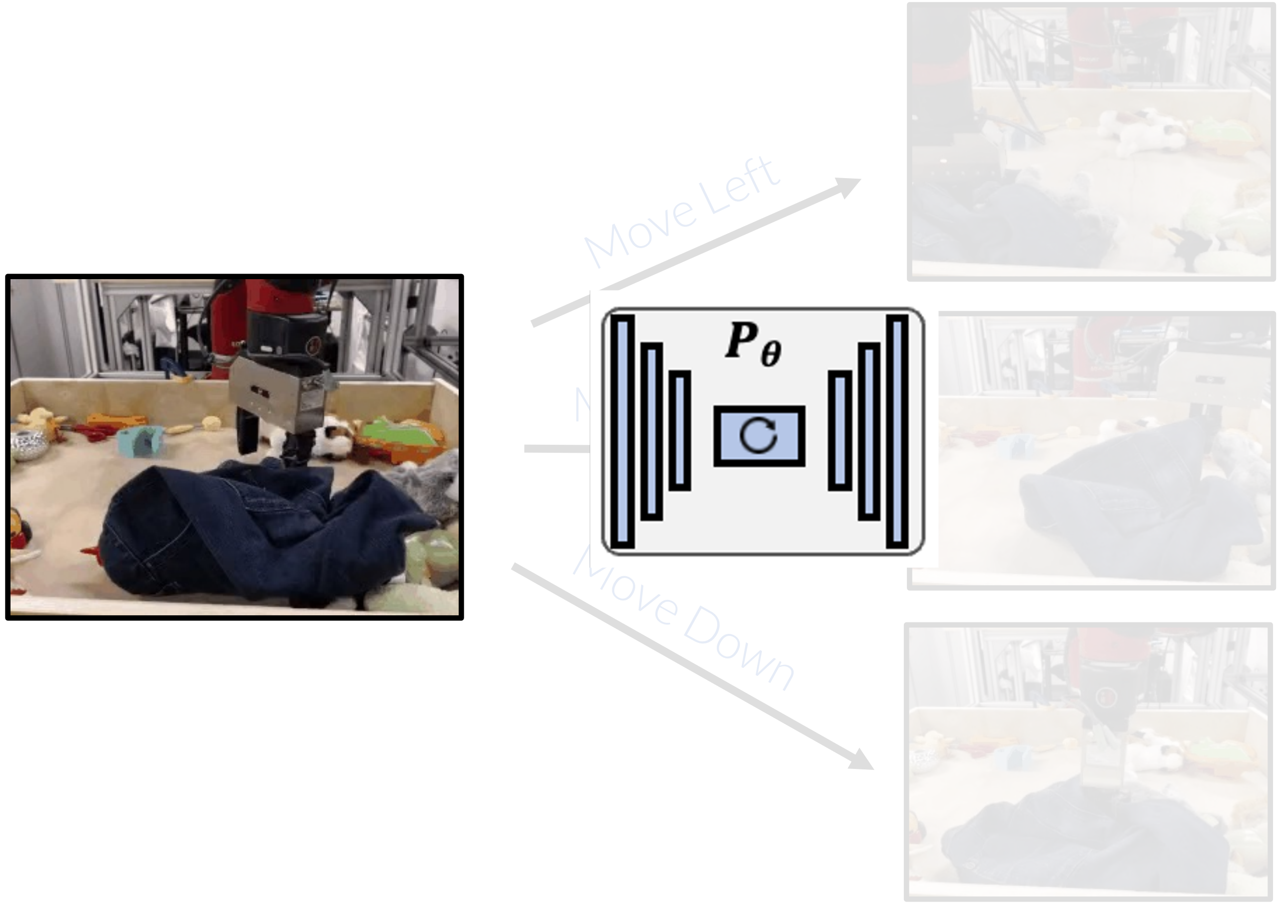
Know Thyself: Transferable Visual Control Policies Through Robot-Awareness

Edward S. Hu, Kun Huang, Oleh Rybkin, Dinesh Jayaraman

Dept. of CIS, GRASP Lab, University of Pennsylvania

ICLR 2022

Learning Visual Dynamics Models for Model-based RL



Learn to predict future video frames given actions.

