CS 188 - Deep Robotics

Michael Laskey

AUTOLAB

-4th yr Phd Student w/ Prof. Goldberg

-Focus on Statistical Theory for Robots

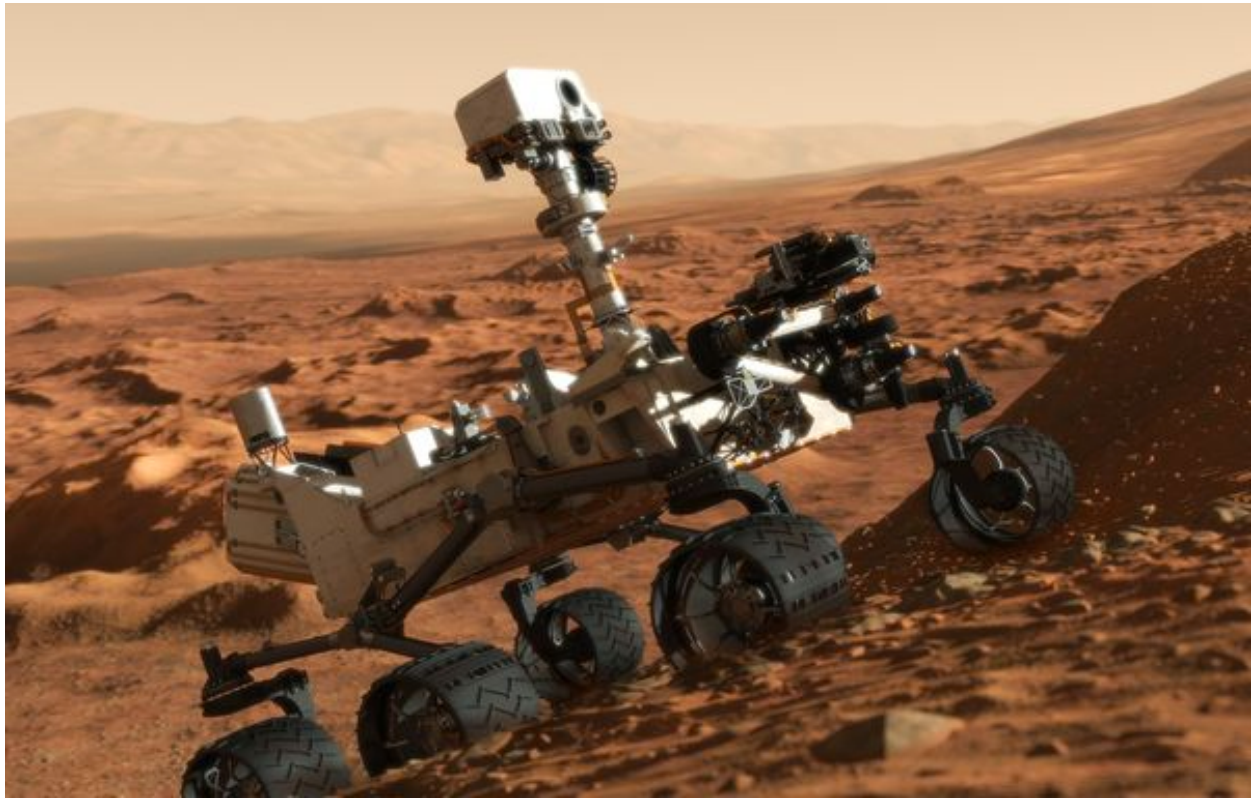
-Passion for Training Industrial Robots

Overview

-Robots of Today
-Robots of Tomorrow
-Directions to Take Us There
    -Physics
    -Unsupervised Learning
    -LfD
-Break
-Manipulation Case Studies
-Robot Debugging

Robots of Today

Field Robotics



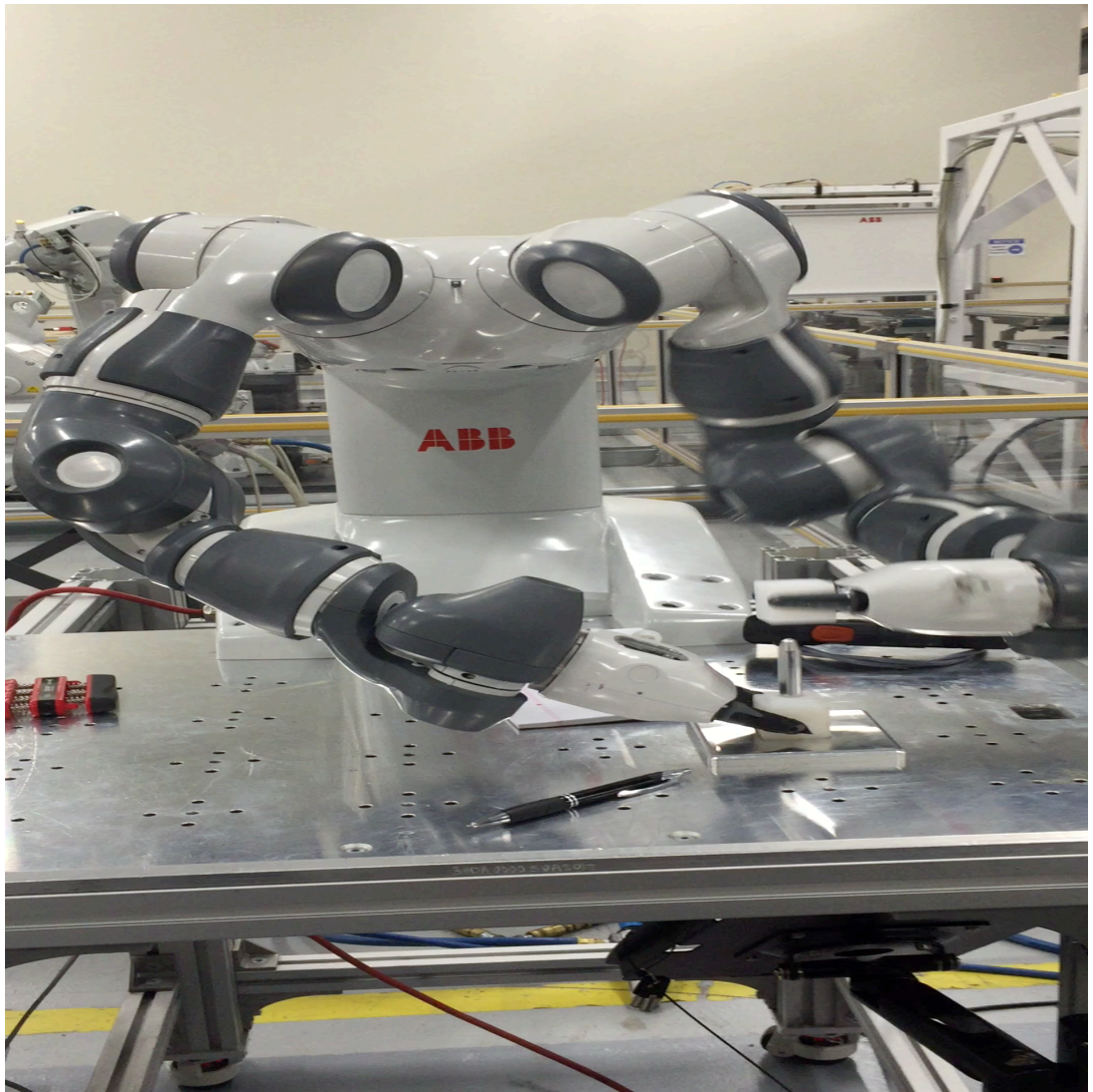
Industrial Robots

Medical Applications

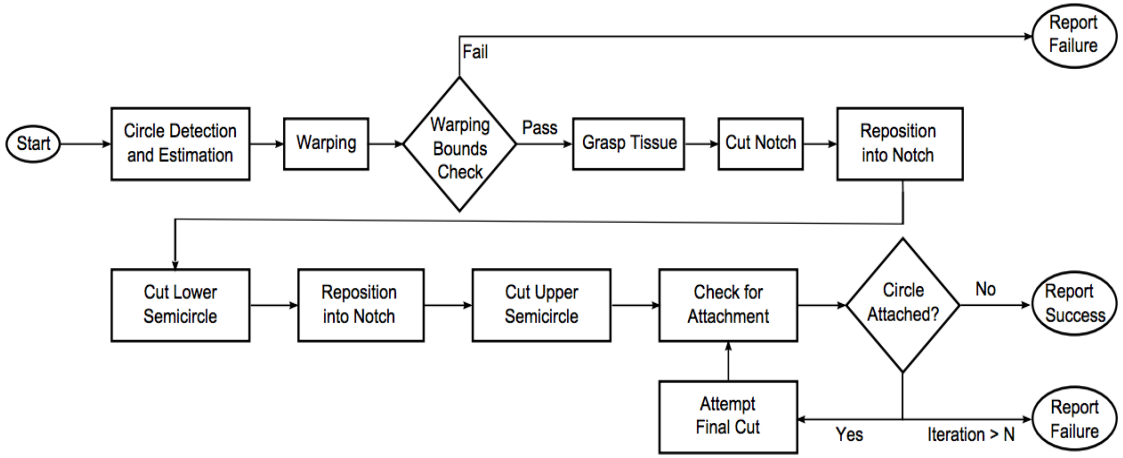
Home Robot

Programming Today's Robots

Motion Replay



Fine-State Machines

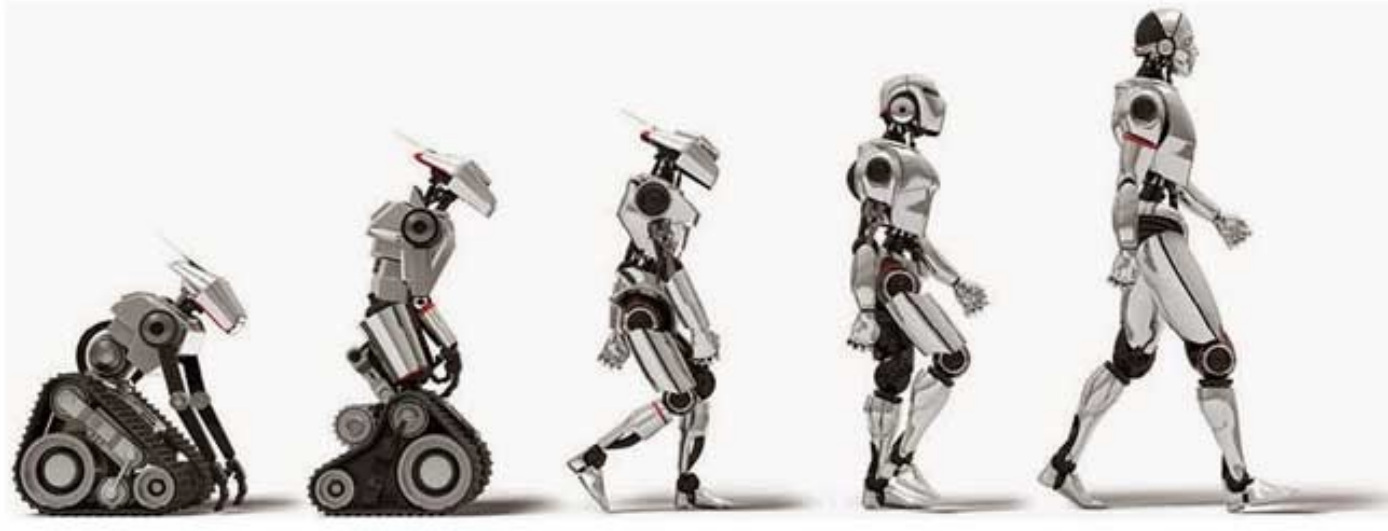


Goals for Robots of the Future

1) Generalize to Unseen Environments

2) Learn to Get Better over Time

3) Minimal Coding Needed



Common MDP for a Robot

Actions A

-Change in Pose or Motor Torques

State S

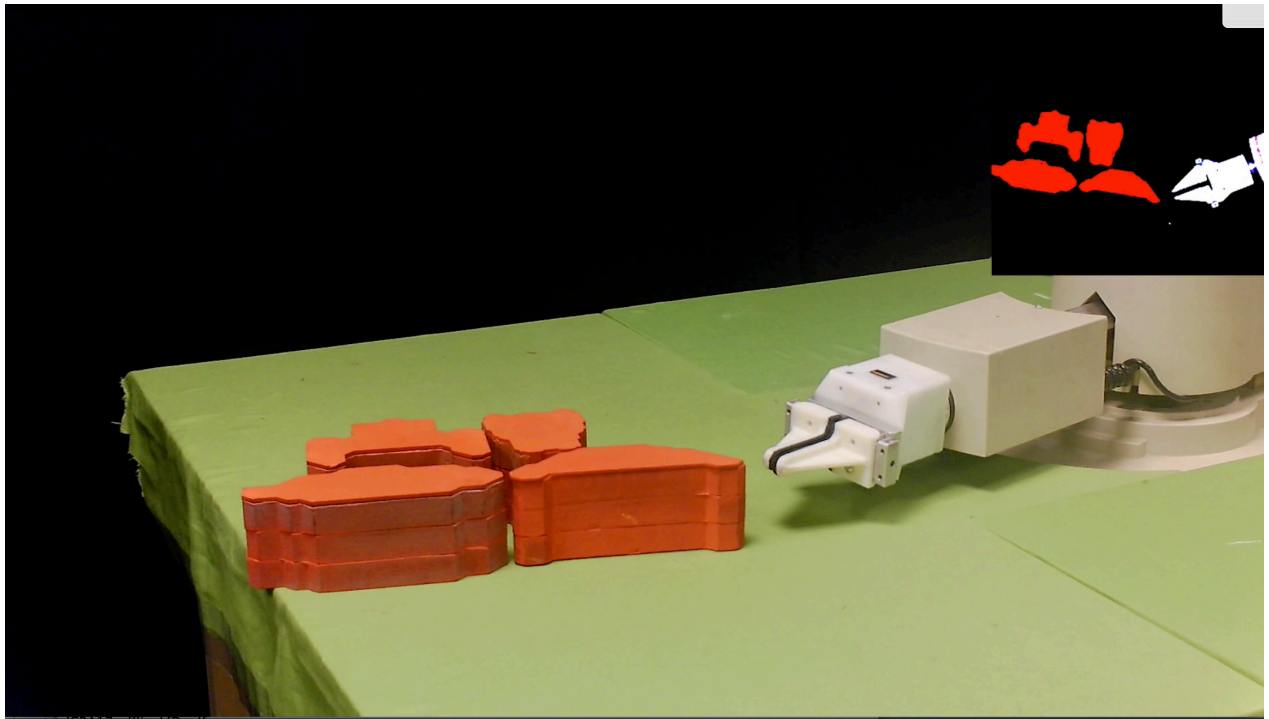
-Image of the Environment

Reward R(s,s’,a)

-Binary Success or Failed

Transition T(s,s’a)

-Real World (Physics)

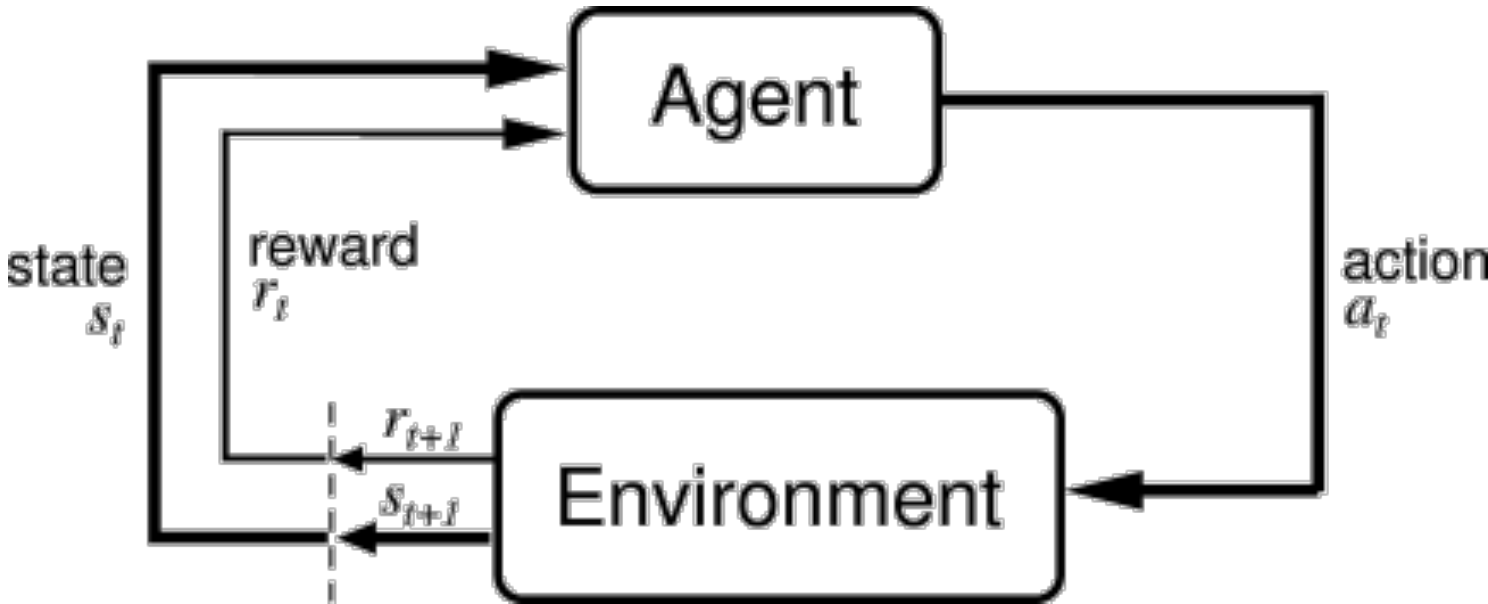


Research Directions

1) Model T(s,s',a)

