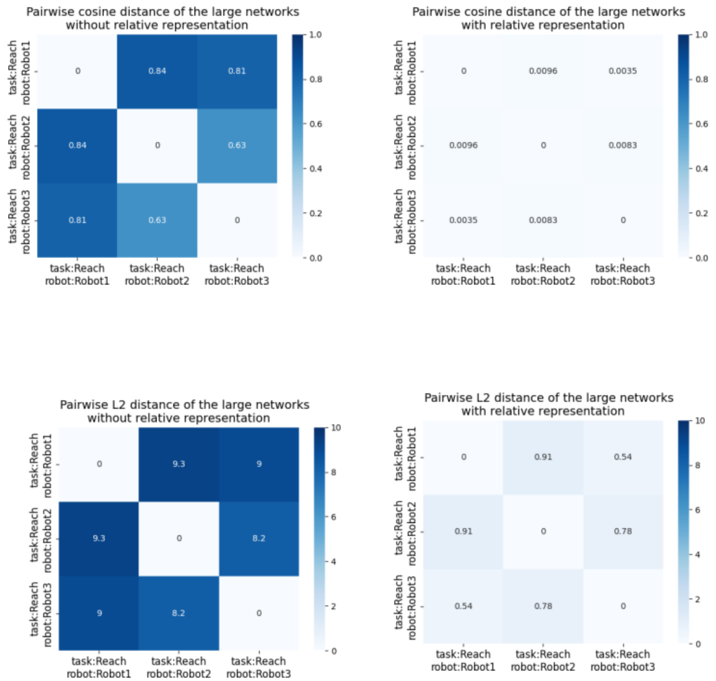
Large networks

Pairwise distance

Ablation	Ours	Ablation	Ours	Ablation	Ours





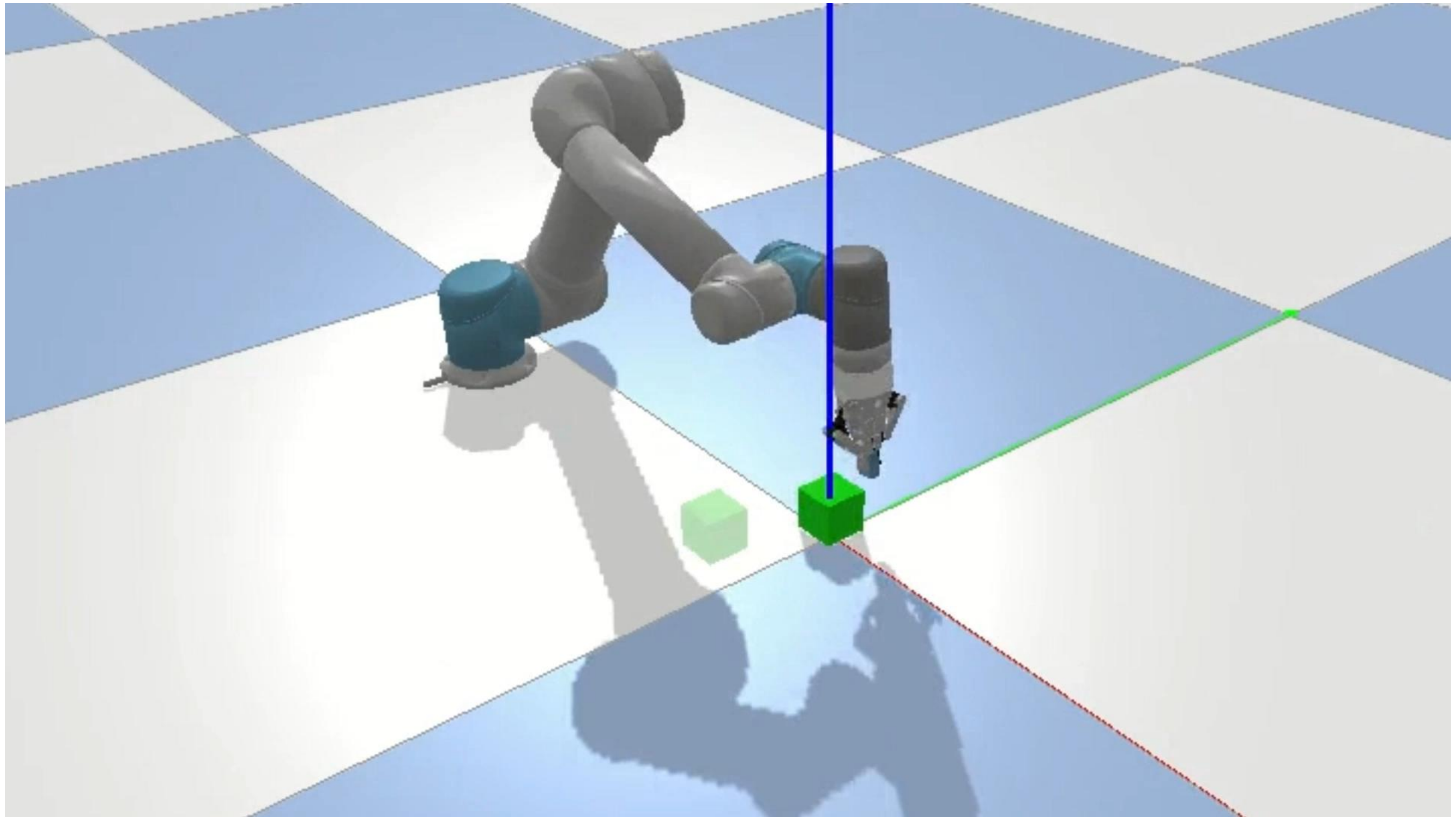


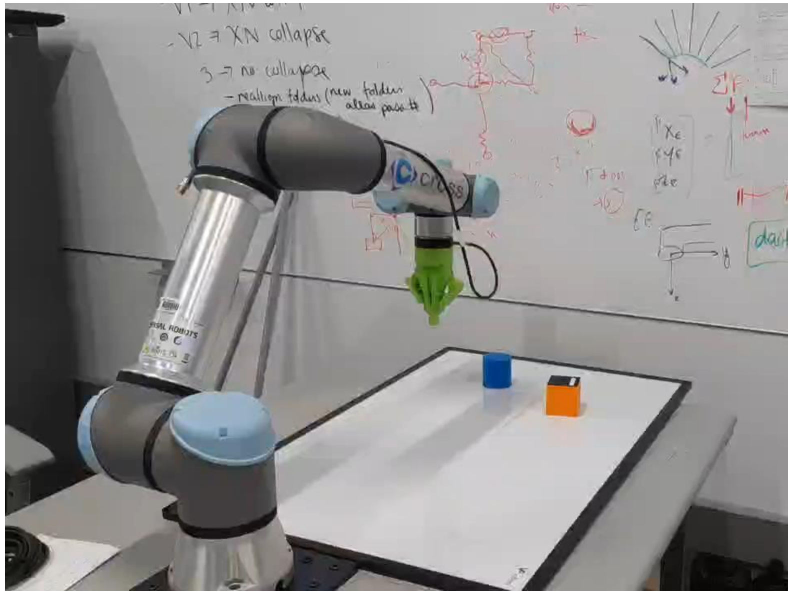
	Small networks		Medium networks		Large networks	
	Cosine distance	**L2 distance**	**Cosine distance**	**L2 distance**	**Cosine distance**	**L2 distance**
Without relative representation:	0.633±0.153	1.066±0.165	0.865±0.367	1.275±0.311	0.865±0.367	1.275±0.311
With relative representation:	0.0082±0.0035	0.149±0.032	0.0055±0.0029	0.240±0.075	0.0055±0.0029	0.240±0.075