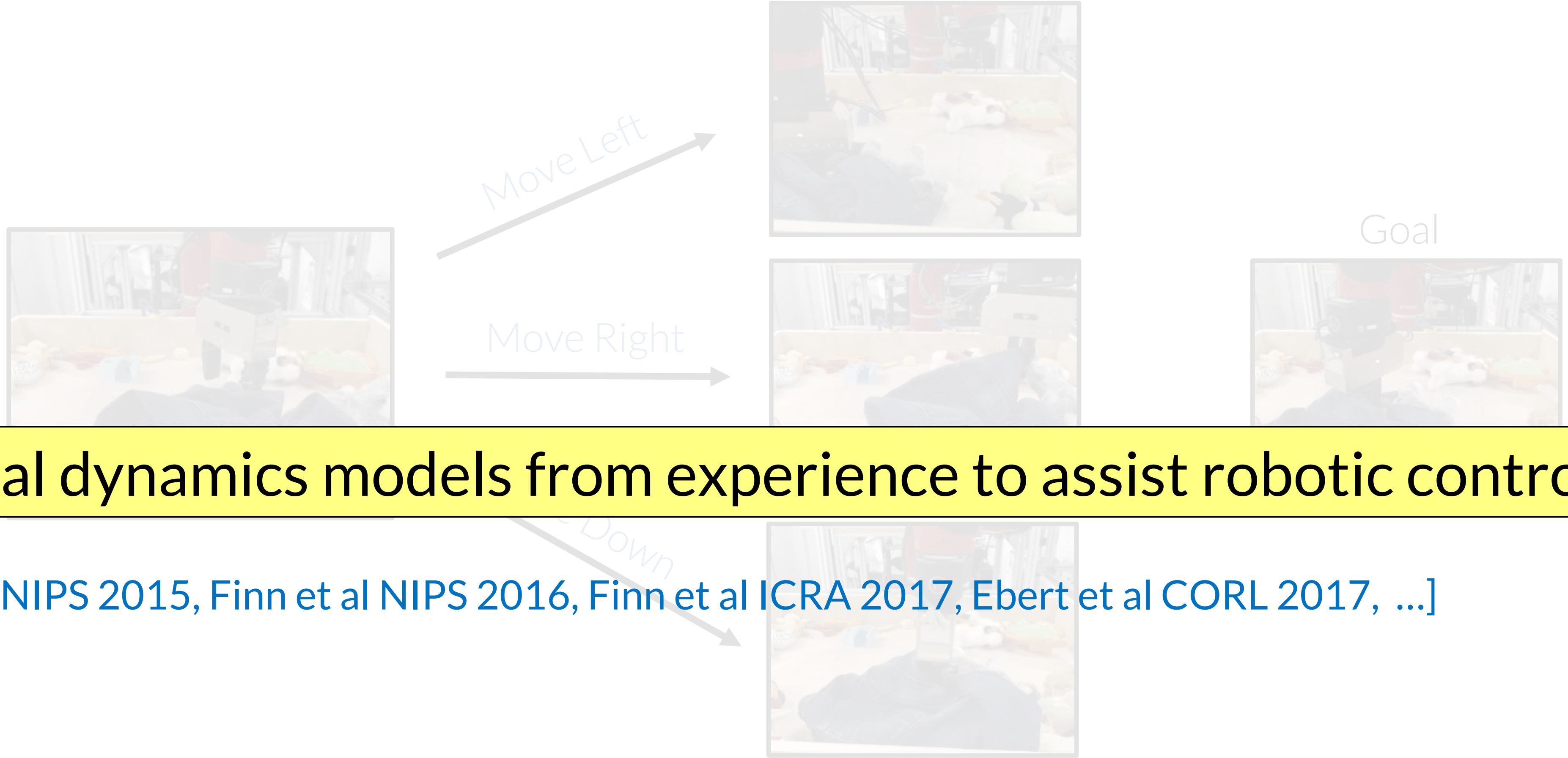
Learning Visual Dynamics Models for Model-based RL

Learn to predict future video frames given actions.

Test time:

1. Sample actions and forecast their outcomes.
2. Select action with best forecasted outcome

Learning Visual Dynamics Models for Model-based RL



Learning visual dynamics models from experience to assist robotic control.

[Oh et al NIPS 2015, Finn et al NIPS 2016, Finn et al ICRA 2017, Ebert et al CORL 2017, ...]

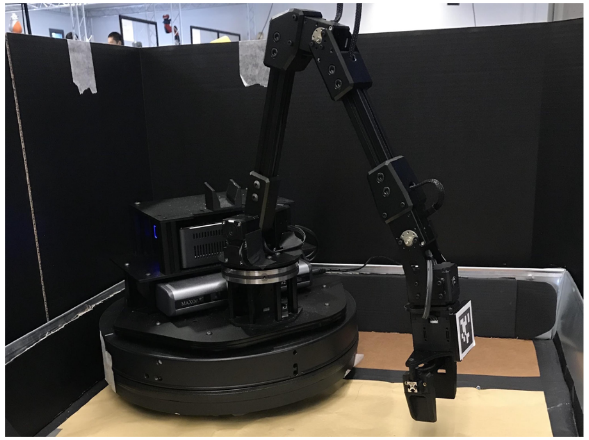
Learn to predict future video frames given actions.
Test time:

1. Sample actions and forecast their effects.
2. Select action with best forecasted outcome

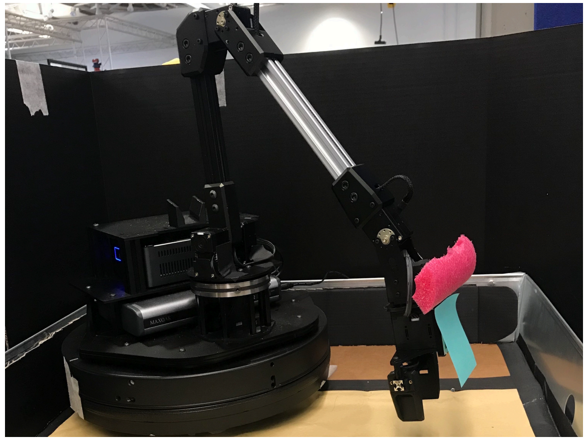
Visual models don't transfer to new robots