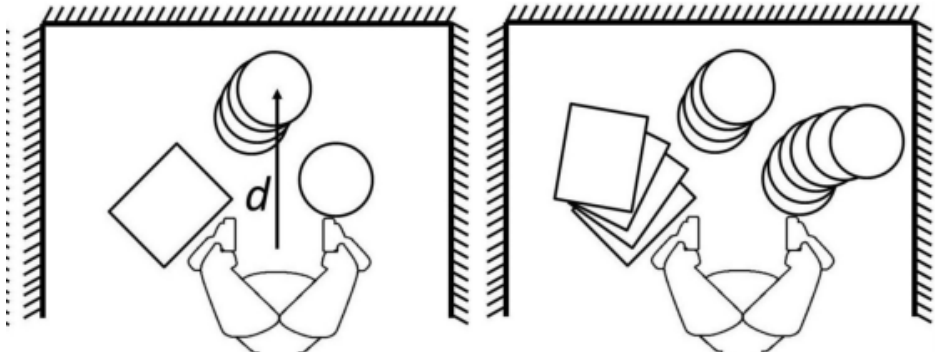
2) Reinforcement Learning

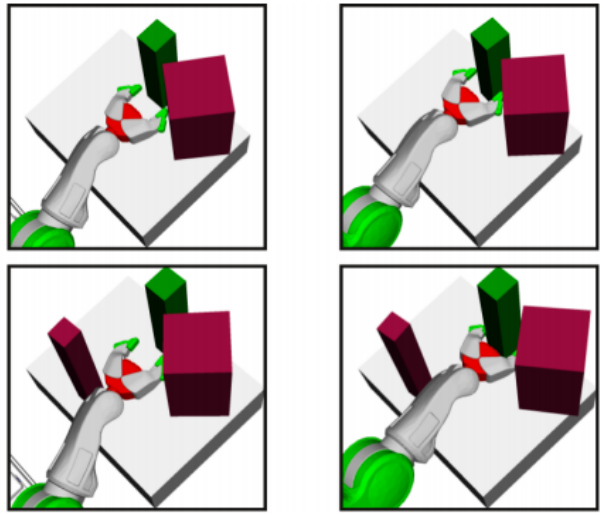
3) Learn from Humans

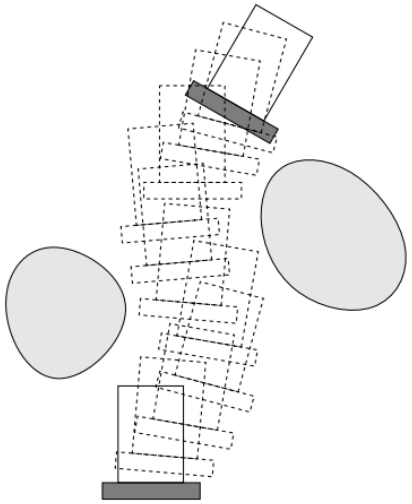


Idea:
Use Physics to ModelTransition Function

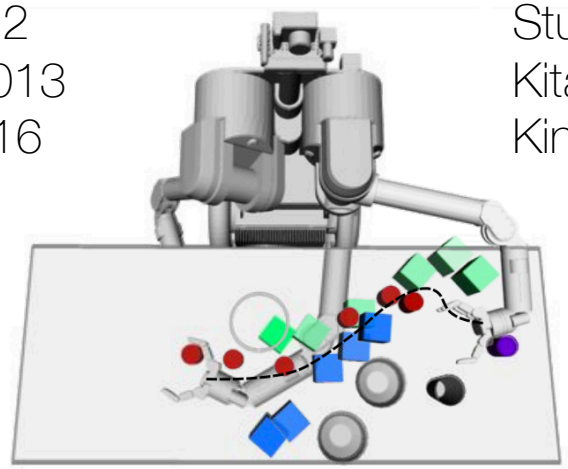
Grasping Related Work







Mason et al. 1989
Dogar et al. 2011
Dogar et al. 2012
Cosgun et al. 2013
Mahler et al. 2016

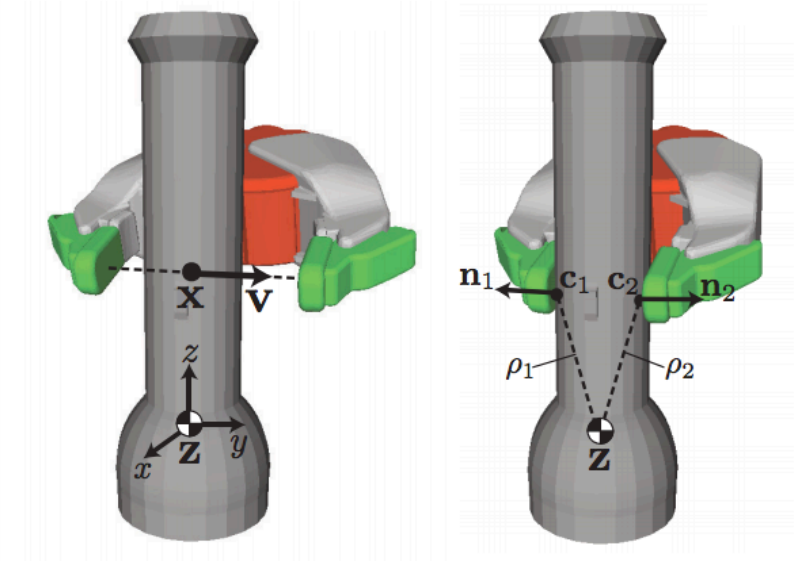


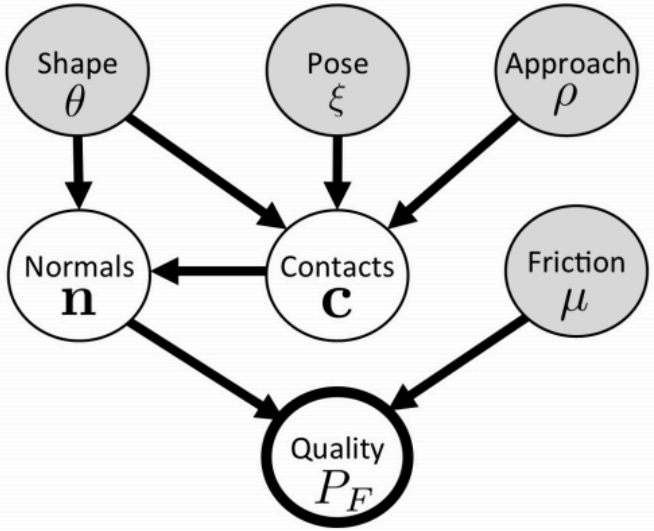
Lynch et al. 199
Van Den Berg et al. 2009
Stulp et al. 2011
Kitaev et al. 2015
King et al. 2015

Grasp Quality Metrics

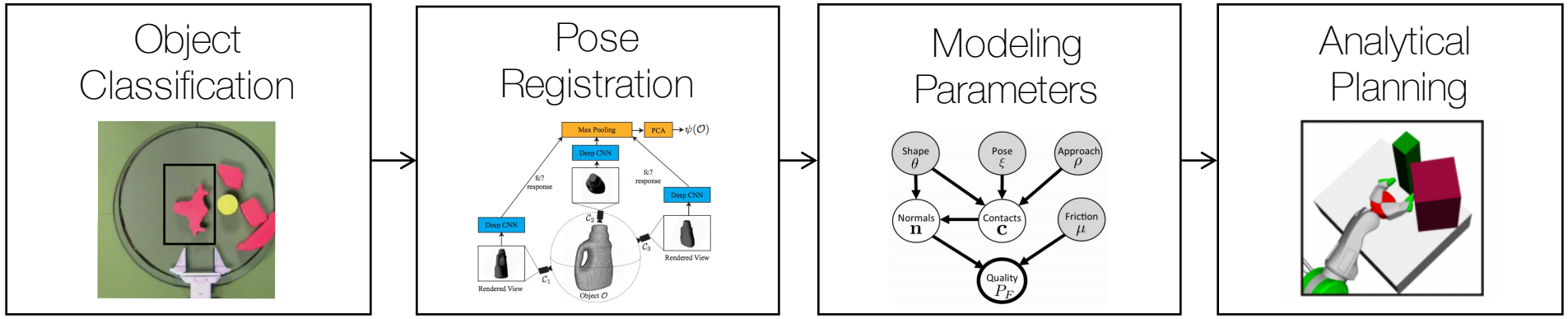
Given a set of proposed grasp , G, determine the grasp g*, where

$$g^* = \underset{g \in \mathcal{G}}{\operatorname{argmax}}\, P_F(g)$$



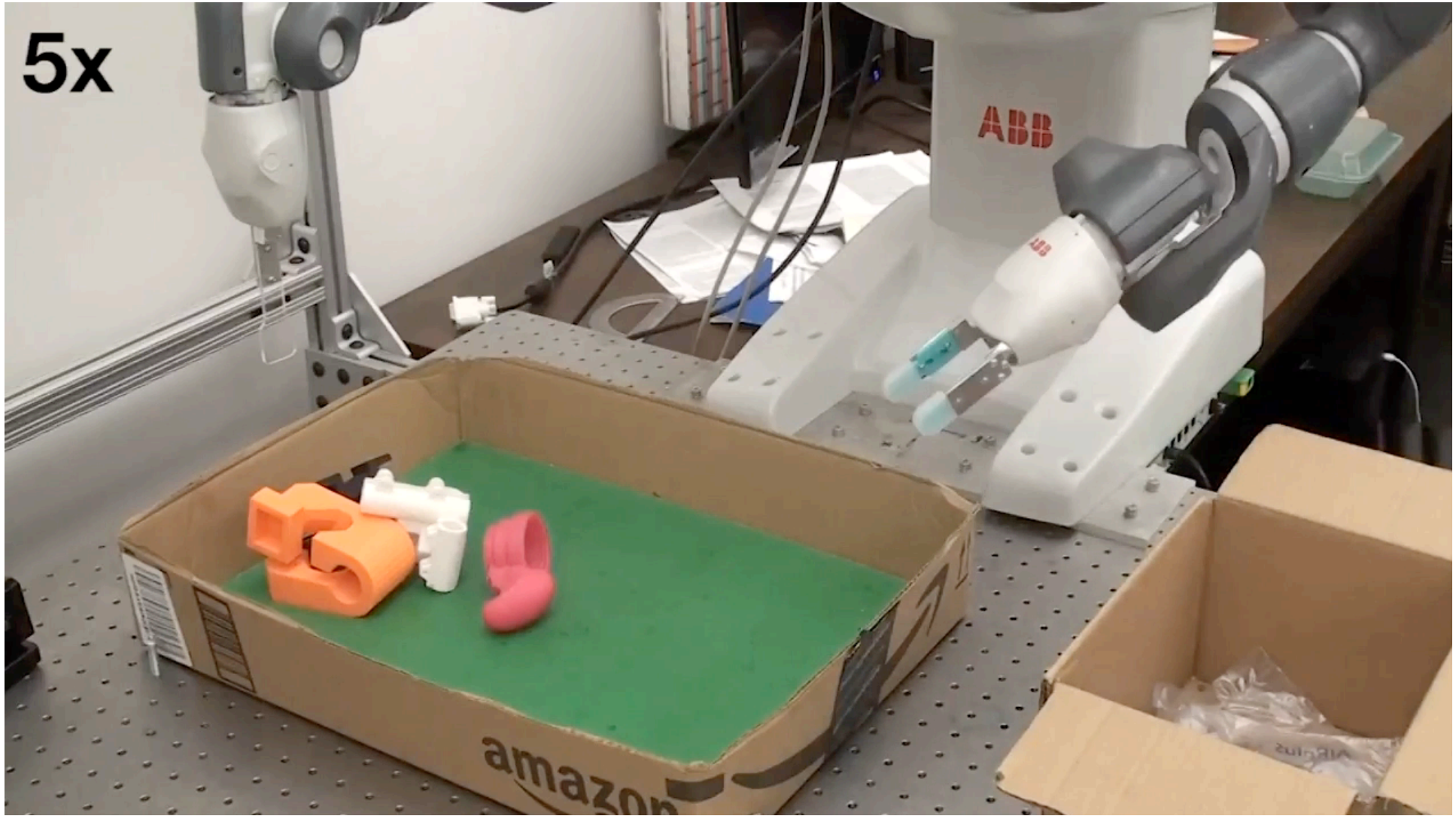


Run Time Planning



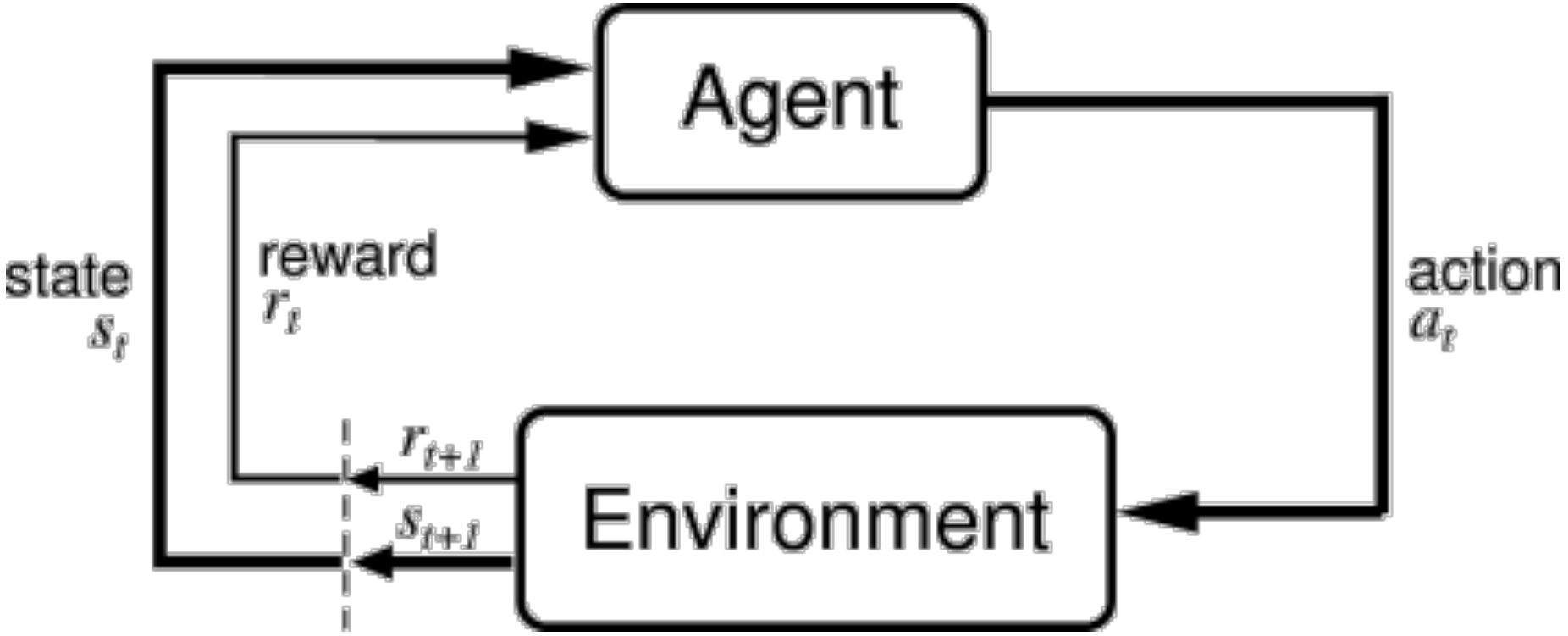
Mahler, J., Pokorny, F. T., Hou, B., Roderick, M., Laskey, M., Aubry, M., ... & Goldberg, K. Dex-Net 1.0: A Cloud-Based Network of 3D Objects for Robust Grasp Planning Using a Multi-Armed Bandit Model with Correlated Rewards.