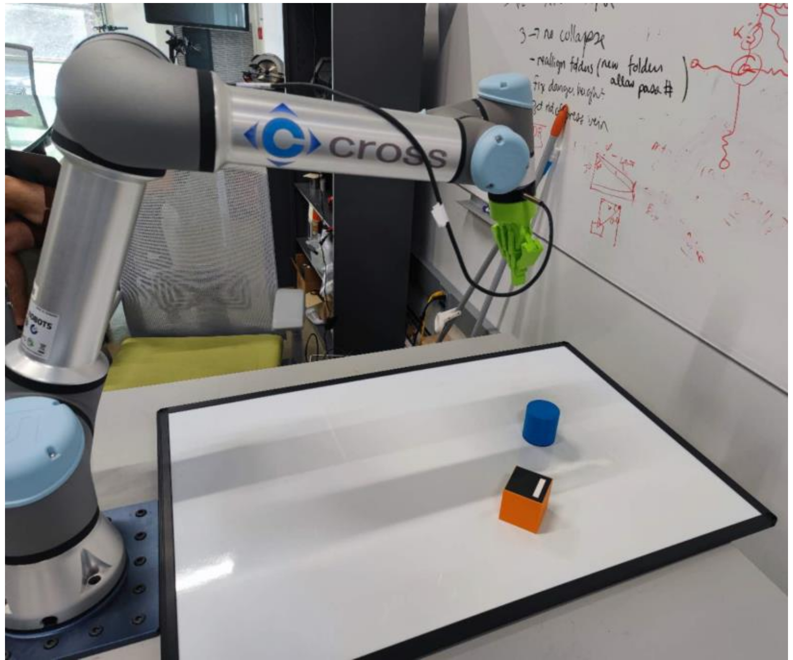
Conclusion

- Train a modular neural network in different environments or with different random seeds, the interfaces of the modules have an isometric transformation relation.
- Our proposed modular network with relative representation eliminates this transformation and ensures the network interfaces to have the same format.
- Our proposed method improves the zero-shot transfer performance in different tasks and accelerates the few-shot transfer learning process.