- Collecting robot data to train visual models is slow, up to a few days

Train on WidowX



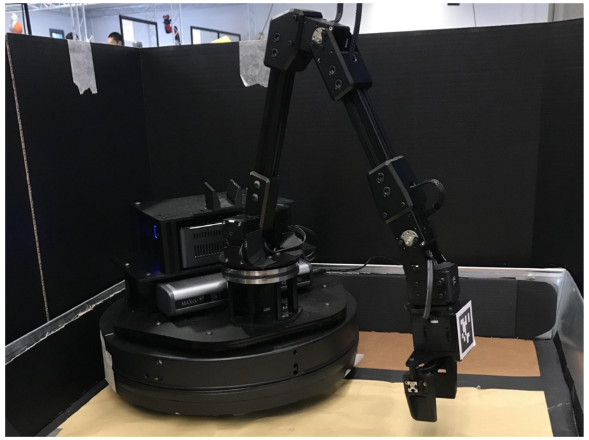
Test on Modified WidowX



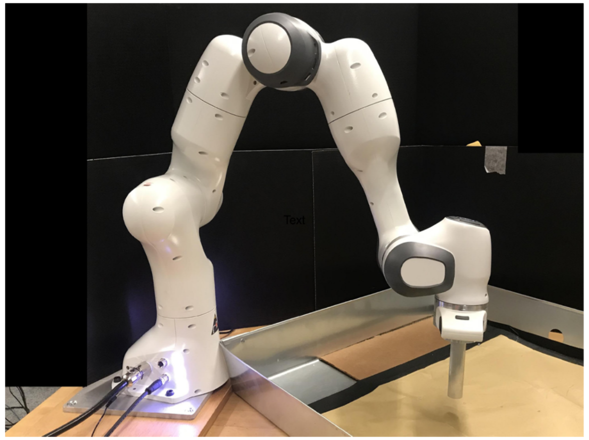
Visual models don't transfer to new robots

- Collecting robot data to train visual models is slow, up to a few days

Train on WidowX



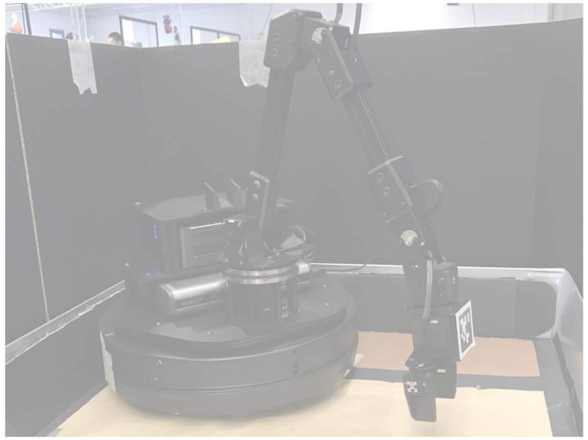
Test on Franka Panda



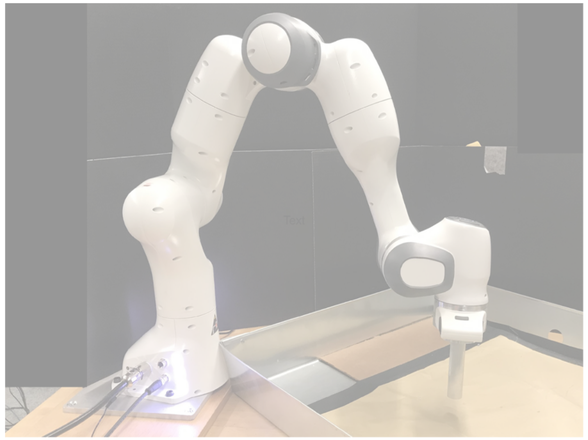
Visual models don’t transfer to new robots

- Collecting robot data to train visual models is slow, up to a few days

Train on WidowX



Test on Franka Panda



Poor Generalization

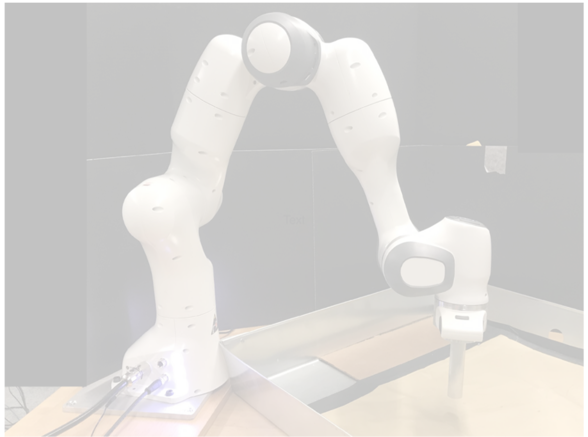
Visual models don’t transfer to new robots

- Collecting robot data to train visual models is slow, up to a few days
- Do not transfer across robots out of the box, and must typically be trained separately for every new robot

Could we learn *transferable* visual dynamics models and controllers?