State of the Art

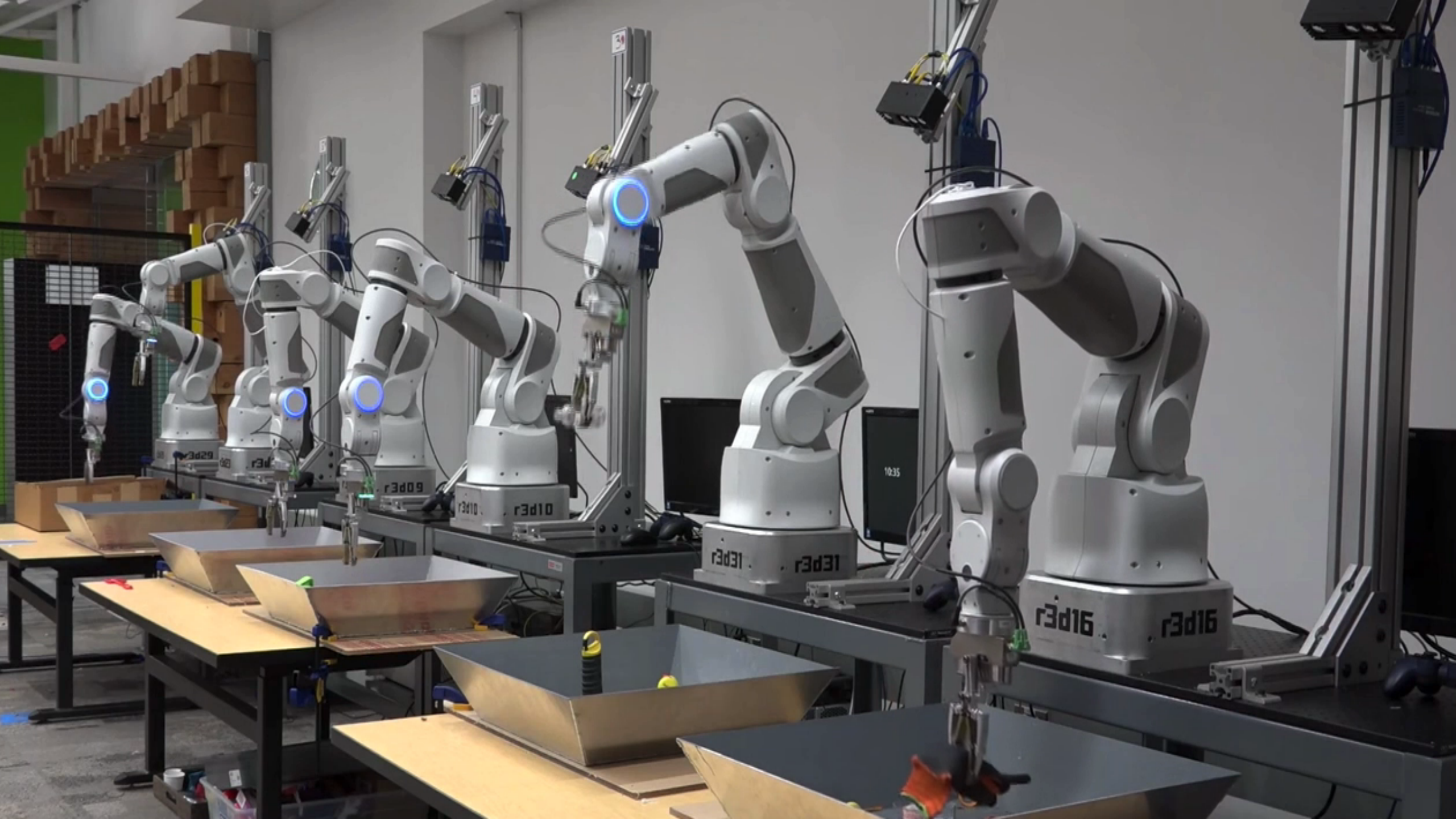


Limitations

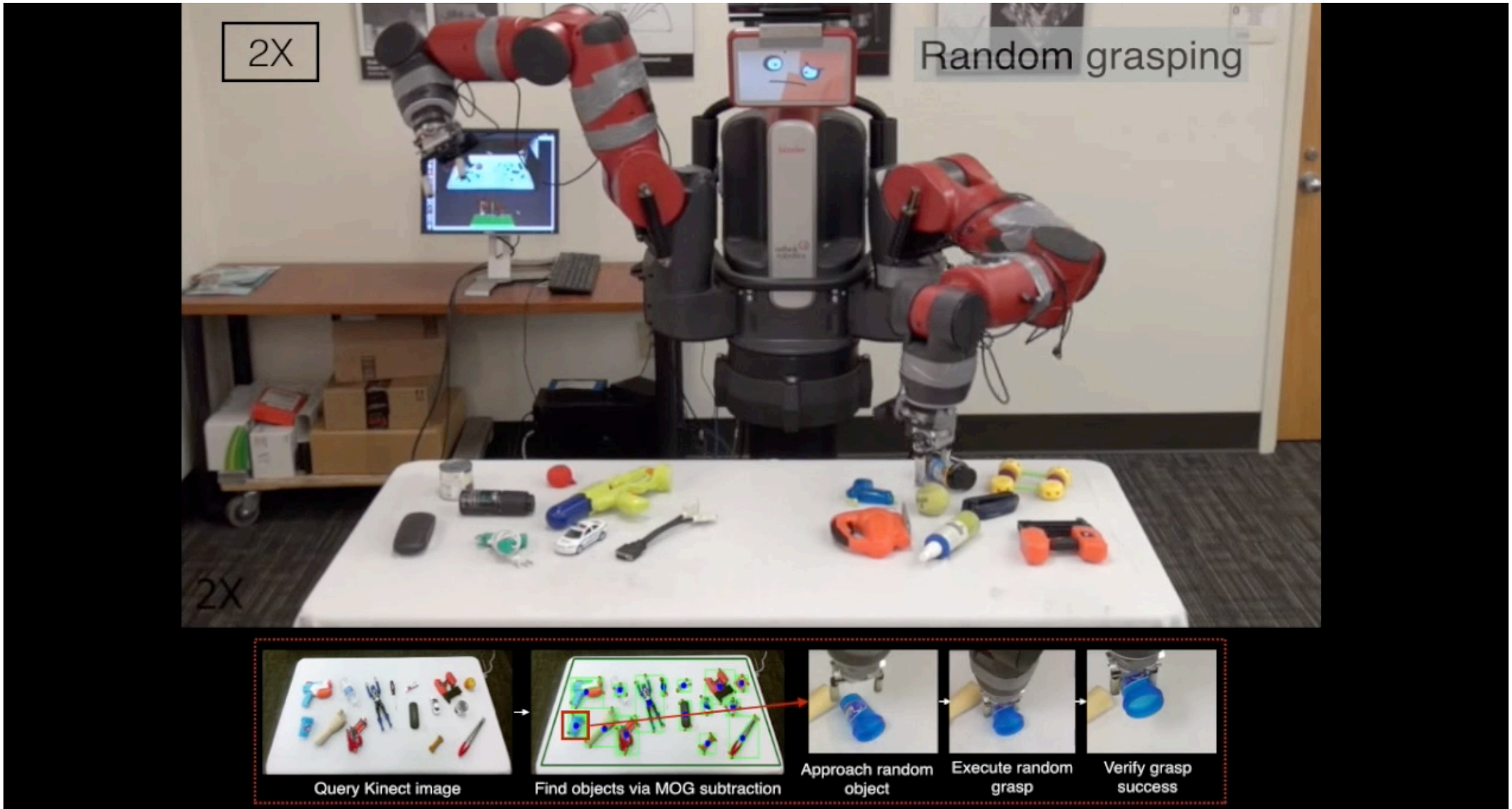
Idea:
Learn Through Interaction




KBA
Google DeepMind
Challenge Match
8 - 15 March 2016
AlphaGo
AlphaGo
Lee Sedol
Google DeepMind
Challenge Match
8 - 15 March 2016


r3d09
r3d10
r3d31
r3d16
10:35

Unsupervised Grasping (Policy Learning)



Unsupervised Grasping (i.e. 1-Step Q-Learning)

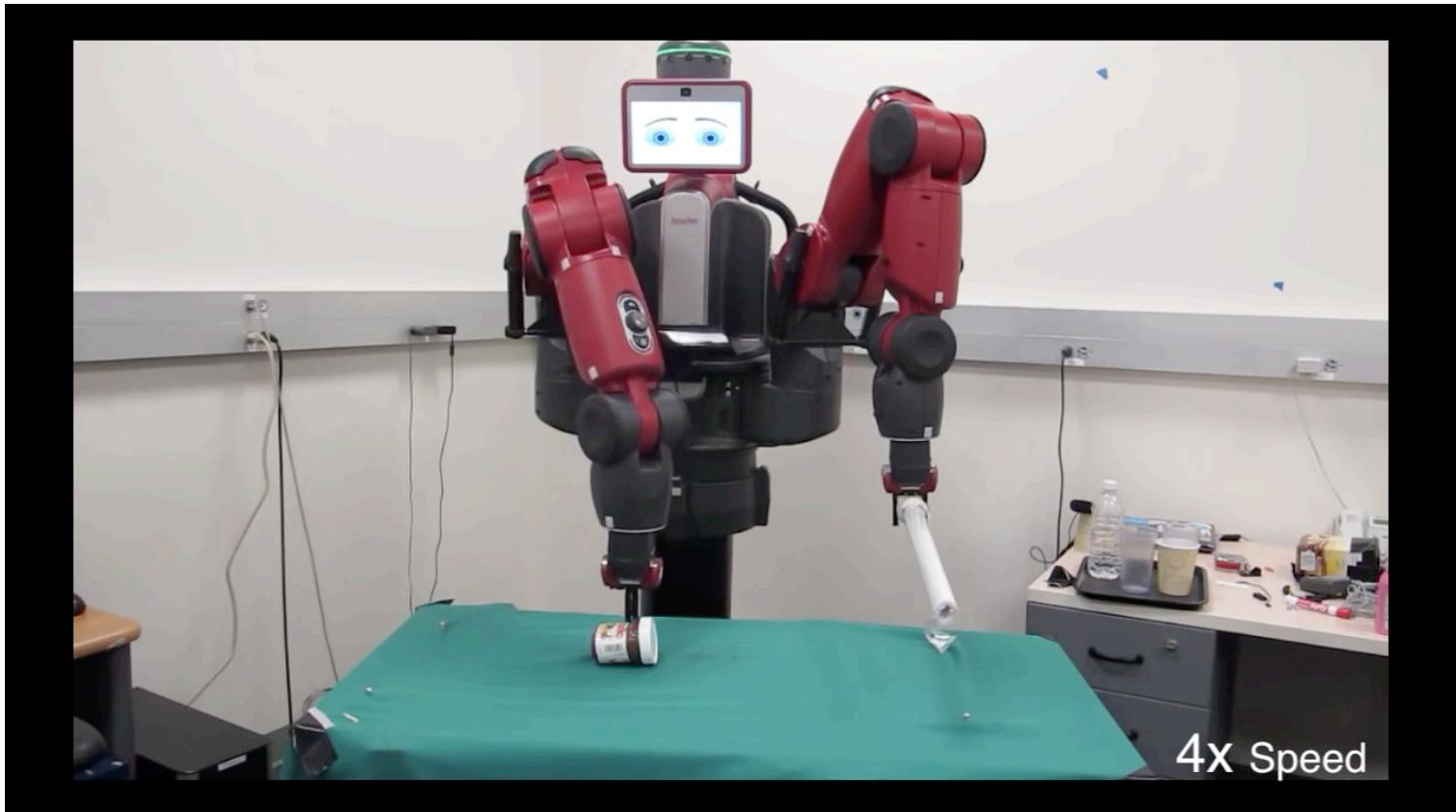
```
data = []
model = random_network
while True:
        grasp = try_grasp(model)
        success = check_grasp_success(grasp)
        if success :
                data.append([1,grasp])
        else:
                data.append([0,grasp])
        model.update(data)
```

Pinto, Lerrel, and Abhinav Gupta. "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours." *arXiv preprint arXiv:1509.06825*(2015).

Model-Based RL

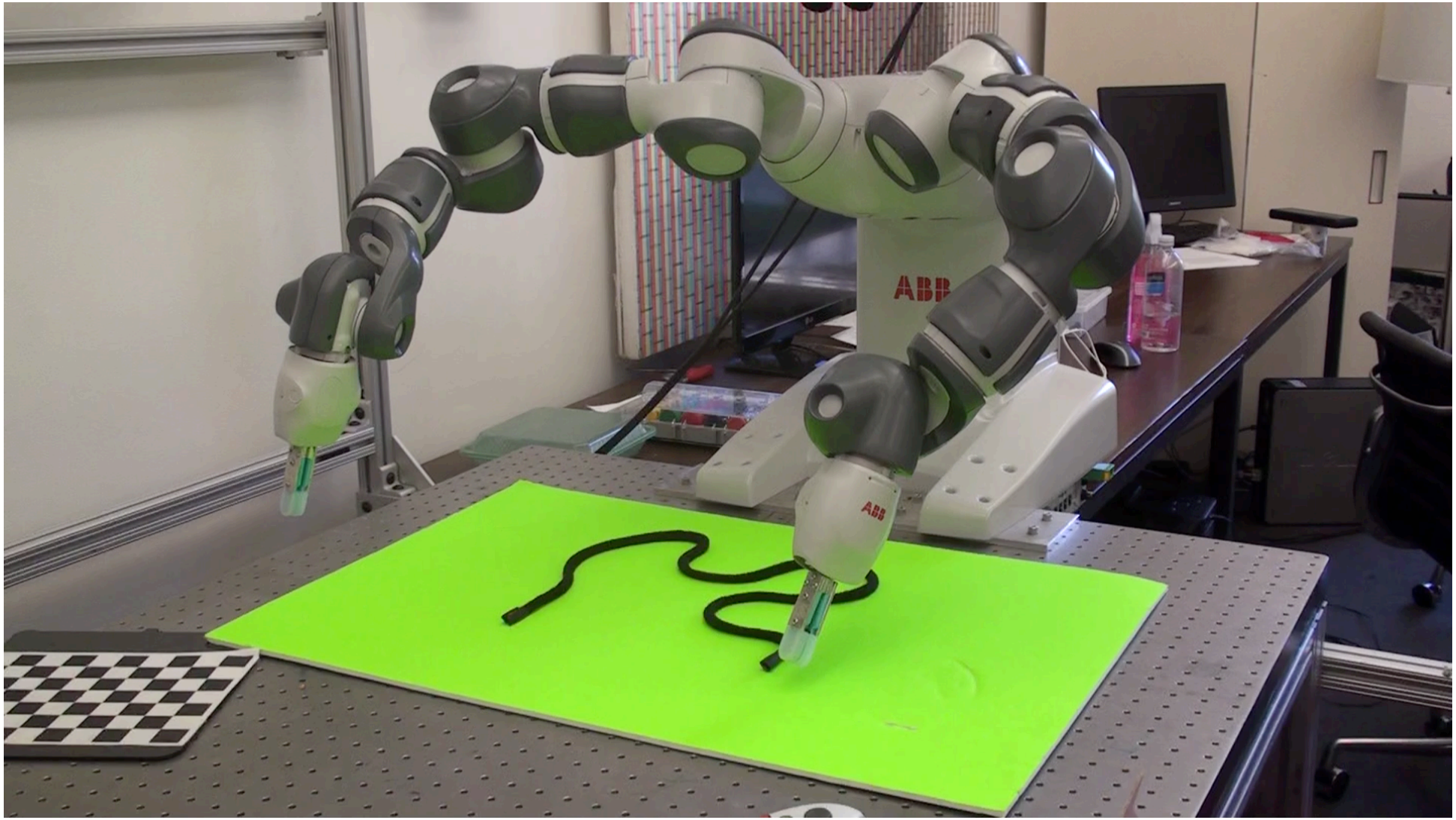
-Policies are Good for a Single Task

-Models can enable planning many tasks



Agrawal, P., Nair, A., Abbeel, P., Malik, J., & Levine, S. (2016). Learning to poke by poking: Experiential learning of intuitive physics. arXiv preprint arXiv:1606.07419.

RL for Rope Tying



RL for Rope Tying

State Space – RGBD Images

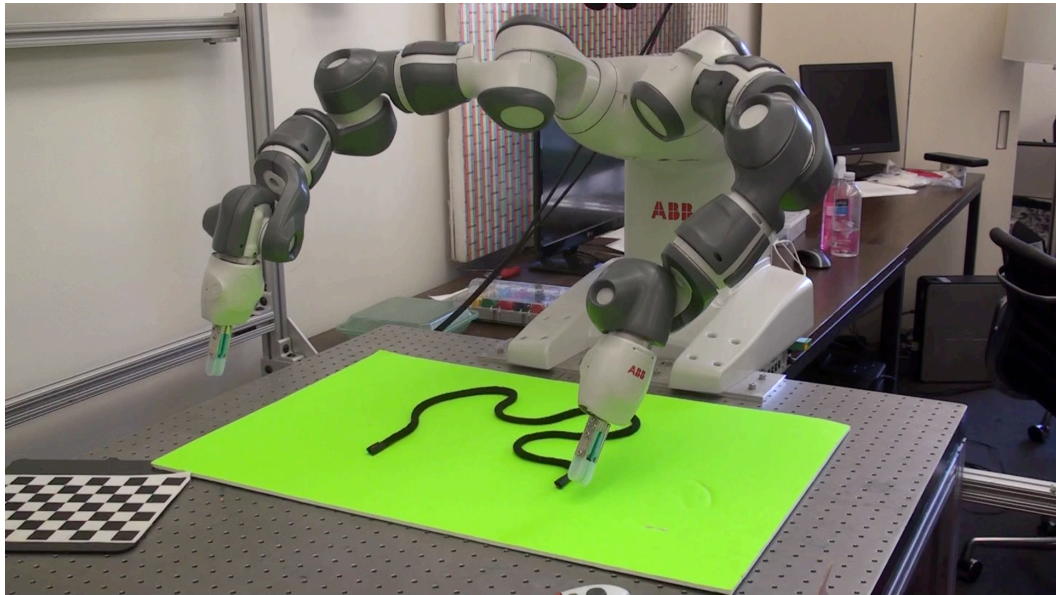
Action Space – Motor Torques

Reward – Binary Success of Rope Tied

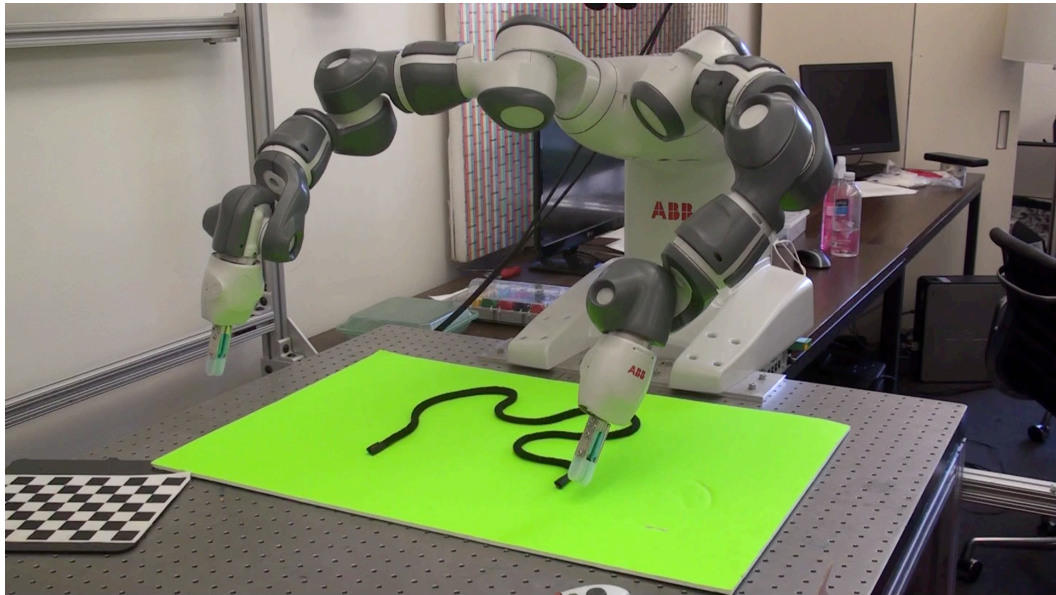
Time Horizon – Finite (200 Timesteps)

Q-Function – Deep Network

Apply Q-Learning?