Ongoing real-world experiments





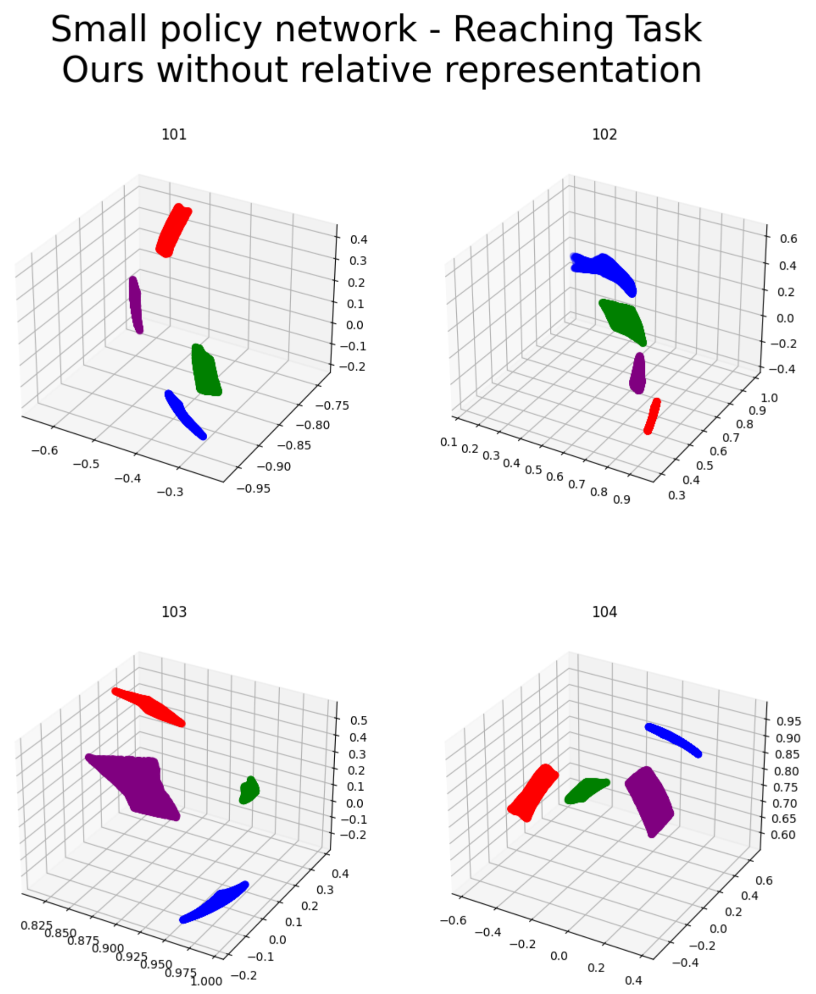


Thank you!

Further analysis of interface w/o relative representation

Seed 101 and 103 have a more similar interface distribution than seed 101 and 102. If we stitch 101 with 103, will it have better performance than stitching 101 and 102?

Conclusion: Yes, it does have better performance!



Ablation method without relative representation

Further analysis of interface w/o relative representation

Pairwise distance between seed 101 and 103:

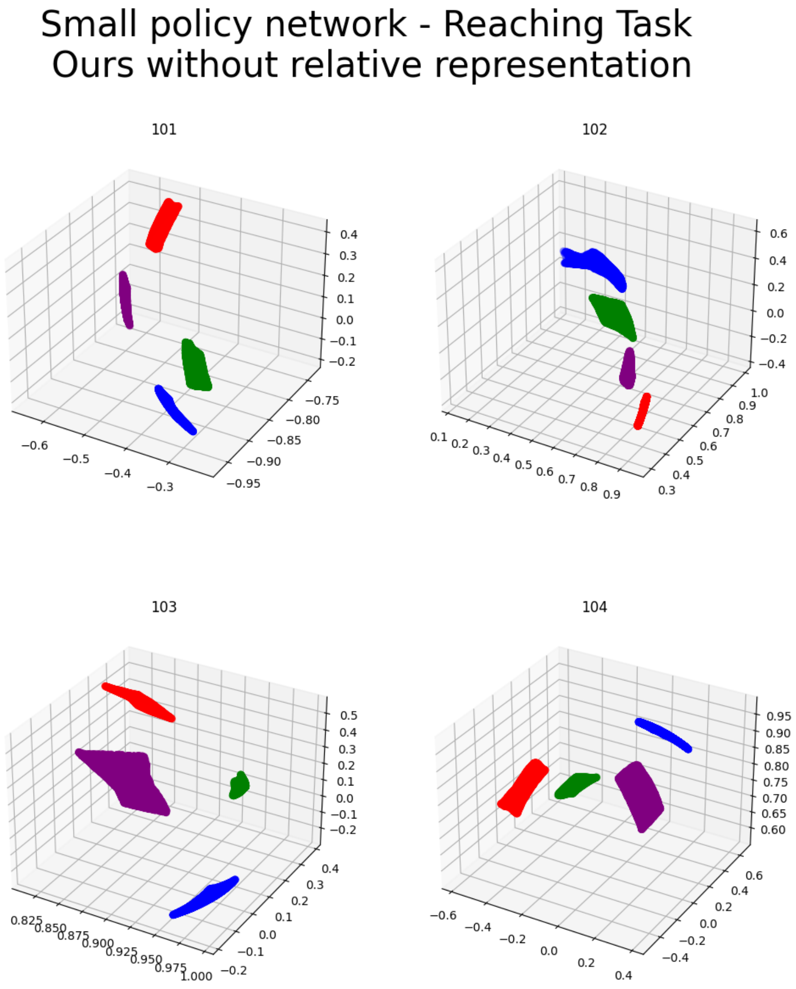
Cosine distance: 1.437

L2 distance: 1.693

Pairwise distance between seed 101 and 102:

Cosine distance: 1.820

L2 distance: 1.905



Ablation method without relative representation

Further analysis of interface w/o relative representation

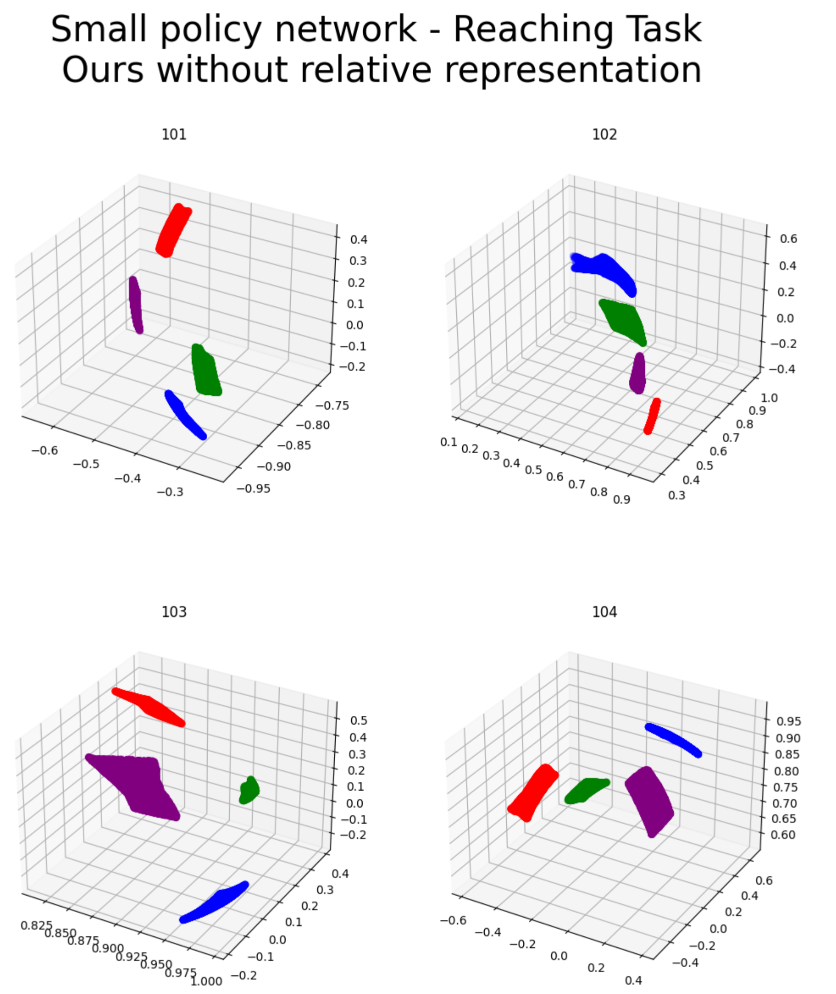
Success rate on reaching task when stitching seed 101 and 103:

0.222±0.0125

Success rate on reaching task when stitching seed 101 and 102:

0.129±0.0208

- Data calculated with mean and std of 5 sets of experiments. Each set has 200 reaching games.



Ablation method without relative representation