Train on WidowX

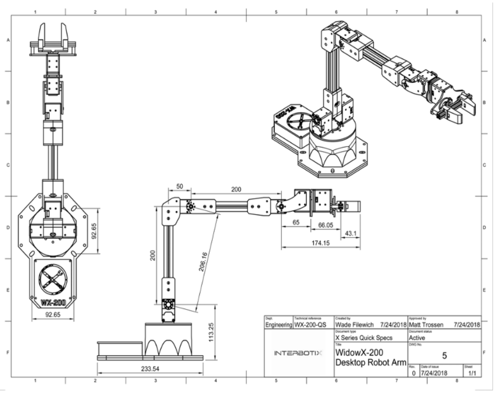
Test on Franka Panda

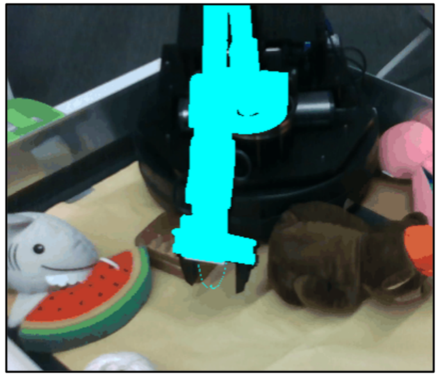


Poor Generalization

Using Robot Knowledge to Improve Transfer

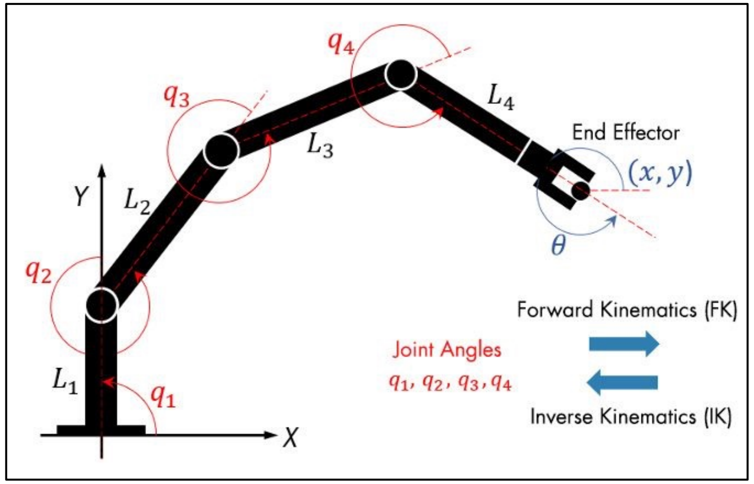
Robot Segmentation





Easy when proprioception and camera calibration are available

Robot Dynamics



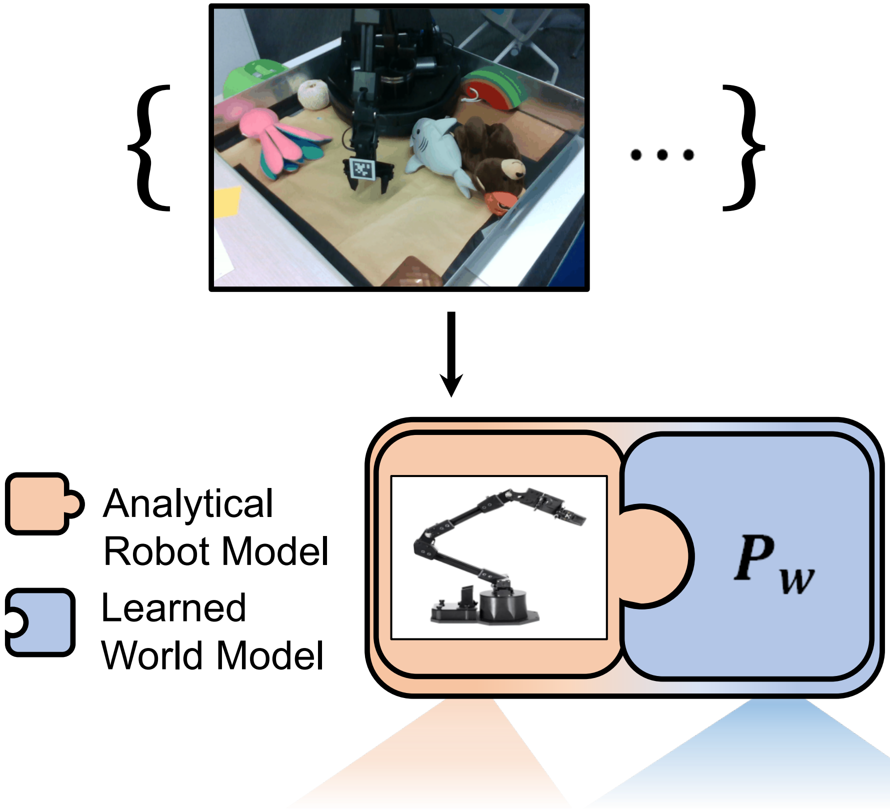
$$P_R(r' | r, a)$$

The analytical kinematics readily specify the robot transition model*

*assuming the robot does not hit any immovable barriers.