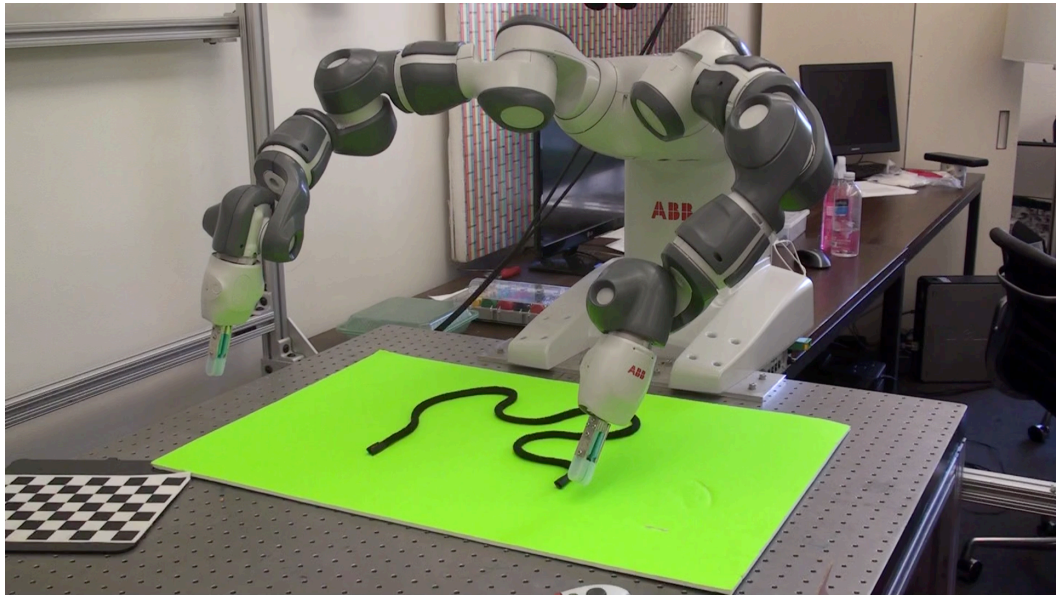


RL for Rope Tying (Limitations)



RL for Rope Tying (Limitations)

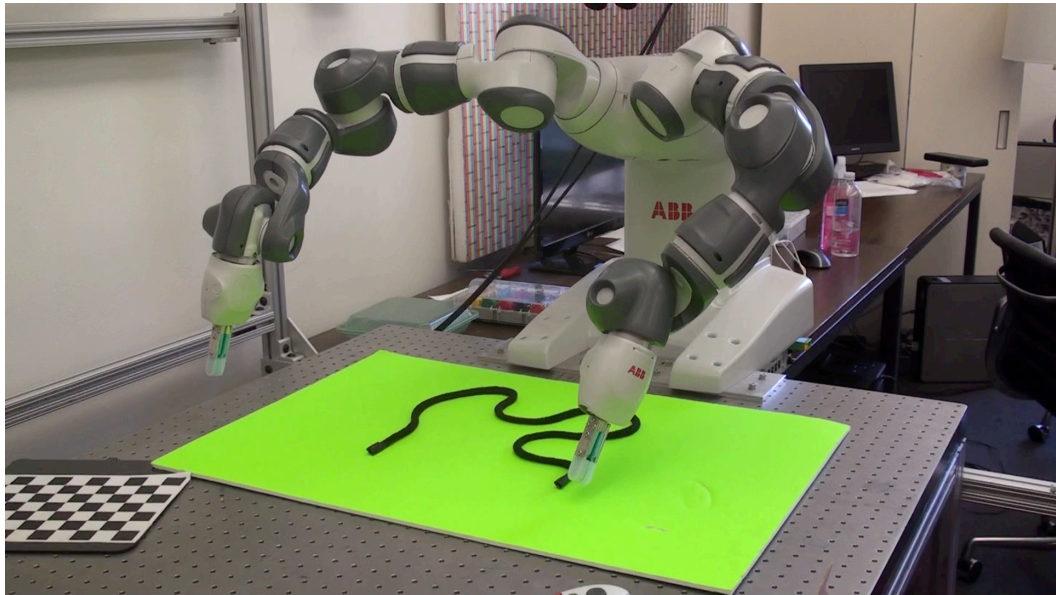
1) Very Delayed Reward



RL for Rope Tying (Limitations)

1) Very Delayed Reward

2) Need to detect if a Rope is Tied

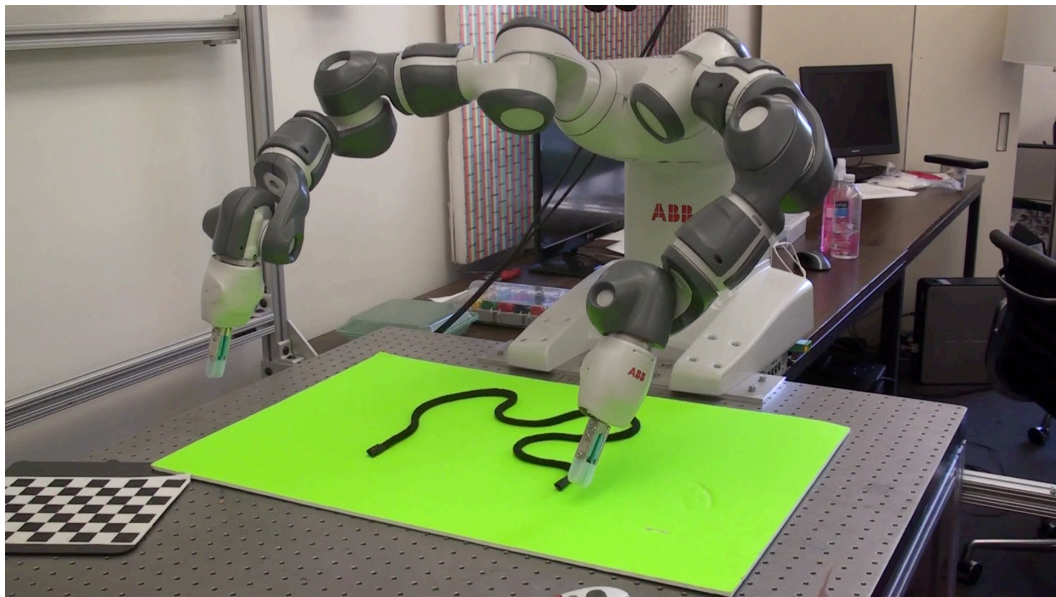


RL for Rope Tying (Limitations)

1) Very Delayed Reward

2) Need to detect if a Rope is Tied

3) Need to untied if Succesful



Try to Recover a Policy from a Supervisor

Learning From Demonstration

State and Controls: $\mathcal{X} \subset \mathbb{R}^{d_x} \quad \mathcal{U} \subset \mathbb{R}^{d_u}$

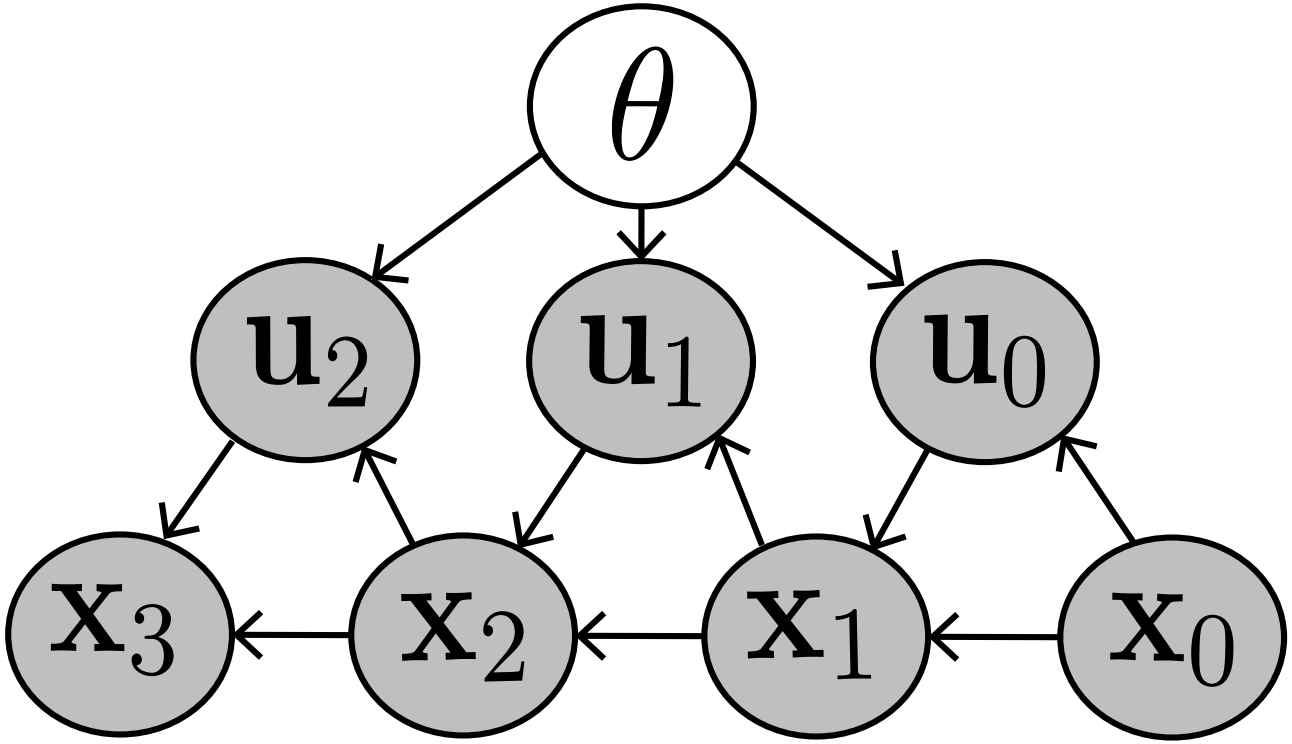
Robot's Policy: $\pi_\theta : \mathcal{X} \to \mathcal{U}$

Supervisor Policy: $\tilde{\pi} : \mathcal{X} \to \mathcal{U}$

State Distribution: $p(\mathbf{x}|\theta)$

Surrogate Loss: $l : \mathcal{U} \times \mathcal{U} \to [0, 1]$

Graphical Model Interpretation



Solving for Robot's Policy

$$p(\tau|\theta) = p(x_0)\prod_{t=1}^{3} p(x_t|x_{t-1}, u_{t-1})p(u_{t-1}|x_{t-1}, \theta)$$

Idea maximize likilihood of $p(\tau|\theta)$ given N demonstrations

$$\max_{\theta} \prod_{i=0}^{N-1} p(\tau_i|\theta)$$

$$\max_{\theta} \prod_{i=0}^{N-1} p(x_{0,i})\prod_{t=1}^{3} p(x_{t,i}|x_{t-1,i}, u_{t-1,i})p(u_{t-1,i}|x_{t-1,i}, \theta)$$

Solving for Robot's Policy

Take the Log operation to break up product

$$\max_{\theta} \sum_{i=0}^{N-1} \log p(x_{0,i}) + \sum_{i=0}^{N-1} \sum_{t=1}^{3} \log p(x_{t,i}|x_{t-1,i}, u_{t-1,i}) + \sum_{i=0}^{N-1} \sum_{t=1}^{3} \log p(u_{t-1,i}|x_{t-1,i}, \theta)$$

$$\max_{\theta} \sum_{i=0}^{N-1} \sum_{t=1}^{3} \log p(u_{t-1,i}|x_{t-1,i}, \theta)$$

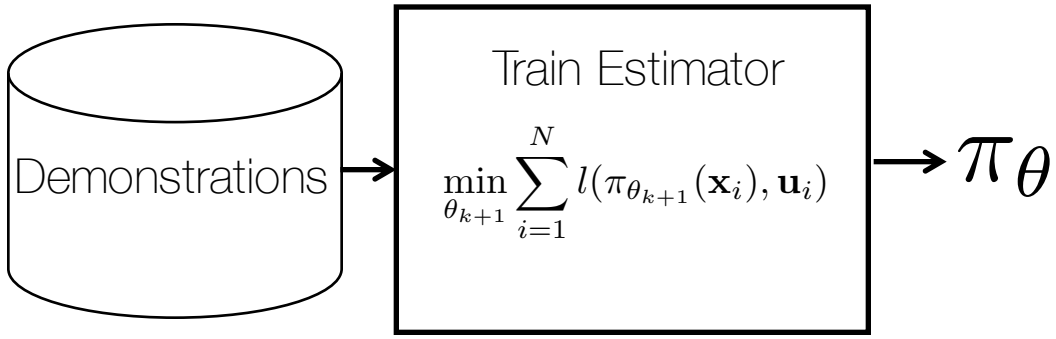
For a Gaussian Distribution $\mathcal{N}(\pi_\theta(x), I)$

$$\max_{\theta} \sum_{i=0}^{N-1} \sum_{t=1}^{3} ||\pi_\theta - u_{t-1,i}||_2^2$$

Learning from Demonstration

1) Collect Demonstrations

1) Compute a Policy that Minimizes Expected Loss

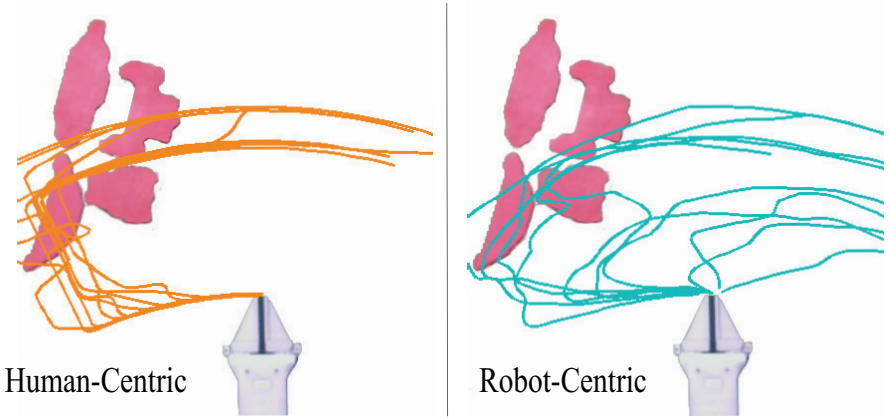


Pomerleau, Dean A. Alvinn: An autonomous land vehicle in a neural network. No. AIP-77. CARNEGIE-MELLON UNIV PITTSBURGH PA ARTIFICIAL INTELLIGENCE AND PSYCHOLOGY PROJECT, 1989.

Error Rates for Learning a Policy

Theorem 5.2: Given a policy π_{θ^N}, the following inequalities holds

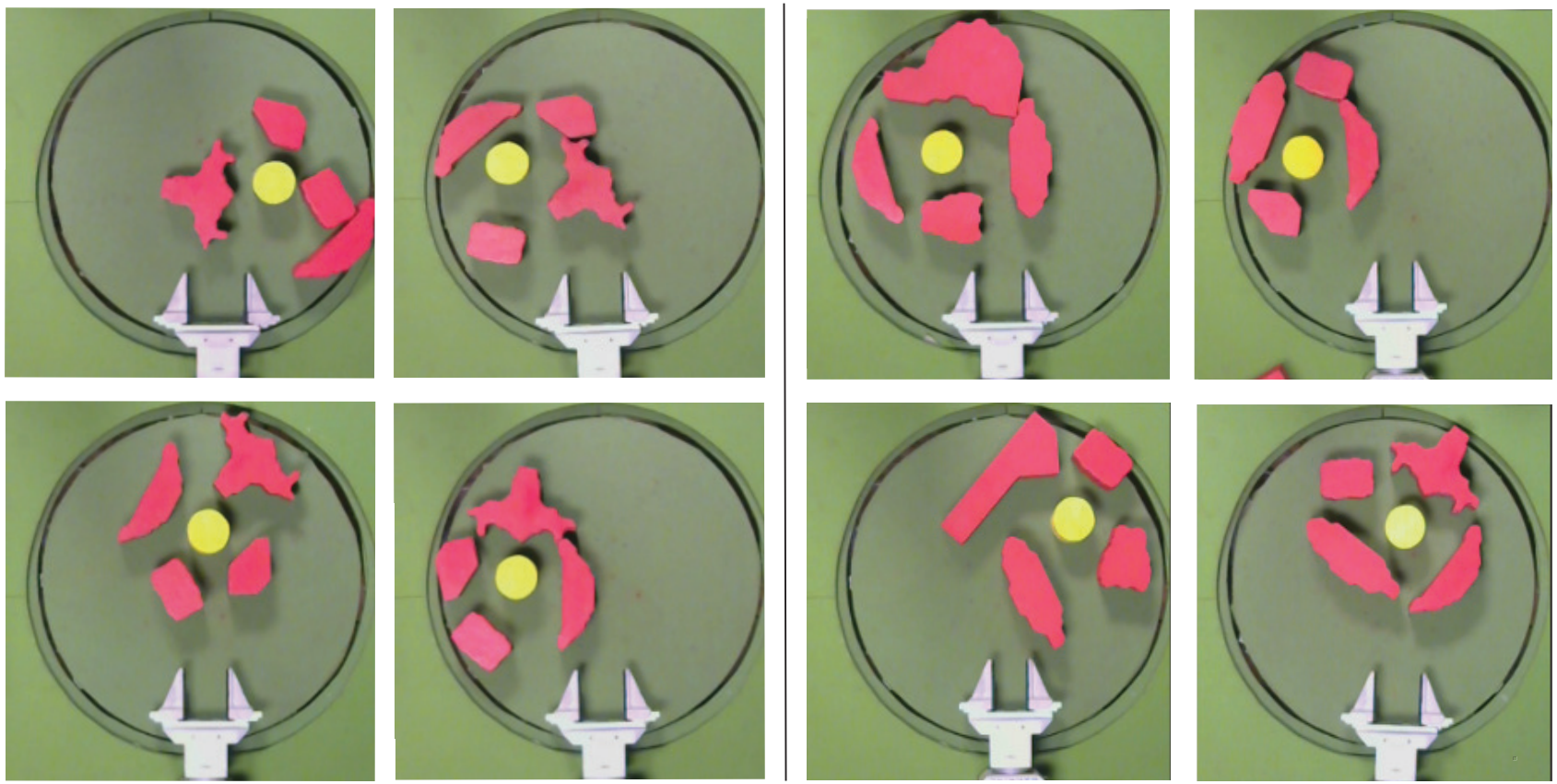
$$E_{p(\tau|\theta^n)}J(\theta^N) \leq T\sqrt{\frac{1}{4\sigma}E_{p(\tau|\theta^*)}J(\theta^N)} + E_{p(\tau|\theta^*)}J(\theta^N)$$