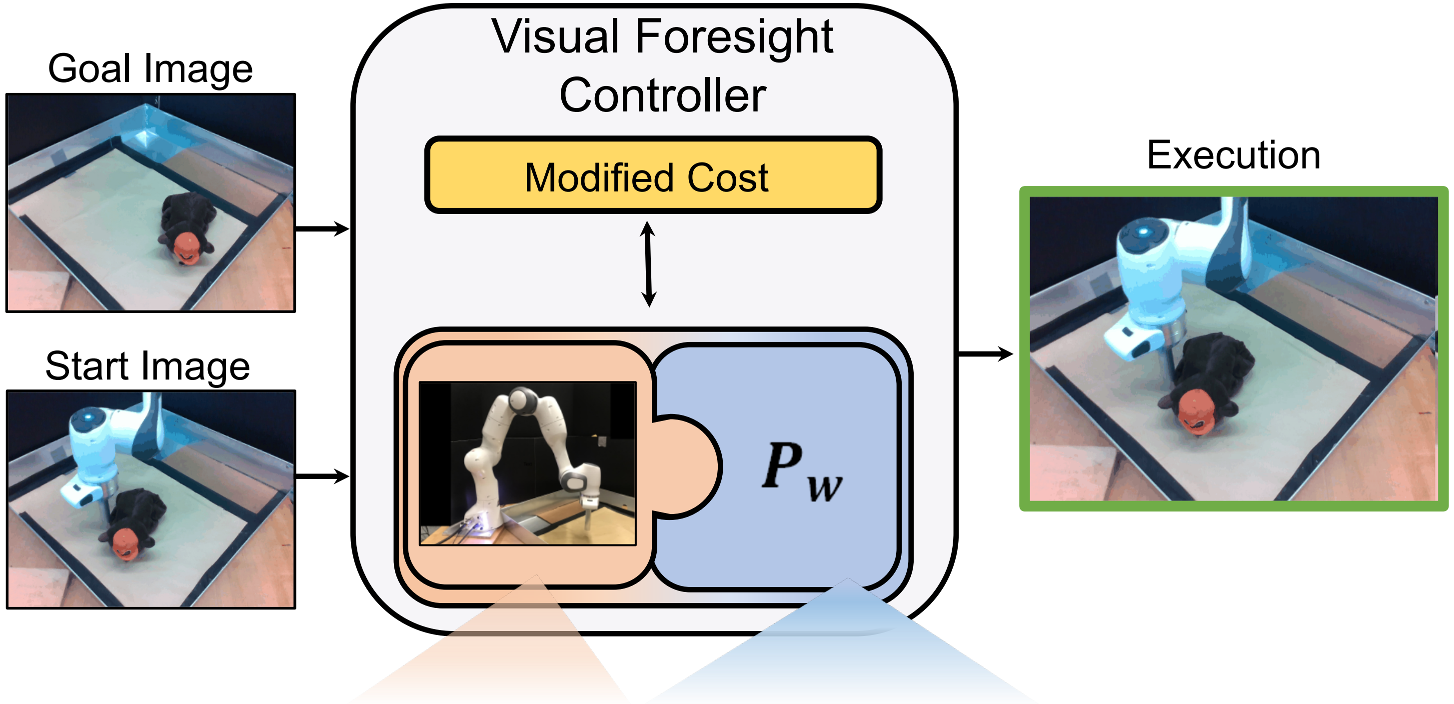
Robot-Aware Control

Idea: Disentangle the robot from the stuff we want to move in the world.



$$P(o' \mid o, a) = P_{R1}(r' \mid r, a)\, P_w(o'_w \mid o_w, r, r'\, a)$$

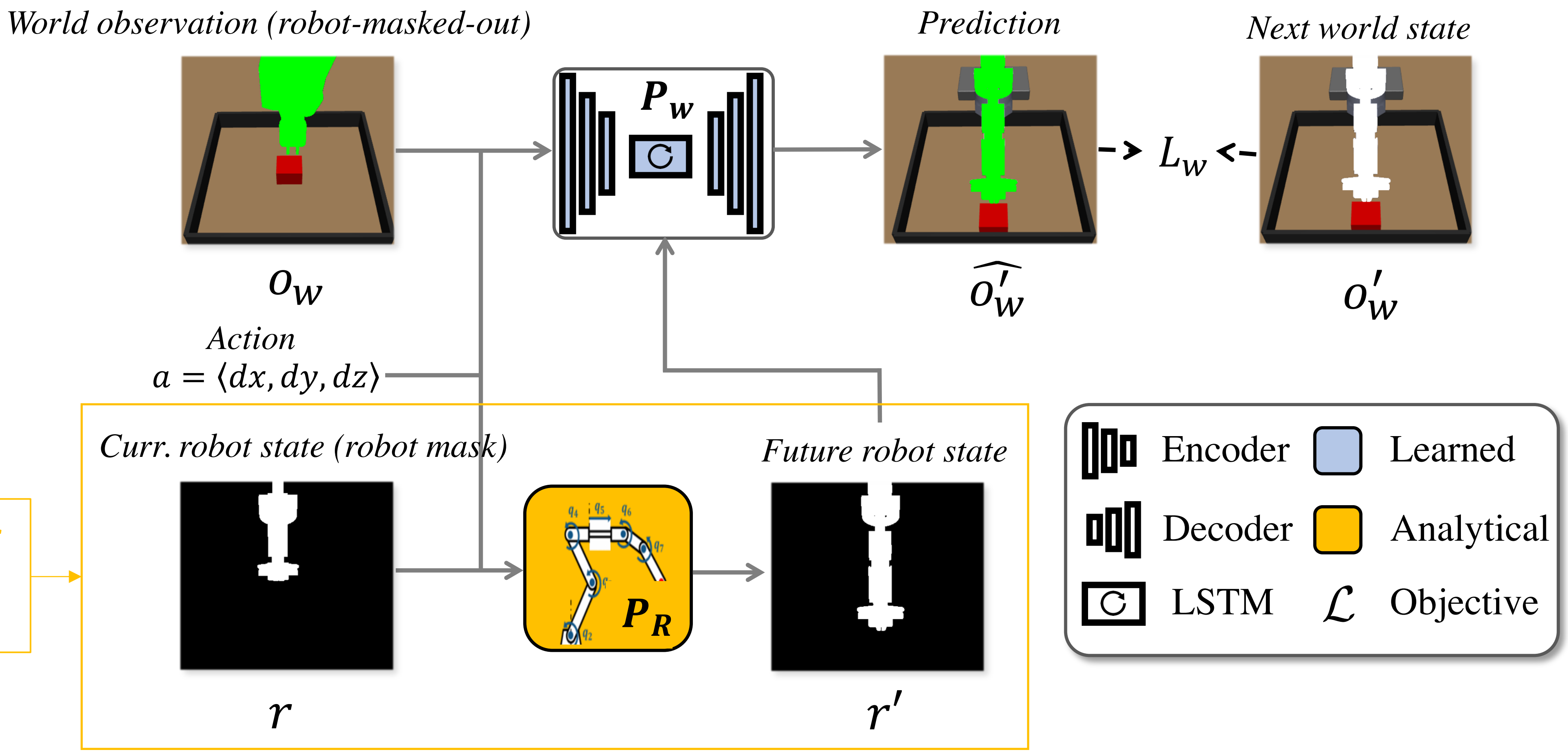
Step 1: Learn robot-aware world model from videos of robot R1



$$P(o' \mid o, a) = P_{R2}(r' \mid r, a)\, P_w(o'_w \mid o_w, r, r'\, a)$$

Step 2: Zero-shot prediction / control with robot R2

RAC Component #1: Robot-Aware World Dynamics Model



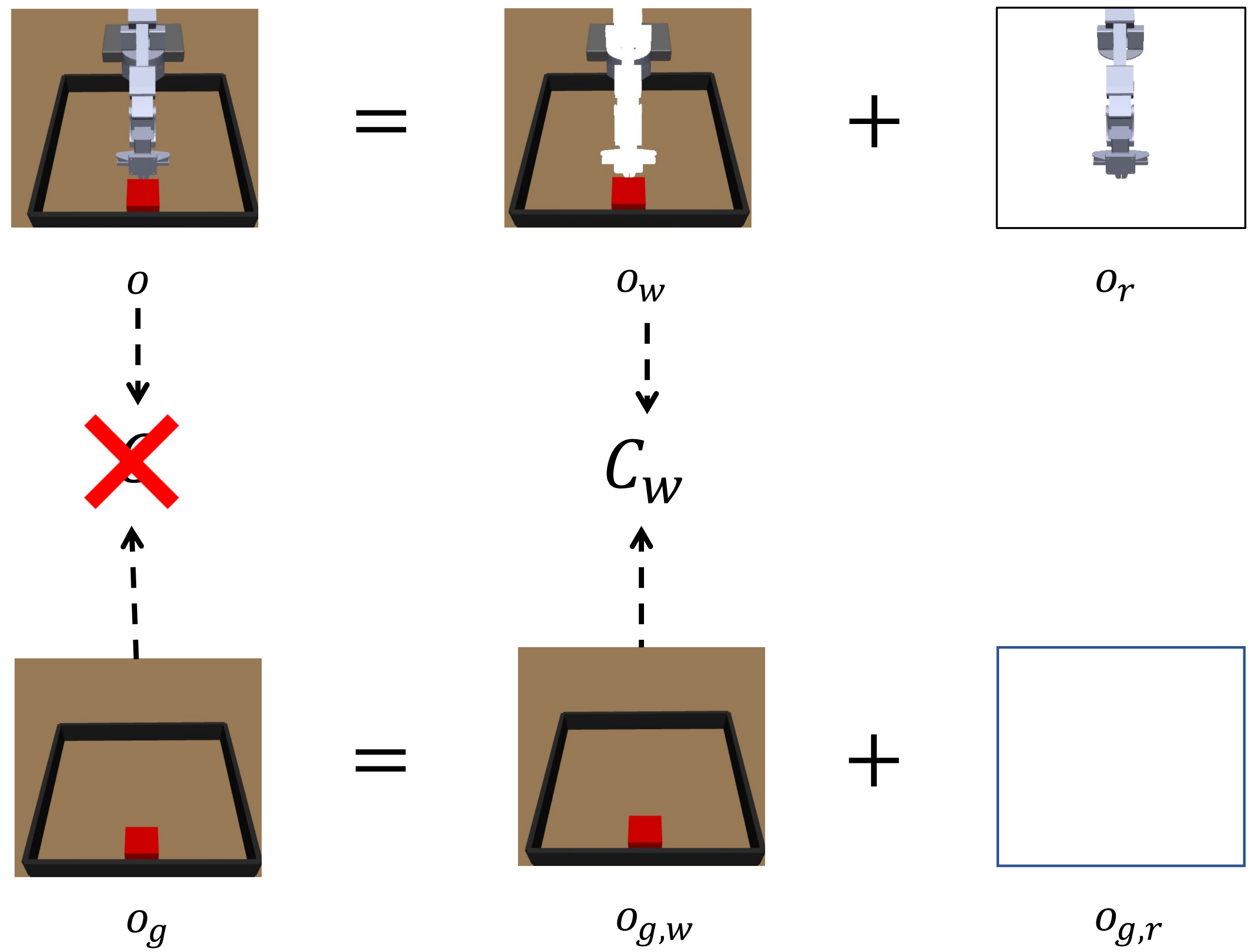
Can swap out for another robot*

$$P(o' \mid o, a) = P_{R1}(r' \mid r, a) P_w(o'_w \mid o_w, r, r' a)$$

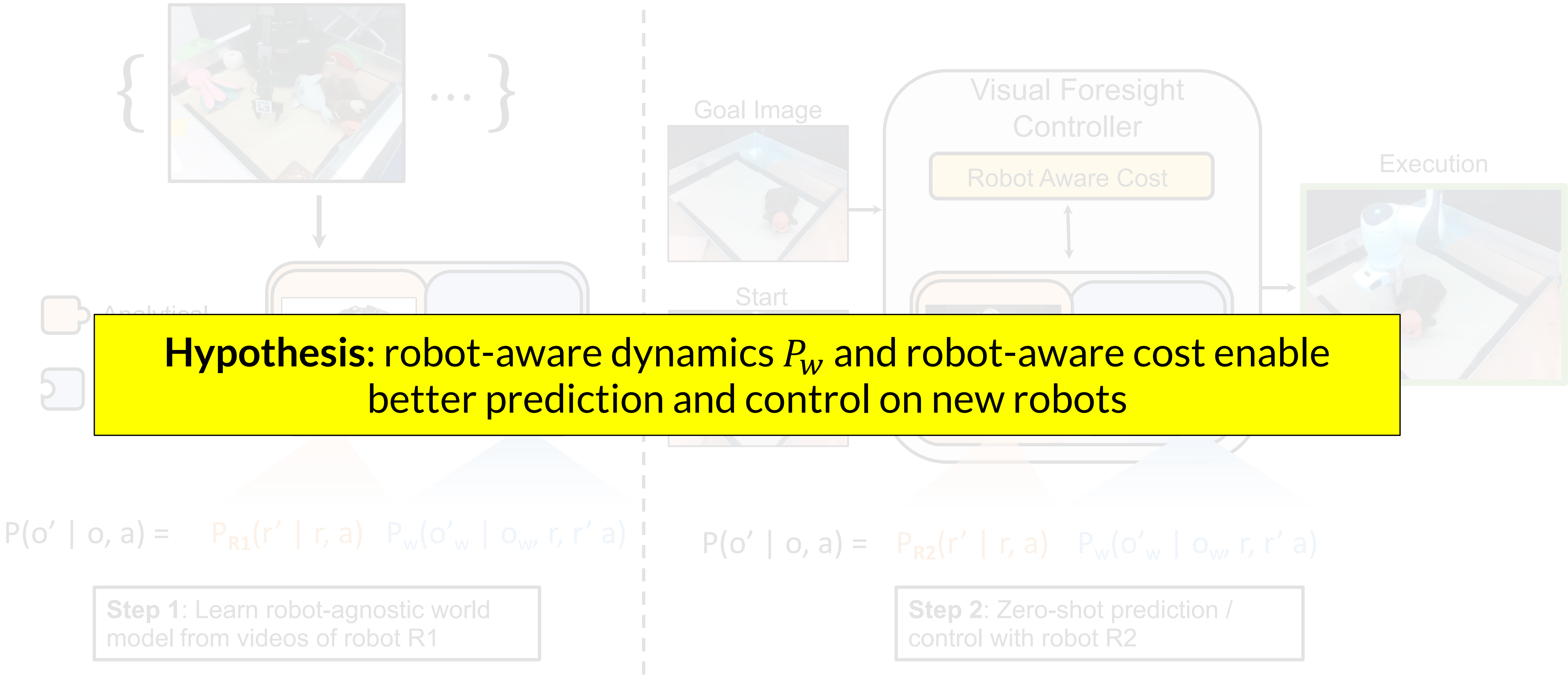
*Assuming similar end-effectors