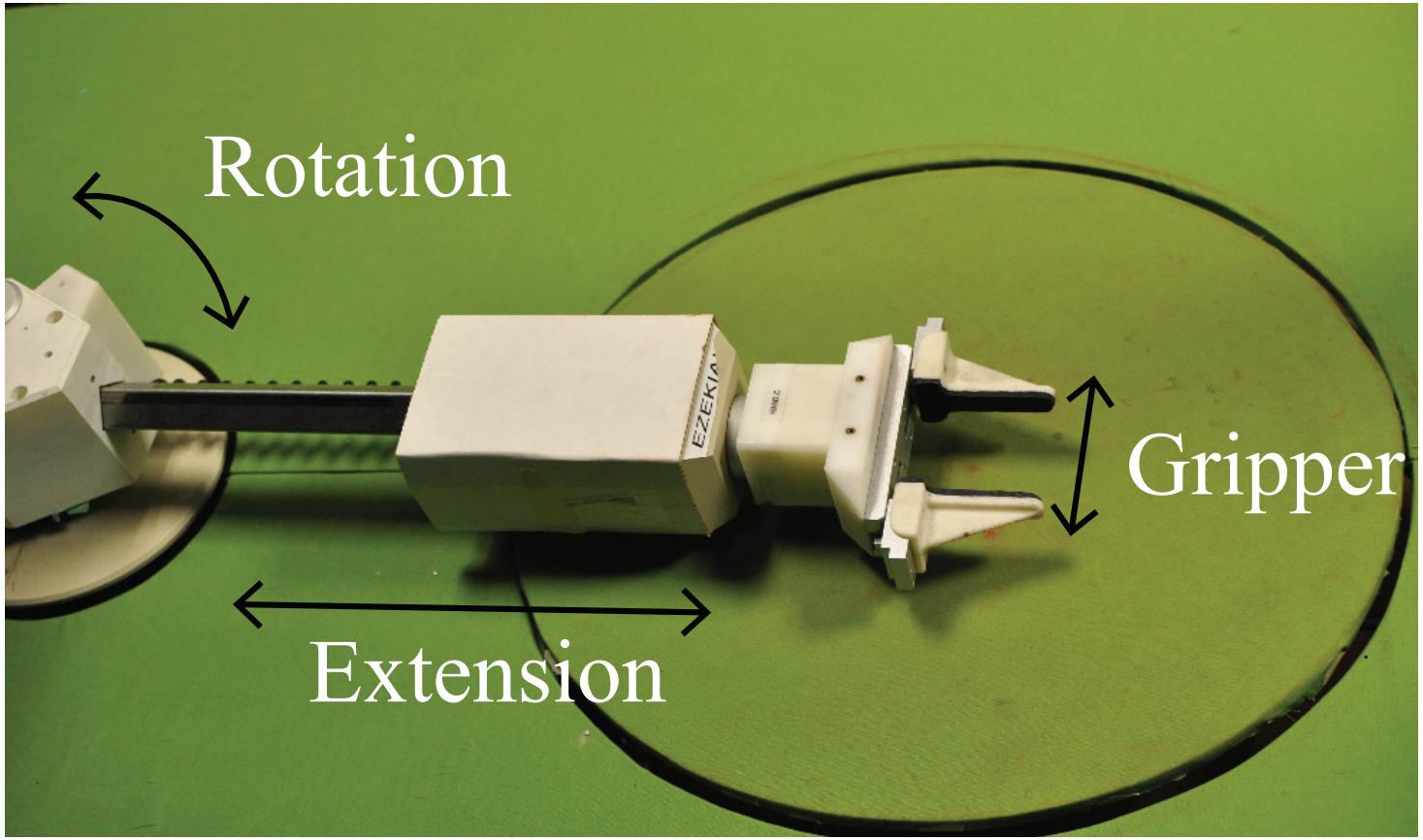
"Comparing Robot-Centric versus Human Centric for Deep Robot Learning from Demonstrations" M. Laskey, C. Chuck, J. Lee, S. Krishnan, J. Mahler, K. Jamieson, A. D. Dragan, and K. Goldberg

Break

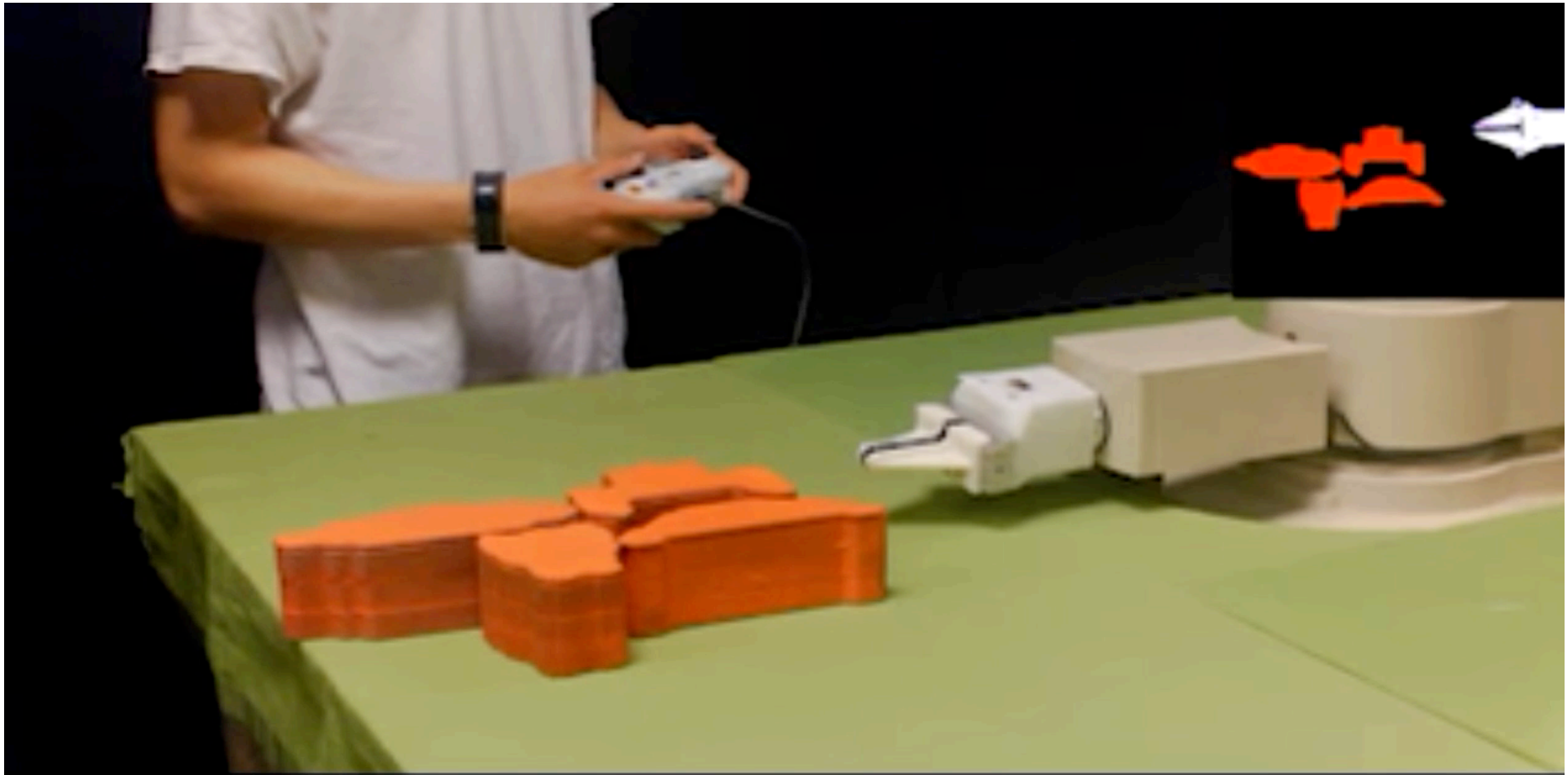
Case Study: Grasping in Clutter



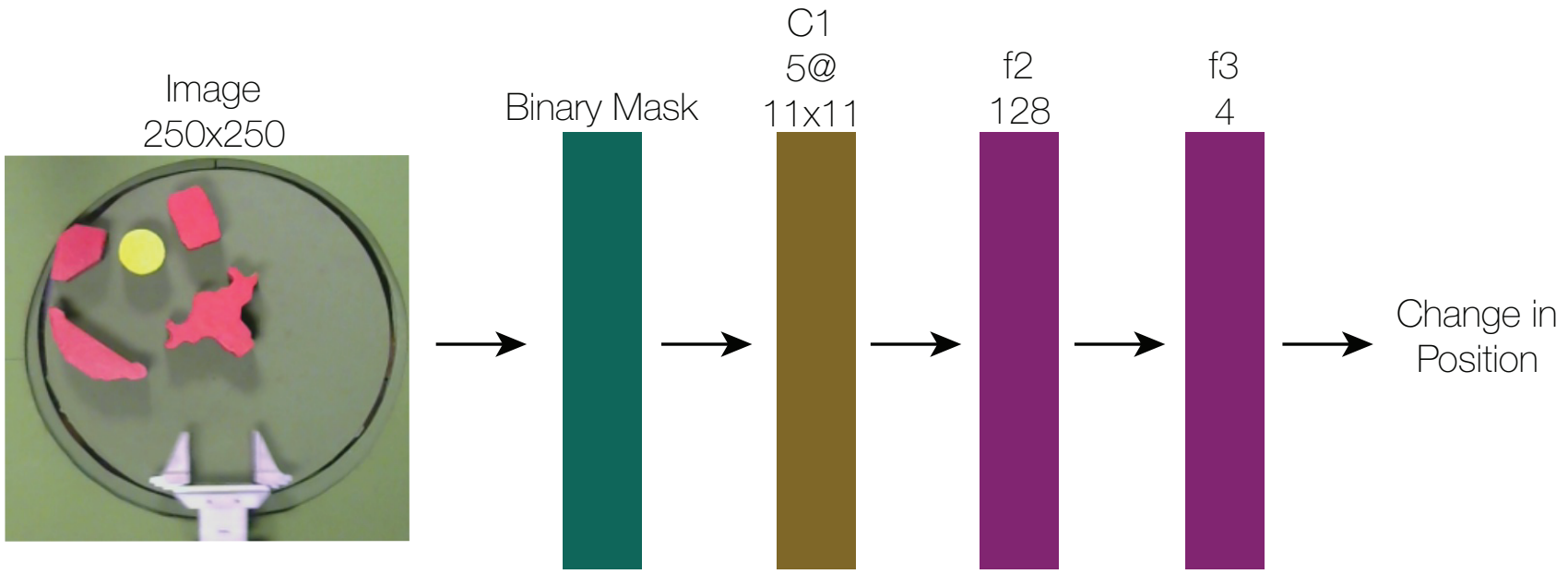
Zymark Robot



Demonstrations Through Tele-Operation



Deep Controller

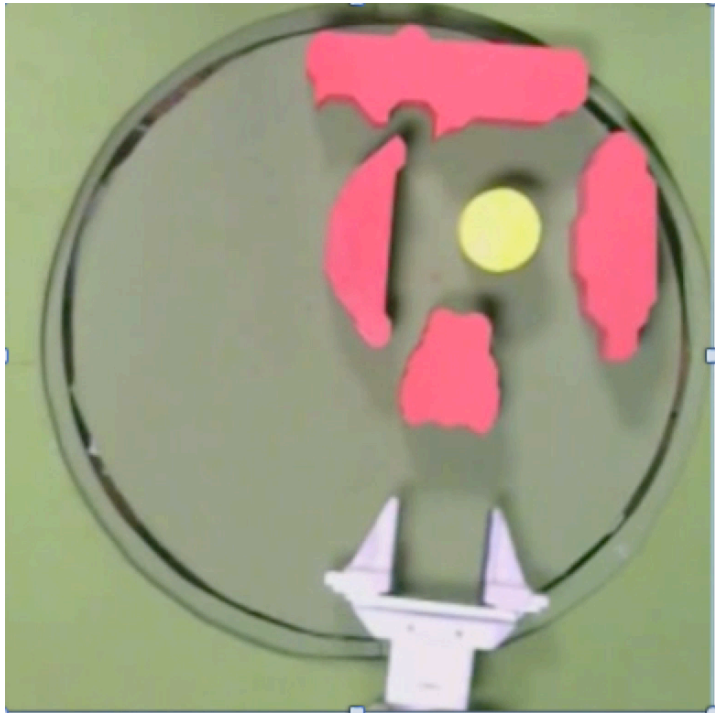


Network Training Details

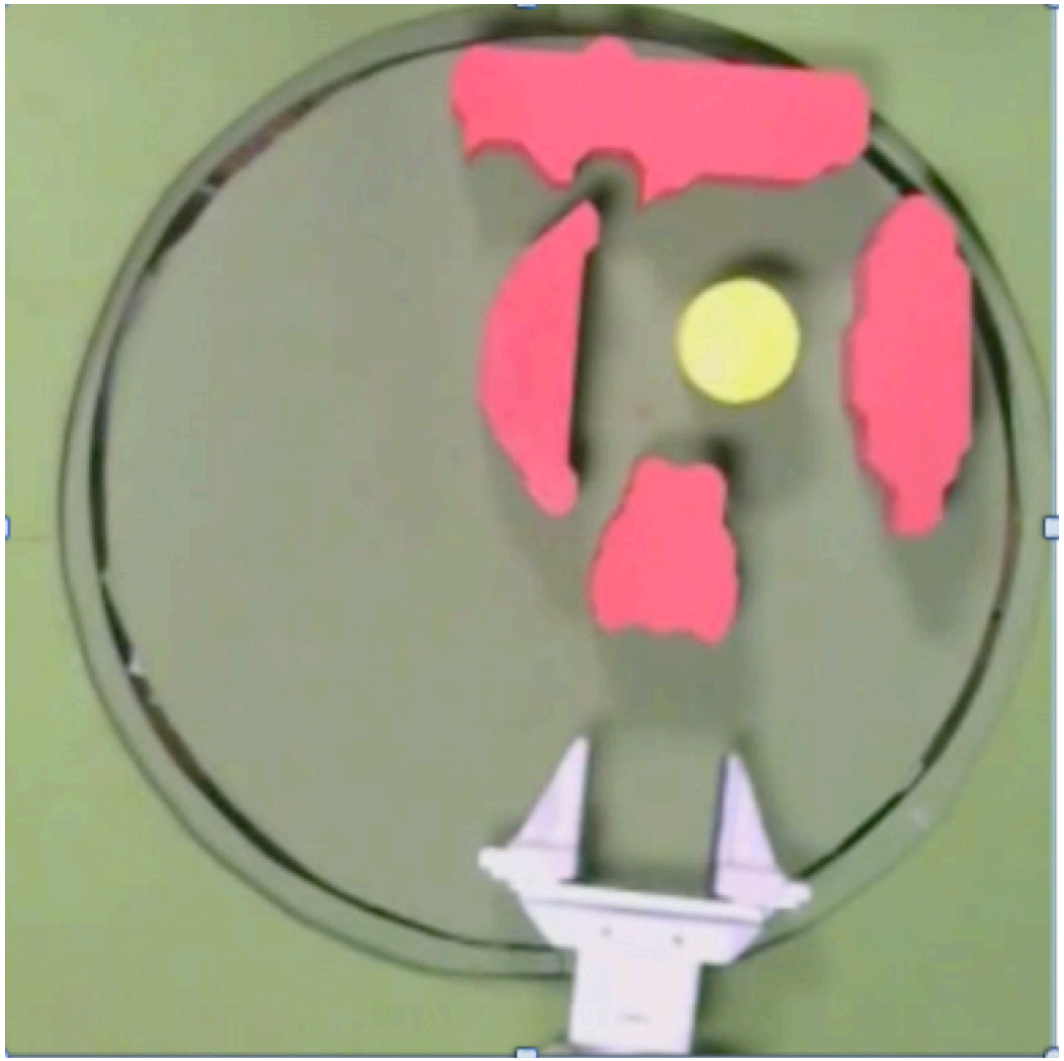
- Trained in TensorFlow on a NVIDIA Tesla K40
- Trained with Stochastic Gradient Descent w/ Momentum
- Batch= 200, Momentum = 0.01, Learning Rate = 0.9
- Parameters found via 40 binary perturbations and chose net with best test loss on 6K dataset of MPIO demonstrations

Why Deep Learning in Robotics?

1. Image Data is a Common State Space Choice
2. Not very clear what a lower-dimensional would be
3. Very Fast Evaluation Time
4. Get funding ;-)

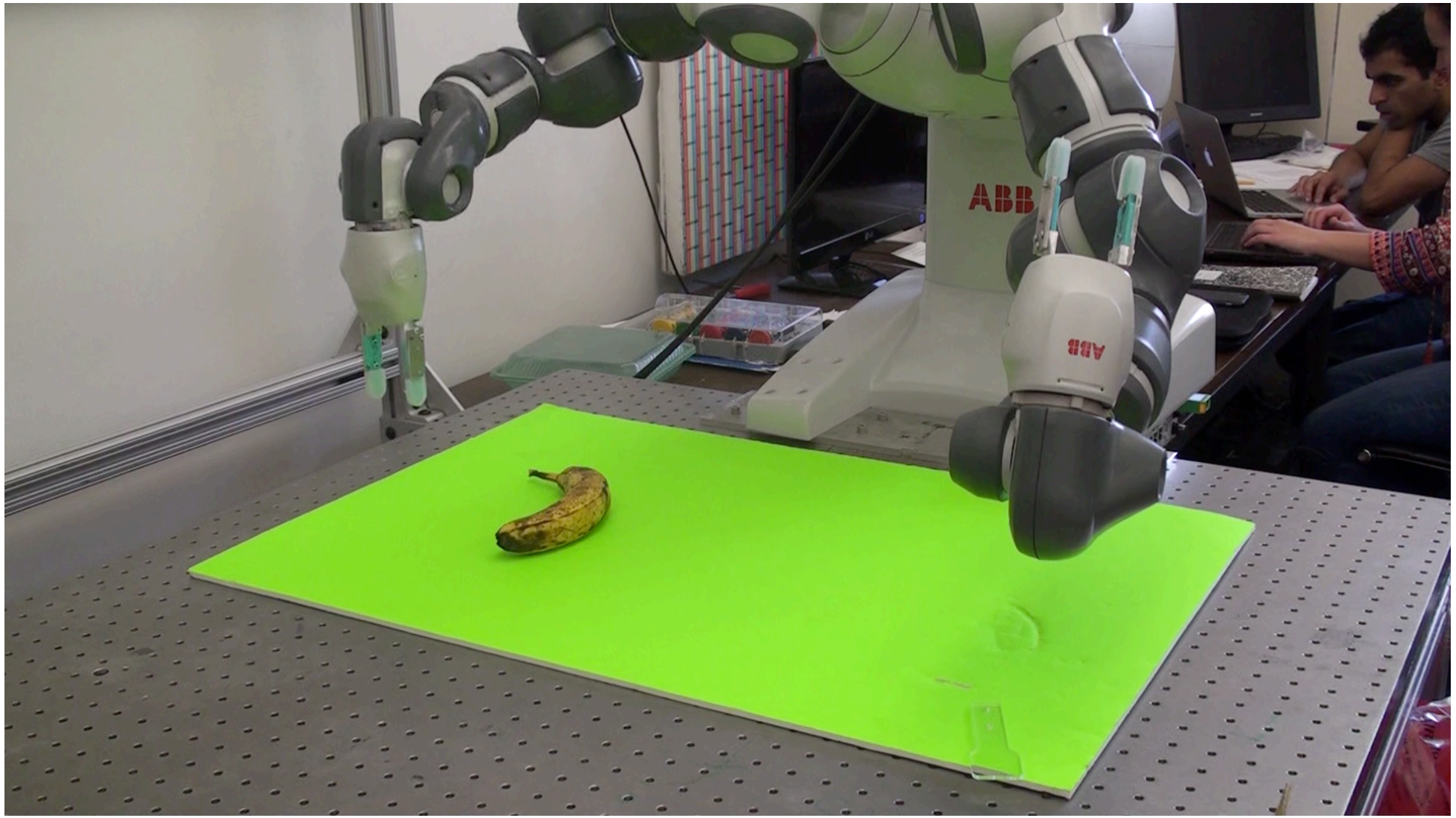


Learned Policy



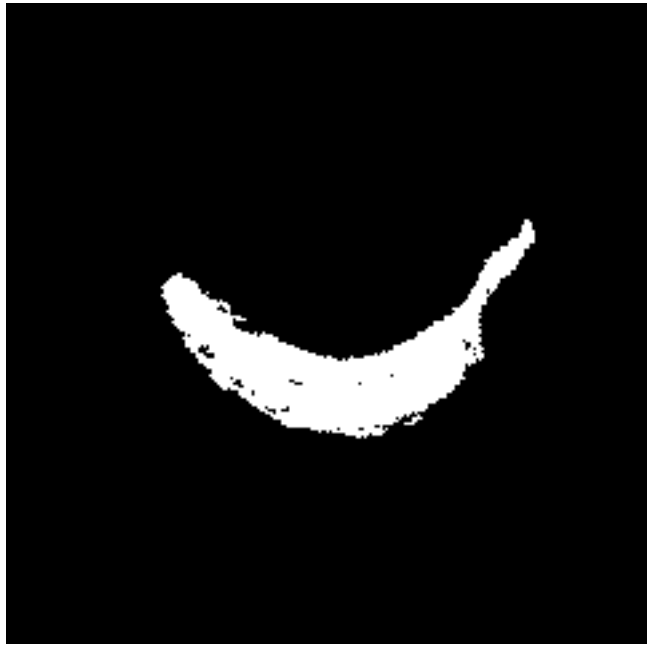
300 Demonstrations (4 hrs)
Trained on Configurations w/ 4 objects
Generalize to even 8 object configurations

Case Study: Banana Grasping



State Space Representation

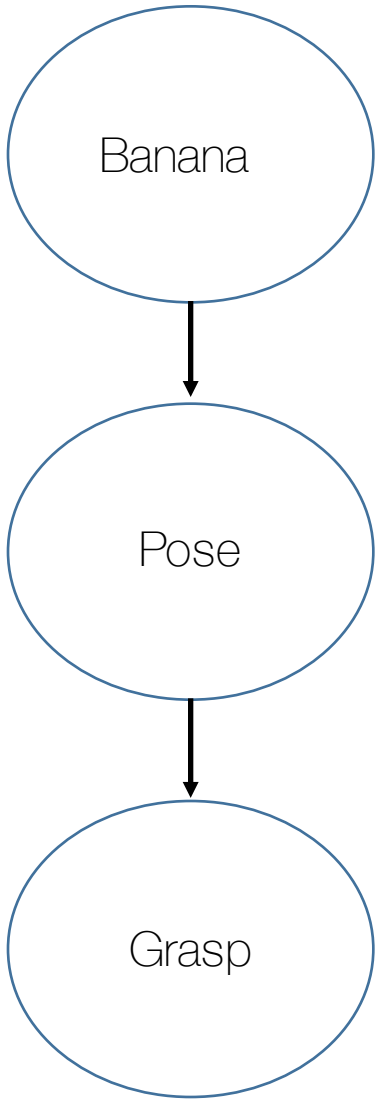
RGB Image (250x250x3 Pixels)
Captures Banana Shape and Texture



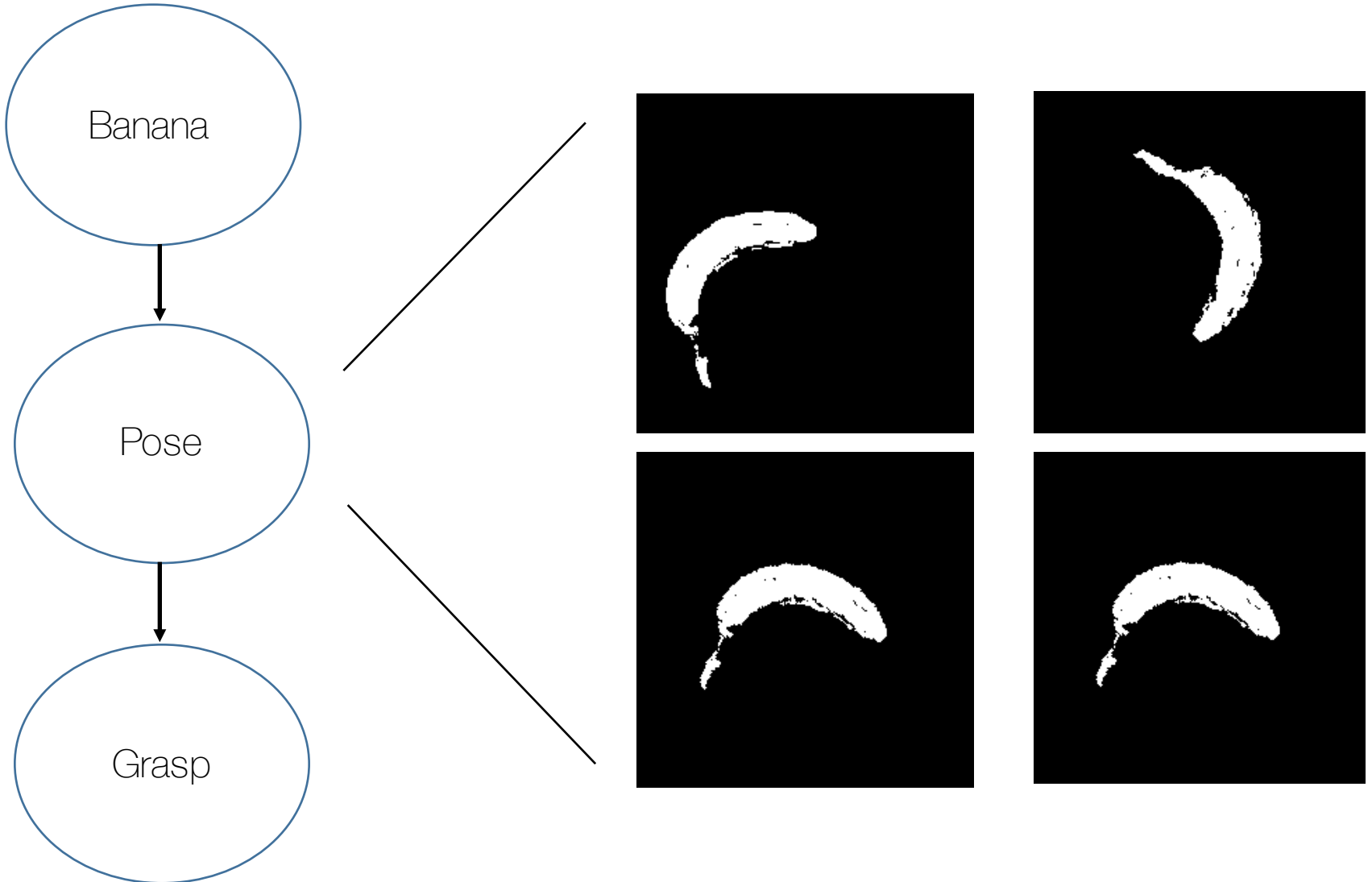
Binary Image (250x250 Pixels)
Captures Banana Shape
Invariant to Color

Binary Mask Robust to Decay

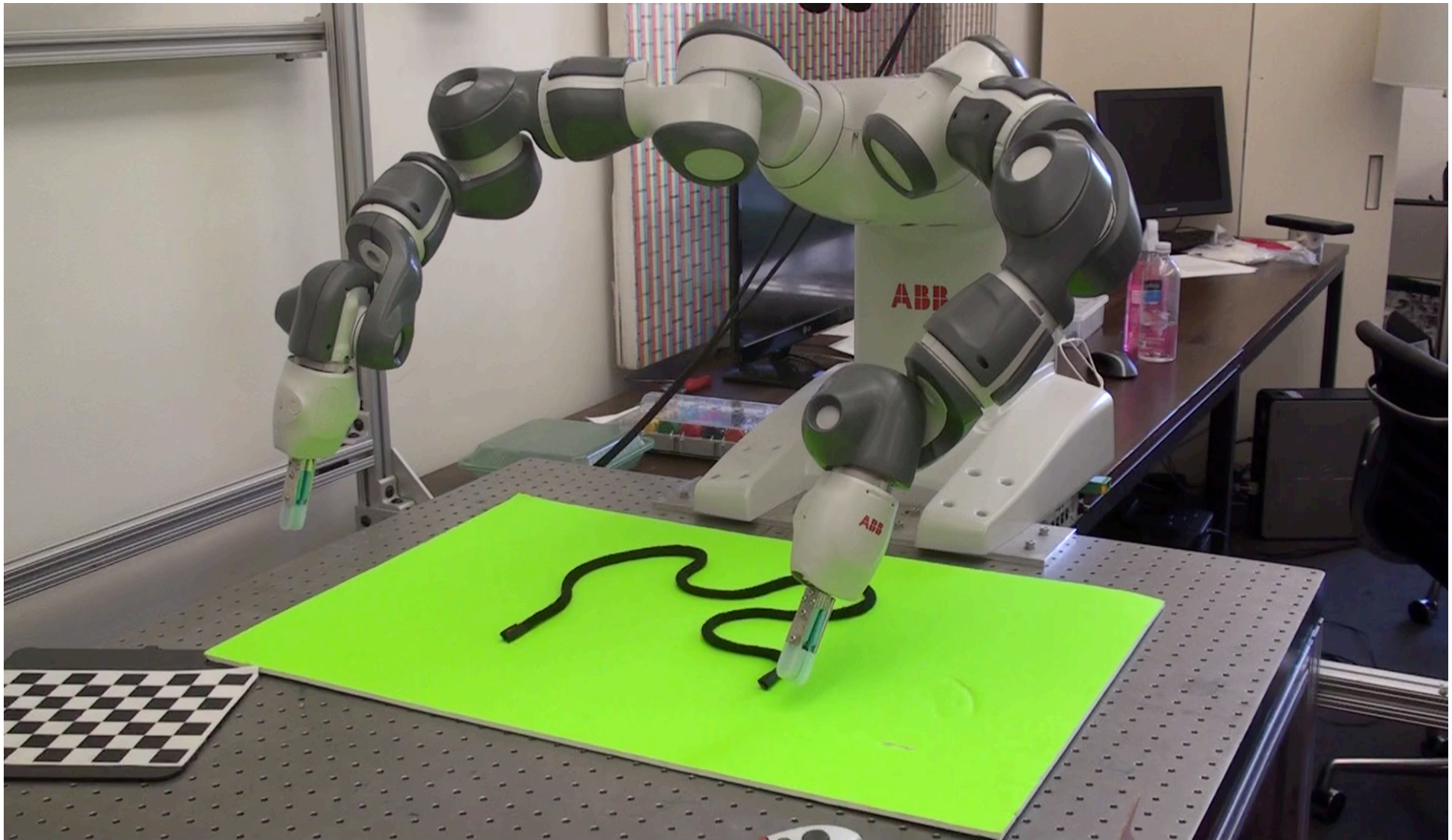
Characterizing Variance



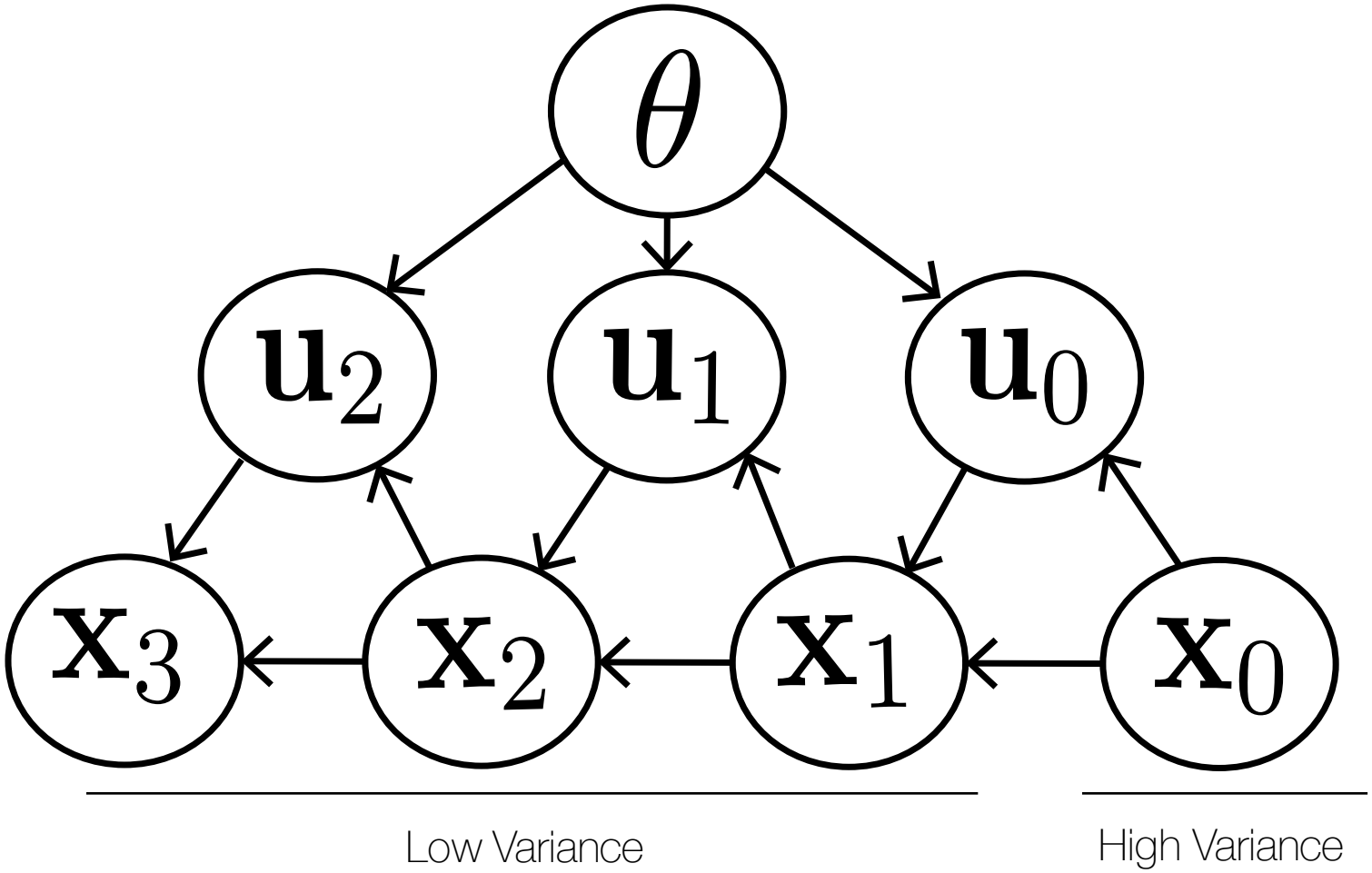
Data Augmentation:
We do have some Models!