RAC Component #2: Robot-Aware Planning Cost

Need a function to measure whether predicted outcomes from an action plan match goal image(s).



Robot-Aware Control Summary

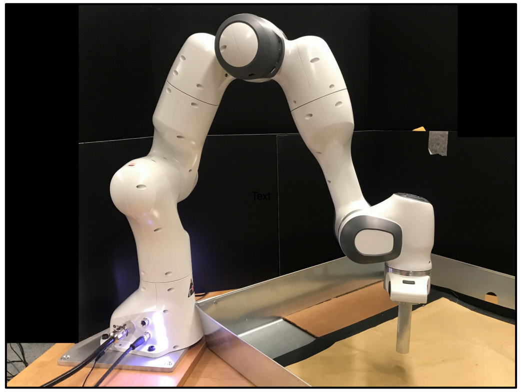


Hypothesis: robot-aware dynamics P_w and robot-aware cost enable better prediction and control on new robots

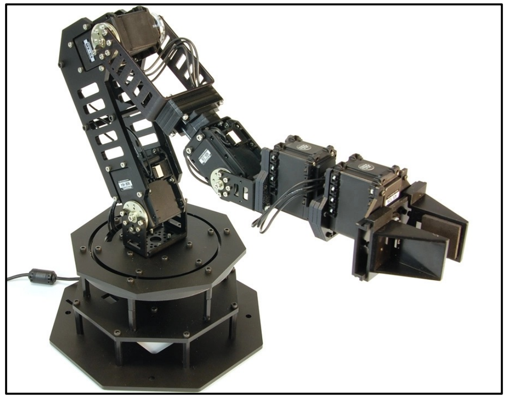
Experimental Validation on Robot Transfer Tasks