Case Study: Rope Tying

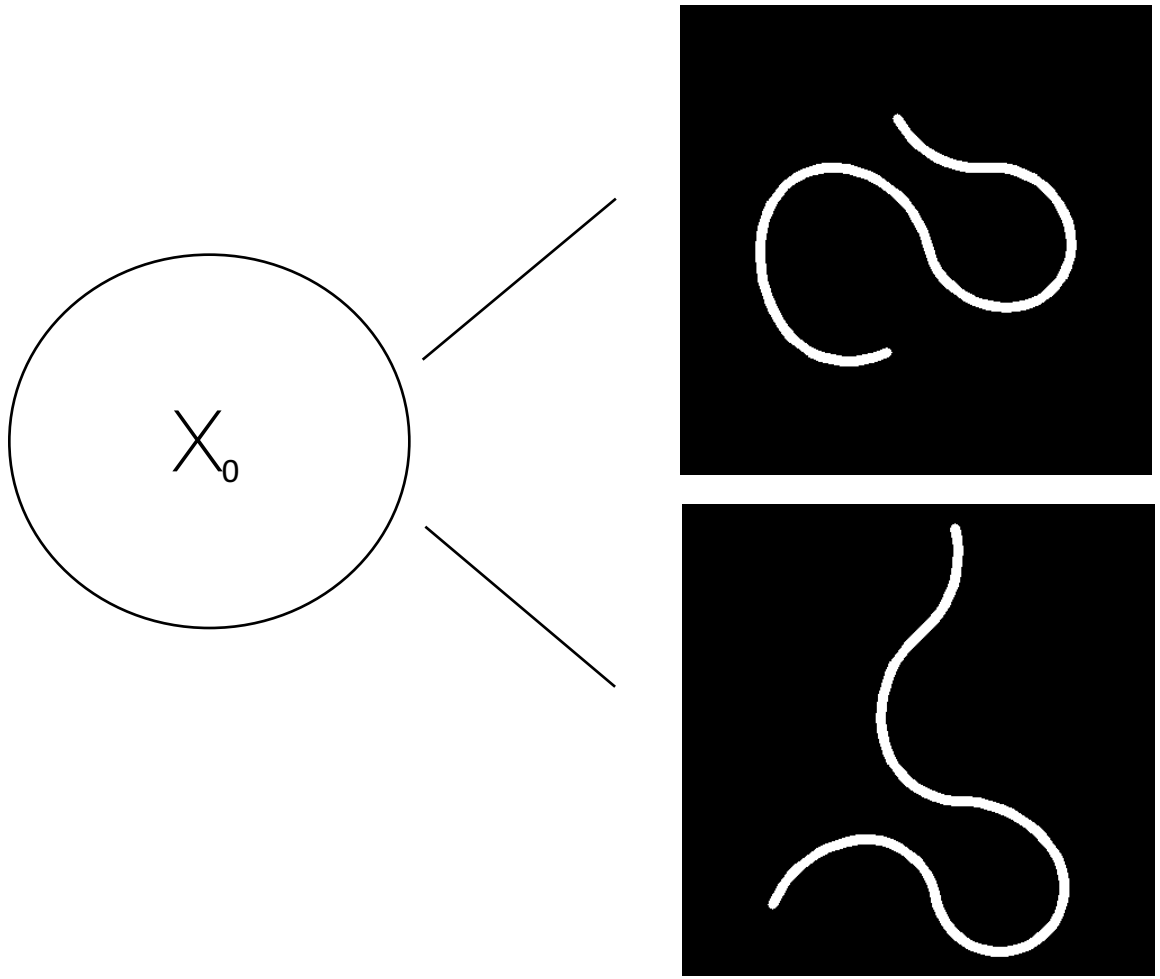


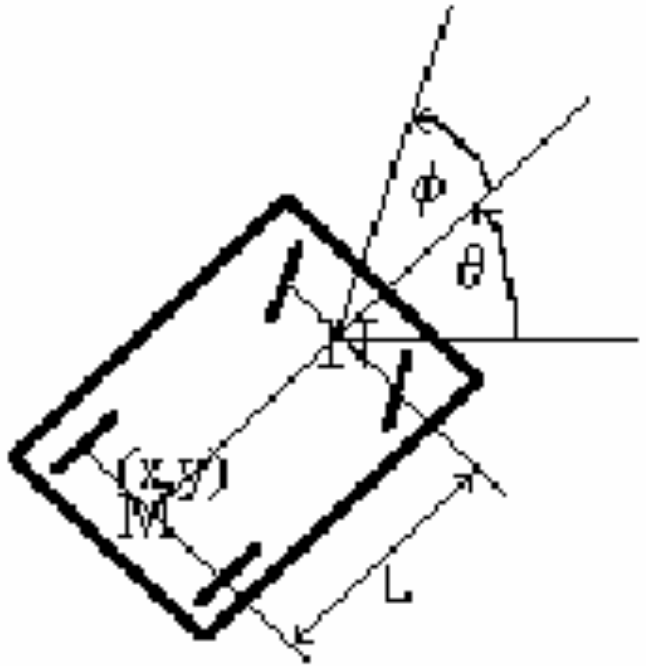
Characterizing Variance



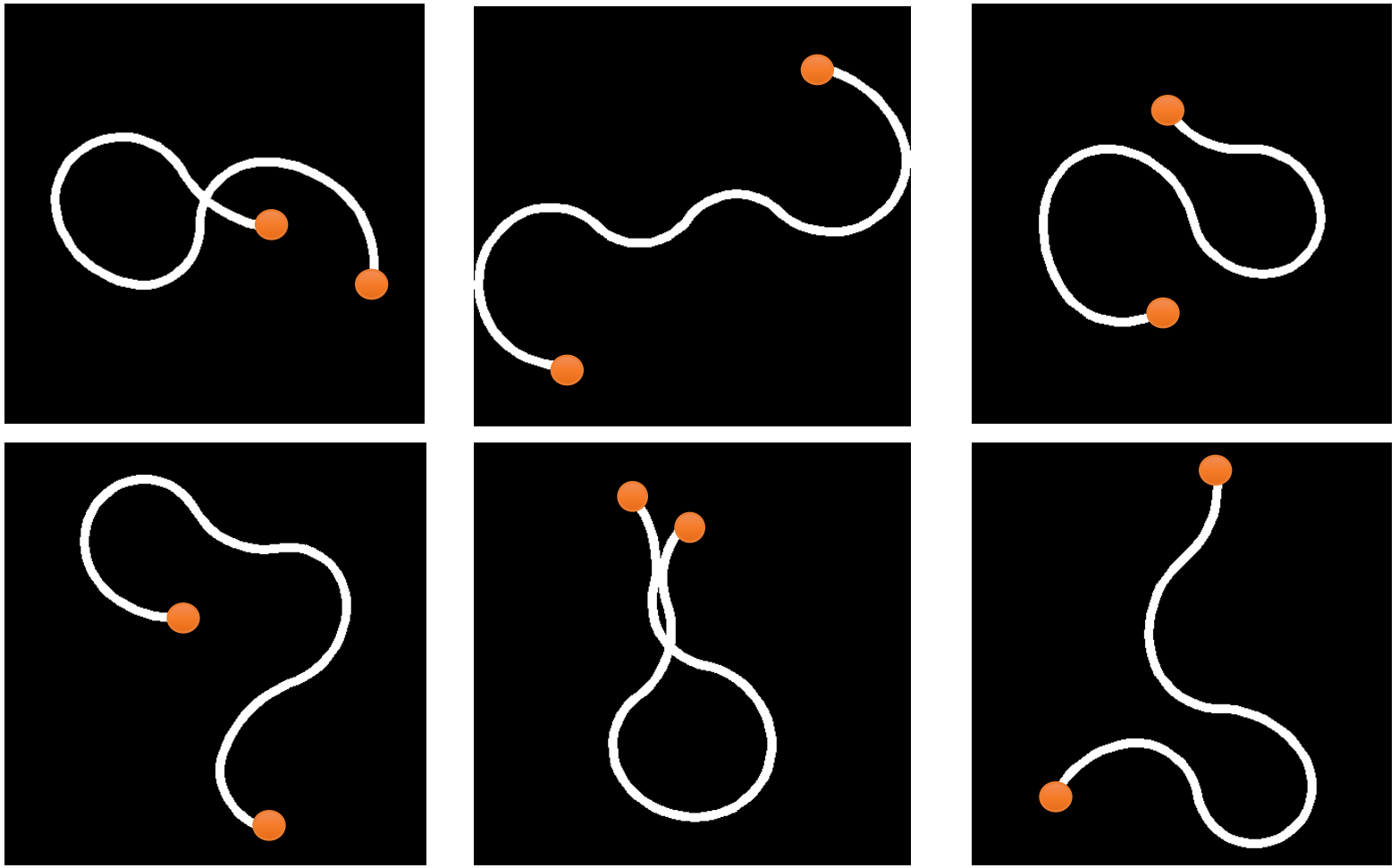
Industrial Robots

Synthetically Sample Initial States

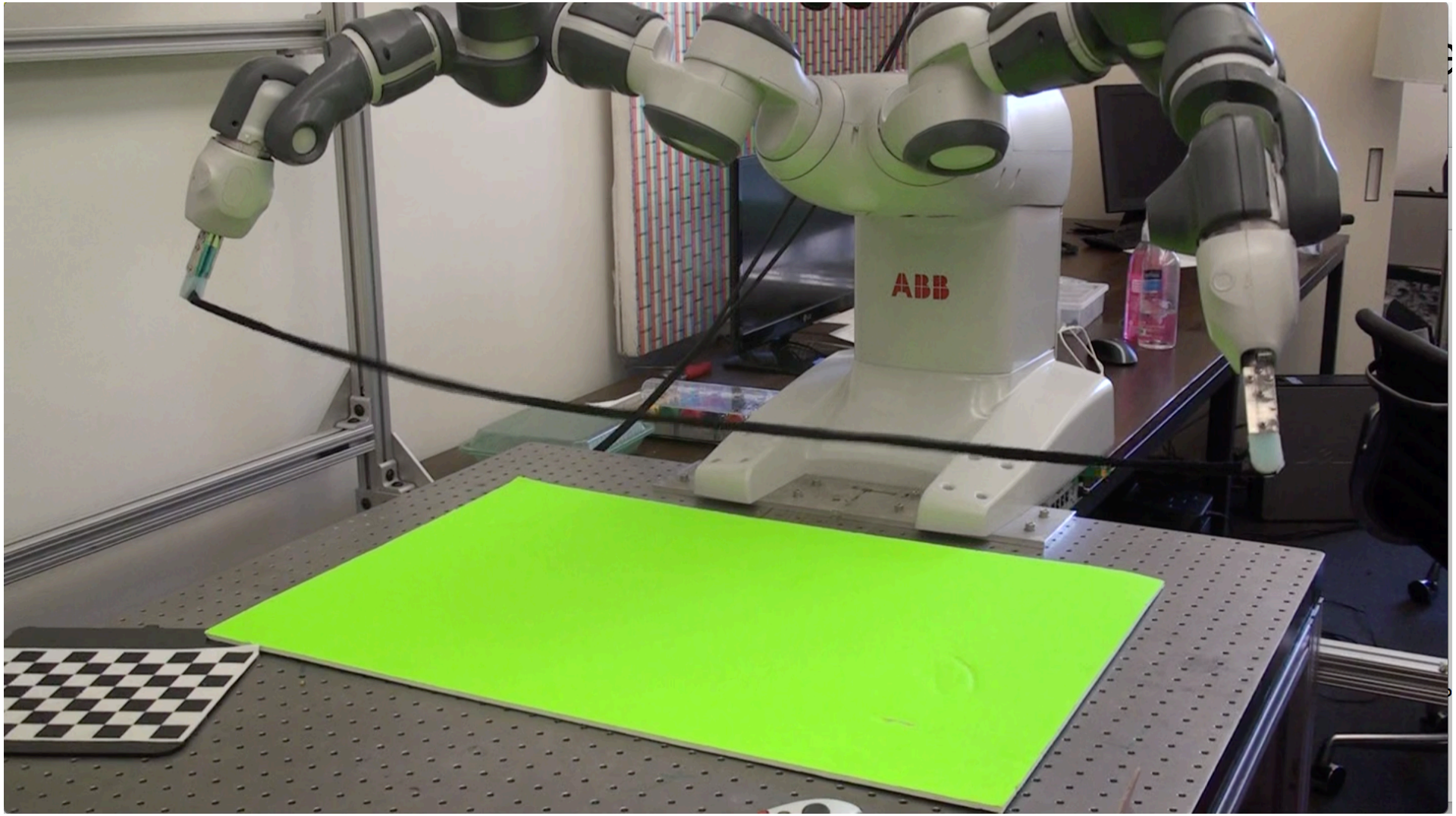




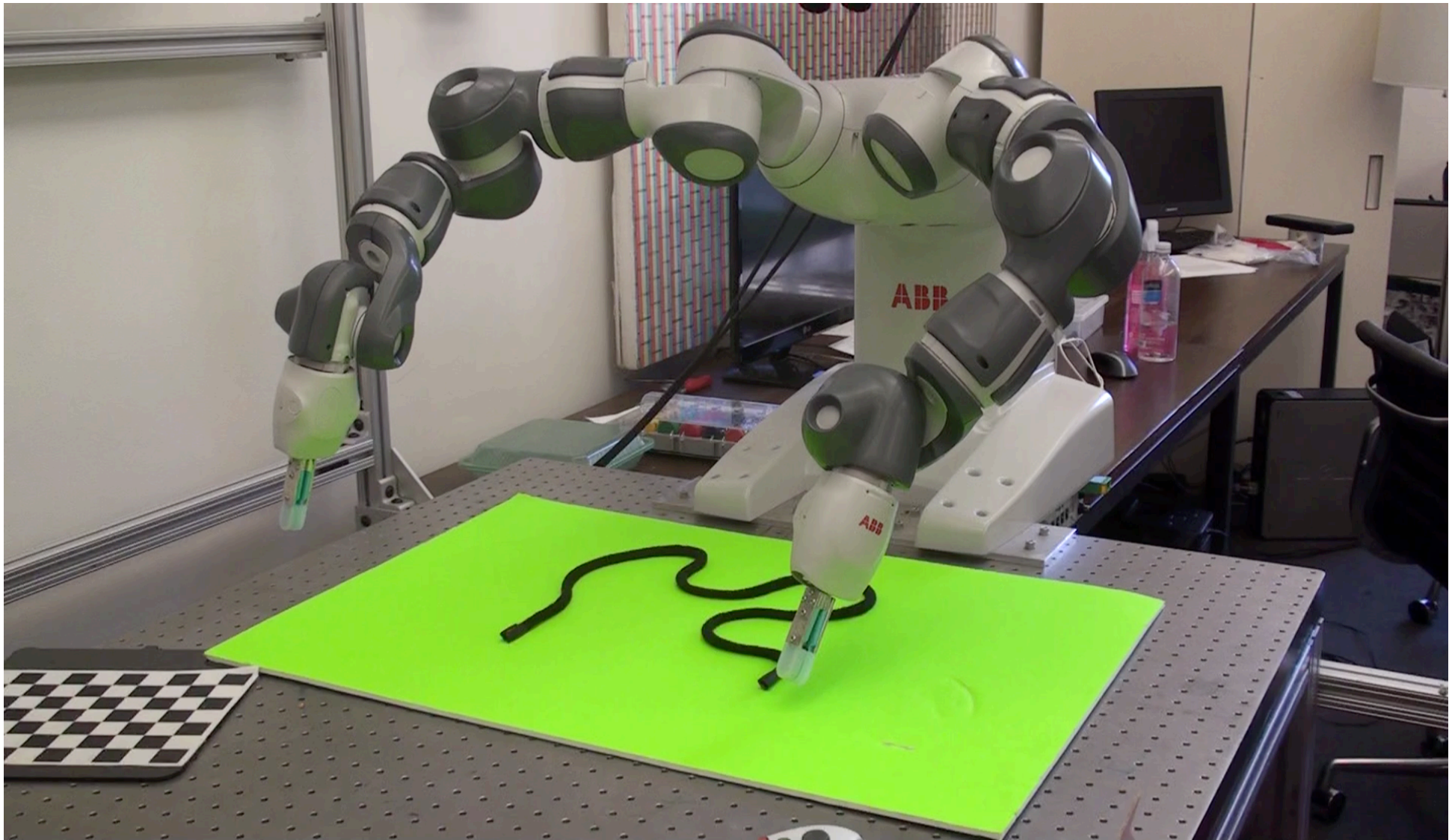
Train to Grasp Endpoints



Minimize Variance By Going to a Fixed State

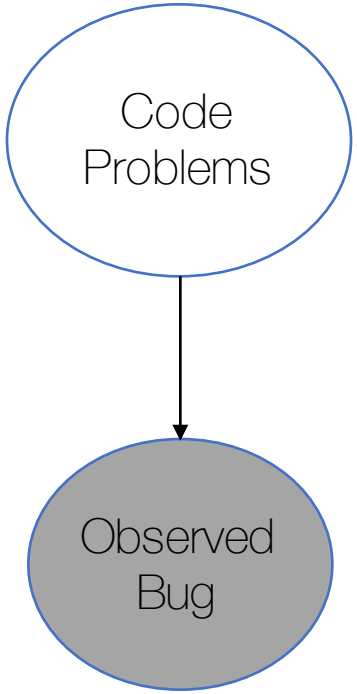


Case Study: Rope Tying

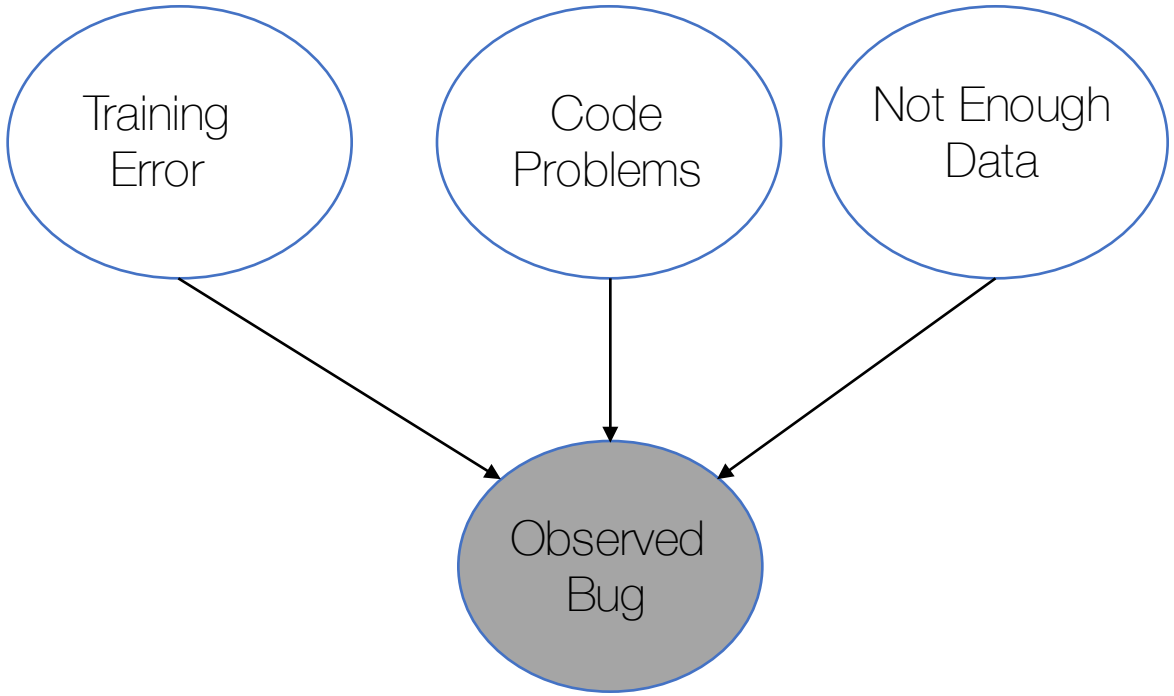


Debugging Your Robot

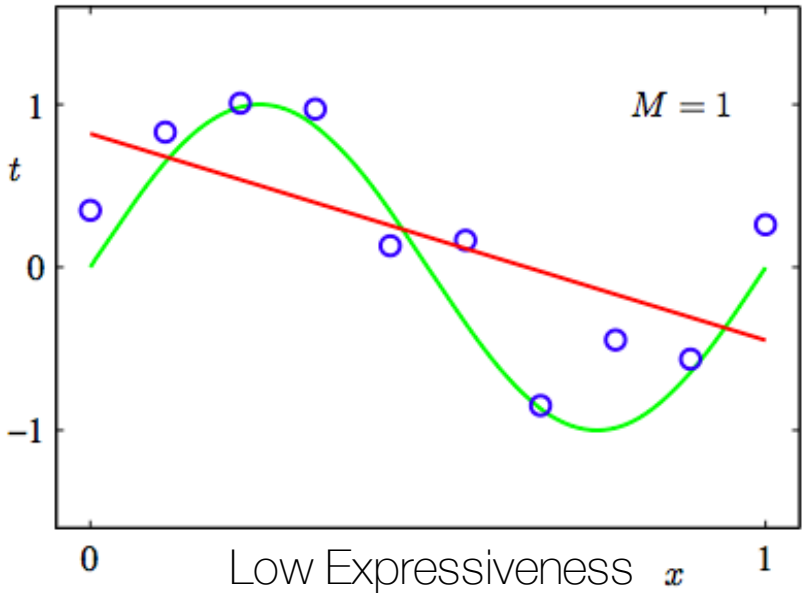
Debugging Code



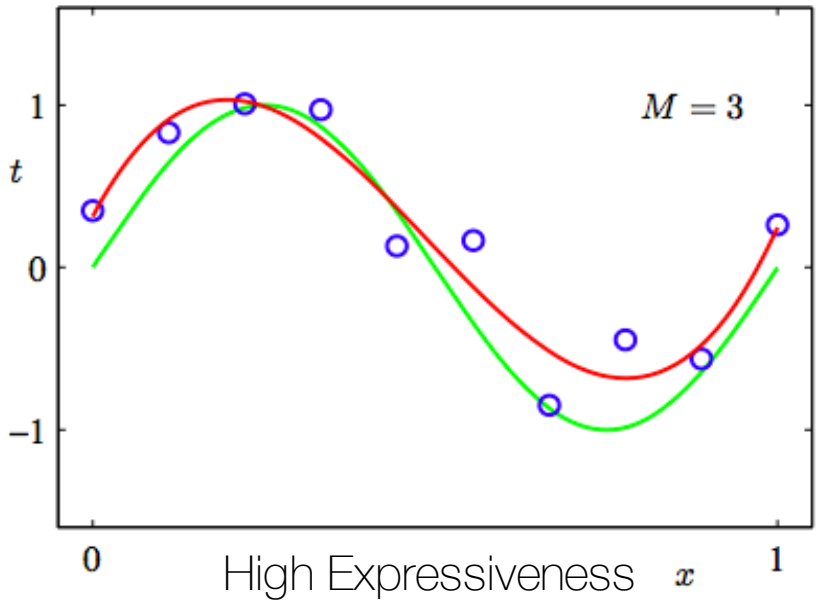
Machine Learning Debugging



Cause1: Too High Training Error