Sawyer

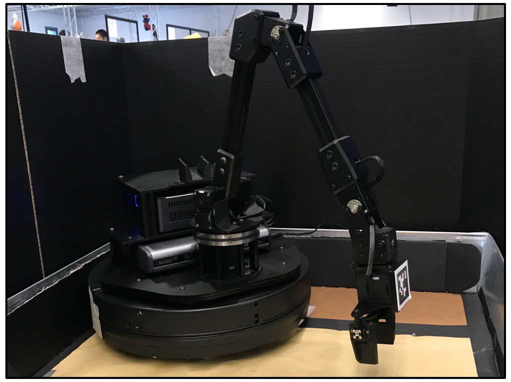
Baxter



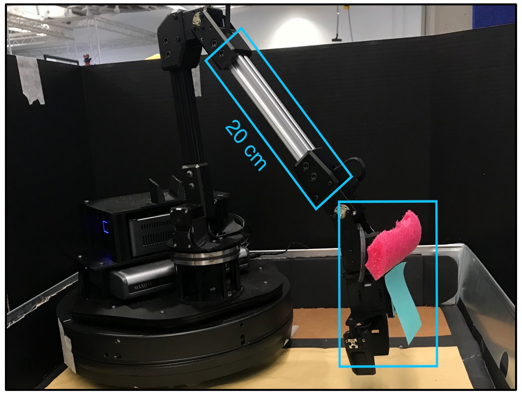
Franka Panda



WidowX

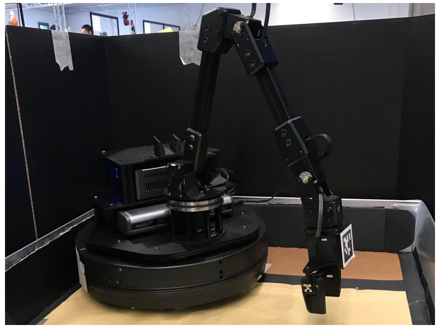


WidowX200

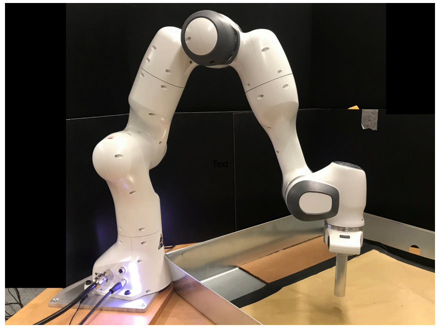


Modified
WidowX200

Zero-shot WidowX to Franka Pushing

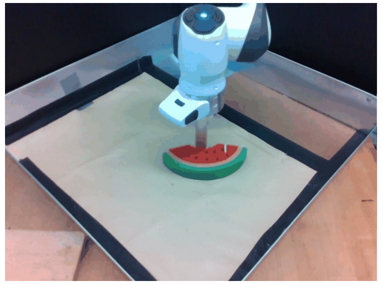


WidowX200
~2k videos

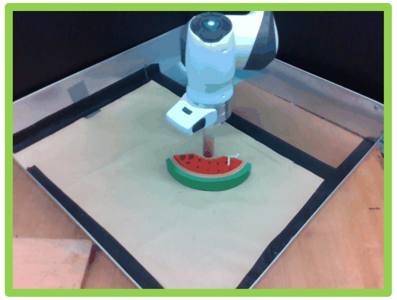


Franka

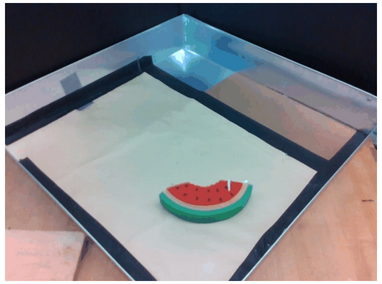
Baseline



RAC



Goal



Results: Zero-Shot Transfer Quantitative Summary

Dynamics Model	Cost	Fetch Push (Sim.)	Fetch Pick-and-place (Sim.)	Franka Push (Real)
VF+State	Pixel	0/20 (0%)	0/20 (0%)	0/30 (0%)
RA	RA	**18/20** (90%)	**8/20** (40%)	**22/30** (71%)

From training on 1 robot, RAC outperforms:

- Visual Foresight baseline with no RA-model and no RA-cost

Results: Zero-Shot Transfer Quantitative Summary

Dynamics Model	Cost	Fetch Push (Sim.)	Fetch Pick-and-place (Sim.)	Franka Push (Real)
CycleGAN+VF+State	Pixel	12/20 (60%)	0/20 (0%)	-
VF+State	Pixel	0/20 (0%)	0/20 (0%)	0/30 (0%)
RA	RA	**18/20** (90%)	**8/20** (40%)	**22/30** (71%)

From training on 1 robot, RAC outperforms:

- Visual Foresight baseline with no RA-model and no RA-cost
- Domain Adaptation baseline with privileged access to 12K images of test robot

Results: Zero-Shot Transfer Quantitative Summary

Dynamics Model	Cost	Fetch Push (Sim.)	Fetch Pick-and-place (Sim.)	Franka Push (Real)
CycleGAN+VF+State	Pixel	12/20 (60%)	0/20 (0%)	-
VF+State	Pixel	0/20 (0%)	0/20 (0%)	0/30 (0%)
RA	Pixel	0/20 (0%)	0/20 (0%)	0/30 (0%)
VF+State	RA	12/20 (60%)	4/20 (20%)	6/30 (20%)
RA	RA	**18/20** (90%)	**8/20** (40%)	**22/30** (71%)

From training on 1 robot, RAC outperforms:

- Visual Foresight baseline with no RA-model and no RA-cost
- Domain Adaptation Baseline with privileged access to 12K images of test robot
- Ablations of model and cost

Results: Zero-Shot Transfer Quantitative Summary

Dynamics Model	Cost	Fetch Push (Sim.)	Fetch Pick-and-place (Sim.)	Franka Push (Real)
CycleGAN+VF+State	Pixel	12/20 (60%)	0/20 (0%)	-
VF+State	Pixel	0/20 (0%)	0/20 (0%)	0/30 (0%)
RA	Pixel	0/20 (0%)	0/20 (0%)	0/30 (0%)
VF+State	RA	12/20 (60%)	4/20 (20%)	6/30 (20%)
RA	RA	**18/20** (90%)	**8/20** (40%)	**22/30** (71%)
VF+State (Multi-robot)	Pixel	-	-	11/30 (36%)
RA (Multi-robot)	RA	-	-	**27/30** (90%)

From training on 1 robot, RAC outperforms:

- Visual Foresight baseline with no RA-model and no RA-cost
- Domain Adaptation Baseline with privileged access to 12K images of test robot
- Ablations of model and cost

Trends hold even after training on multiple robots!

More Experiment Visualizations

Robot-aware model predicts object movement more accurately

Baseline model

RA model

Goal

Takeaway:

Spatially disentangling the robot from the rest of the world enables transferring visual policies to unseen new robots

www.edwardshu.com/rac for more info

Poster 6041

RAC is more prec
better prediction, crucial for pick-and-place.

Baseline policy

RAC policy

Goal

Achieving Human Goal Images

Human Goal