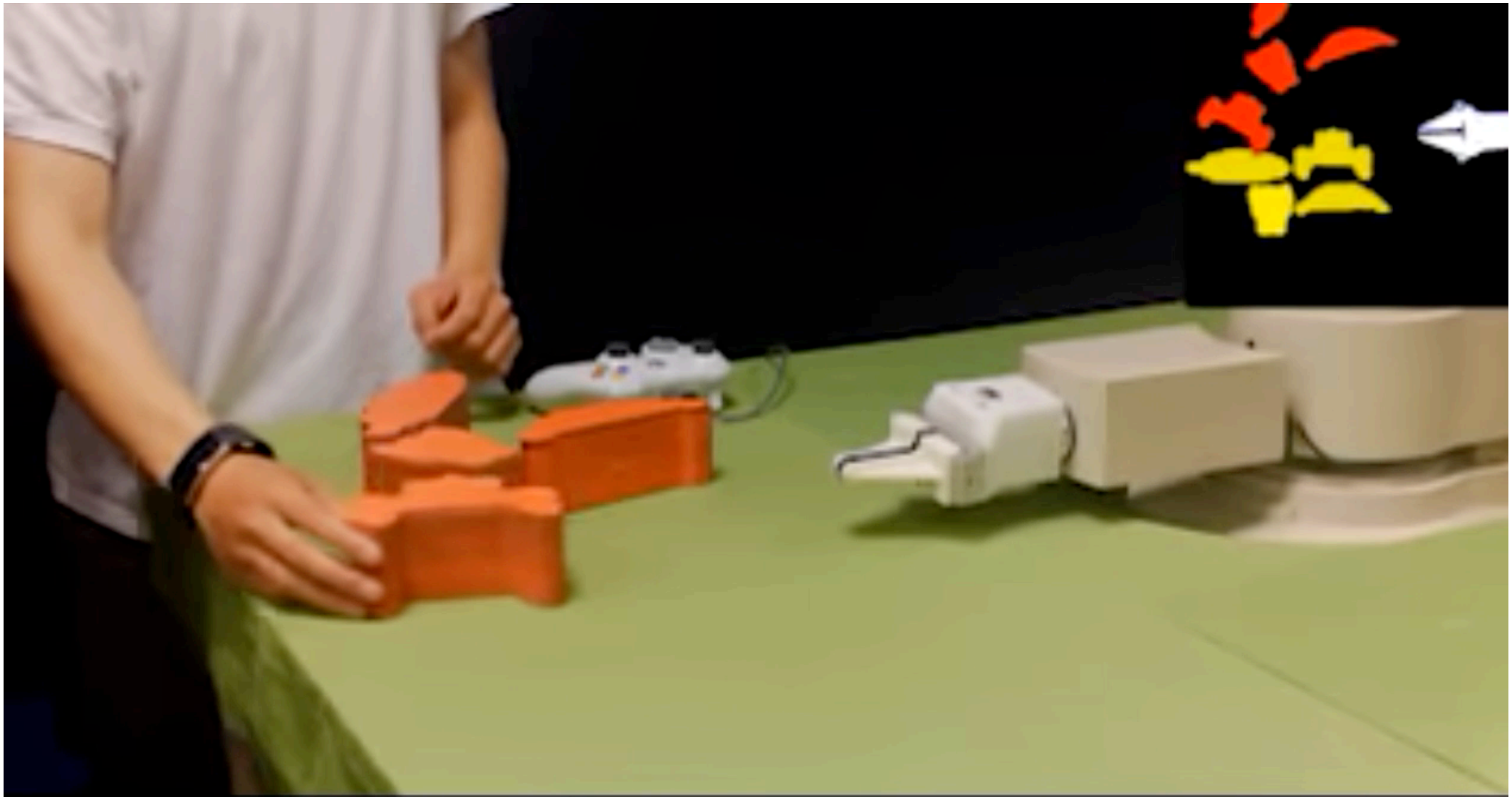


Low Expressiveness



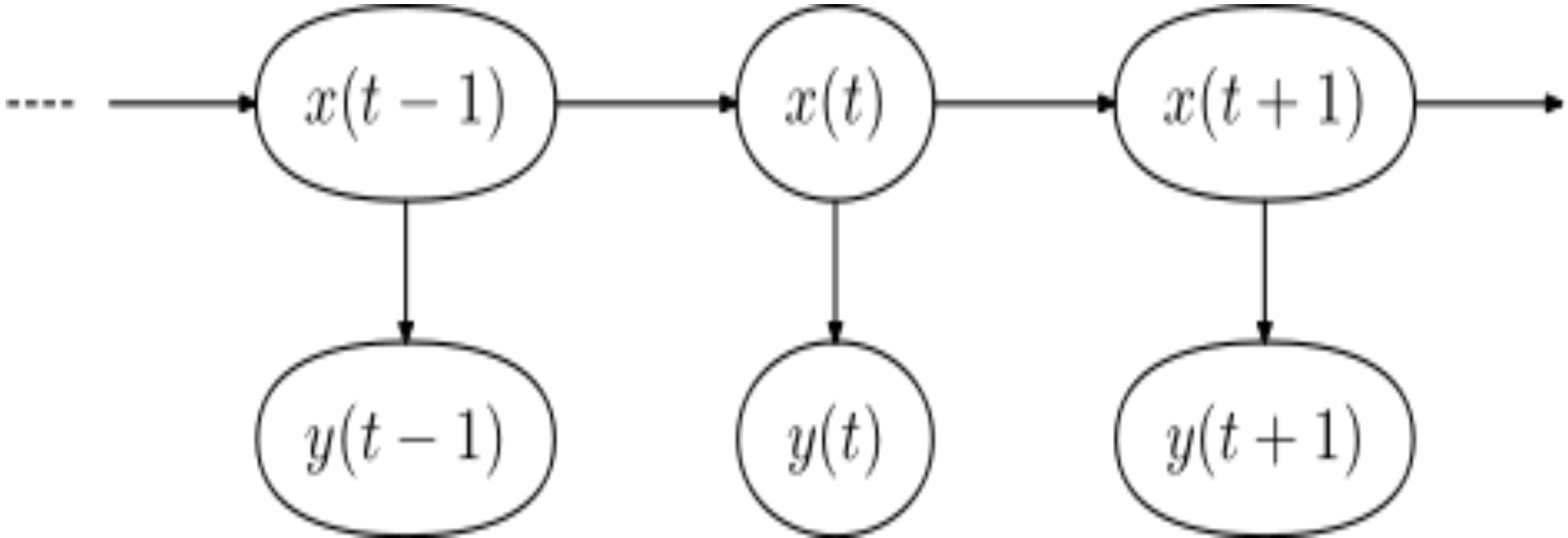
High Expressiveness

Test if Training Error is High



Robot Doesn't Have Low Enough Training Error

Is Your State Space Markovian?



Robot Doesn't Have Low Enough Training Error

Were you supervisor's noisy?

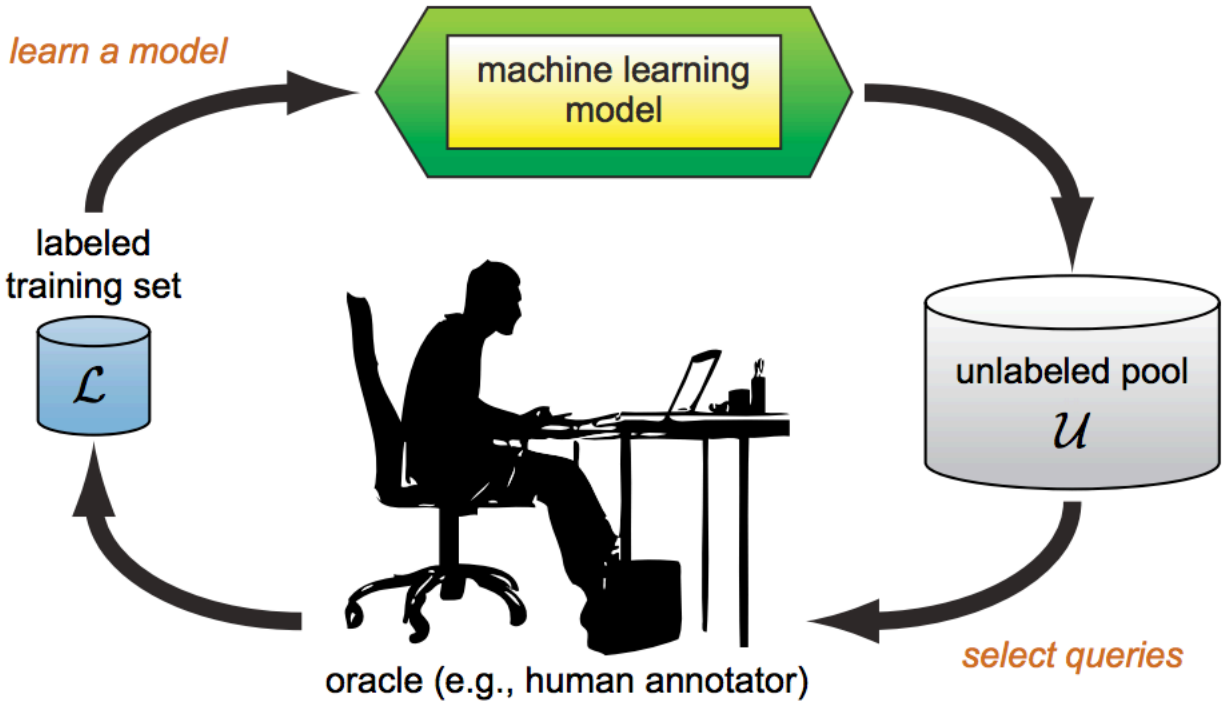
Robot Doesn't Have Low Enough Training Error

Iterate on Network Architecture (i.e. Black Magic)

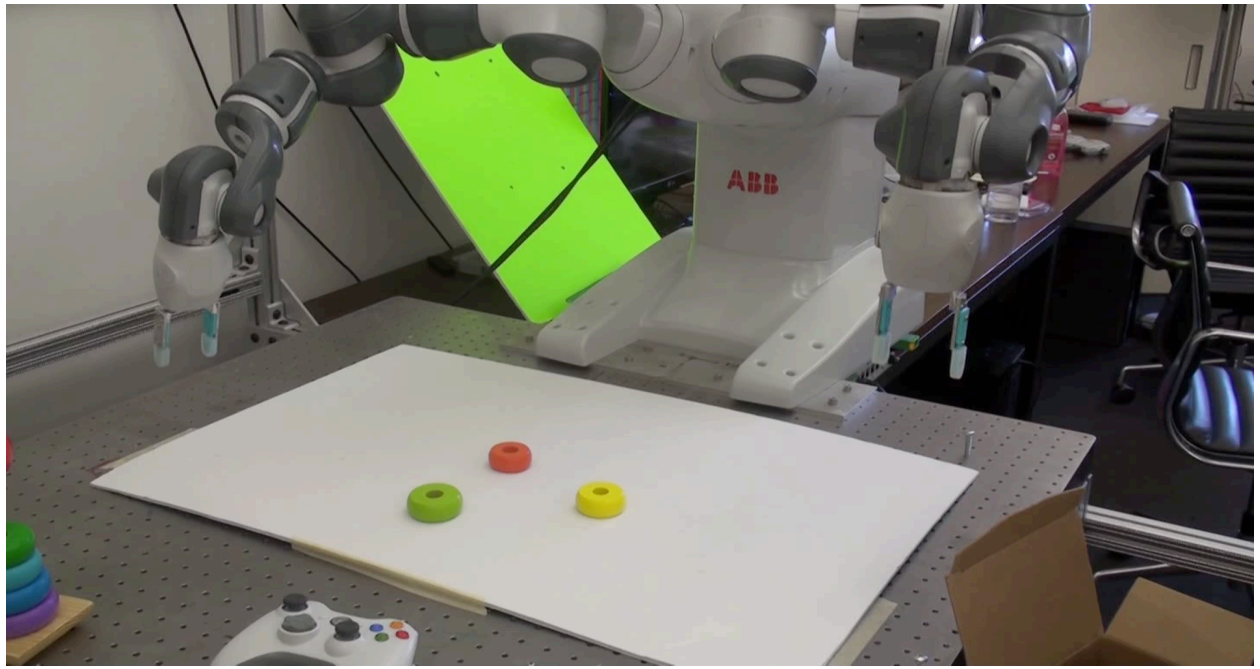
Cause2: Robot Does Have Good Test Error

1) Collect More Data ☹

2) Try Active Learning



Come Meet the Overlords!



AUTOLAB

Email the Following if Interested in Volunteering:

laskeymd@berkeley.edu
jmahler@berkeley.edu