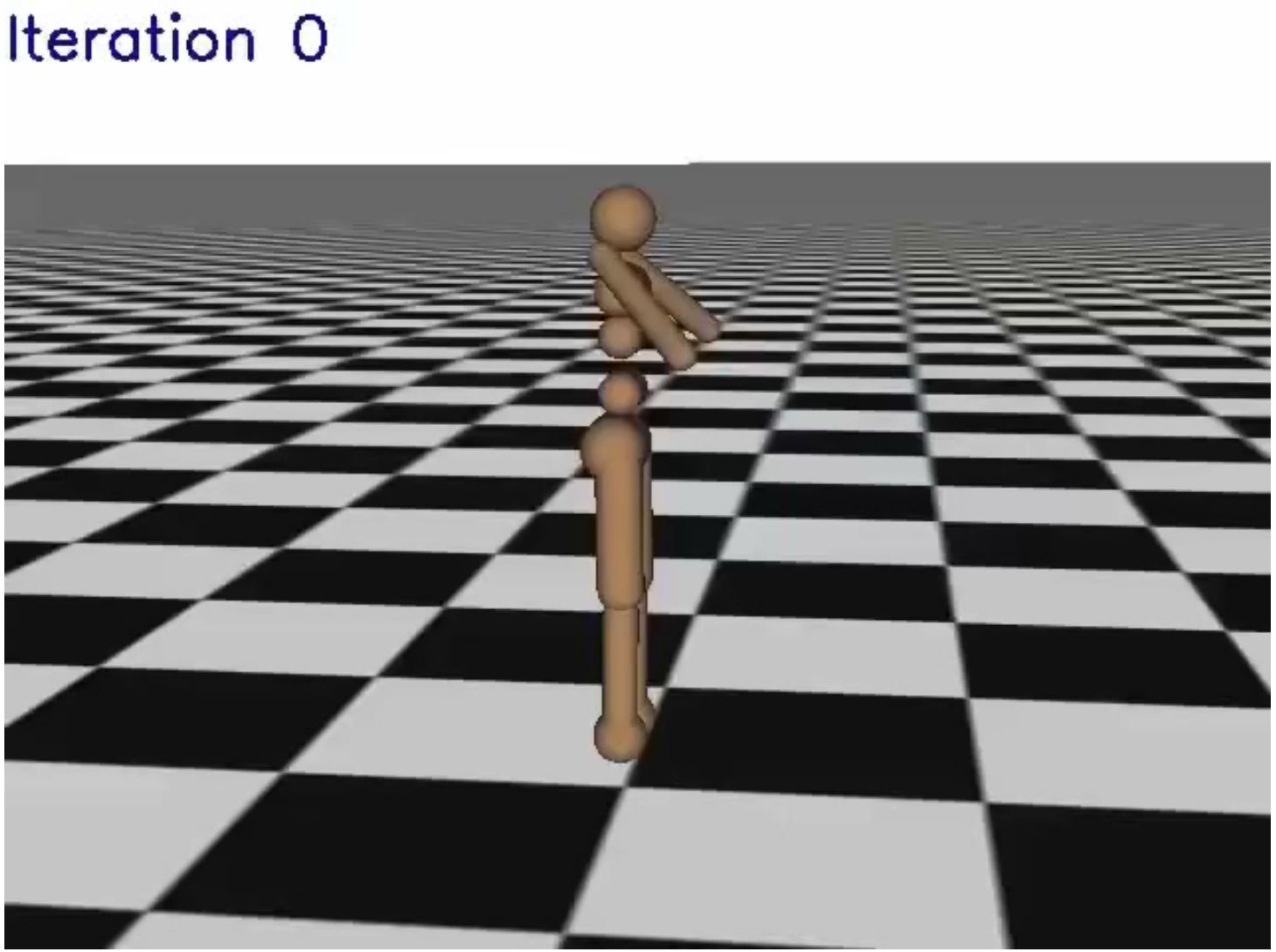
How can we take deep RL into the real world?

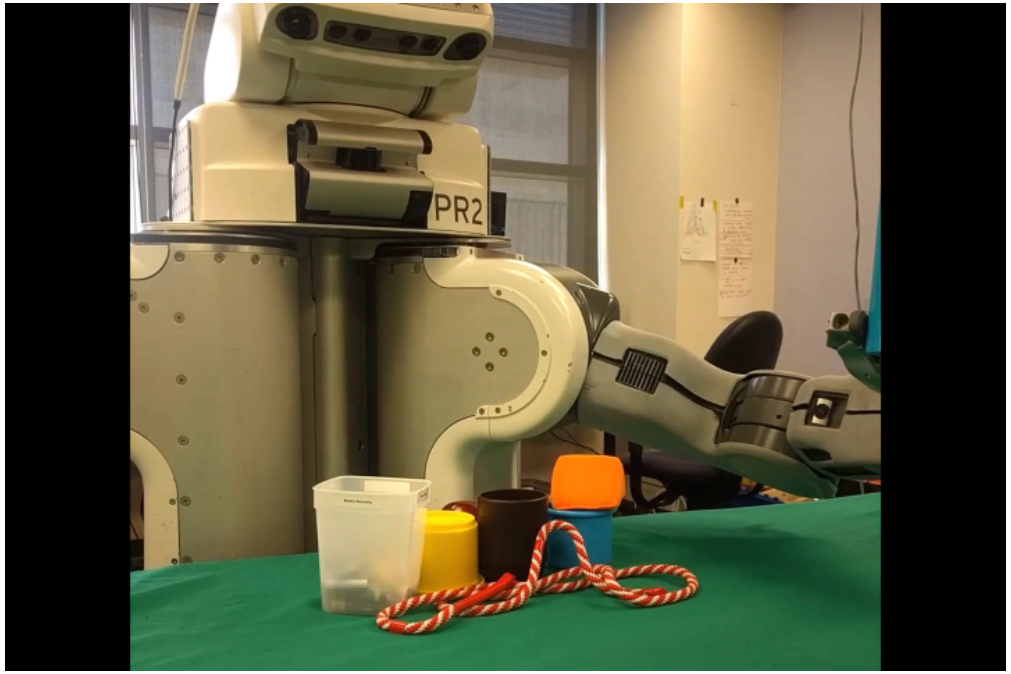
Sergey Levine

UC Berkeley **Google Brain**



Schulman, Moritz, L., Jordan, Abbeel ’16

Levine*, Finn*, et al. ‘16



Devin et al. ‘17

Can’t build a system that doesn’t make mistakes!
But maybe we can learn from those mistakes.

What does it take for RL to learn from its mistakes in the **real** world?

Focus on **diversity** rather than just **proficiency**

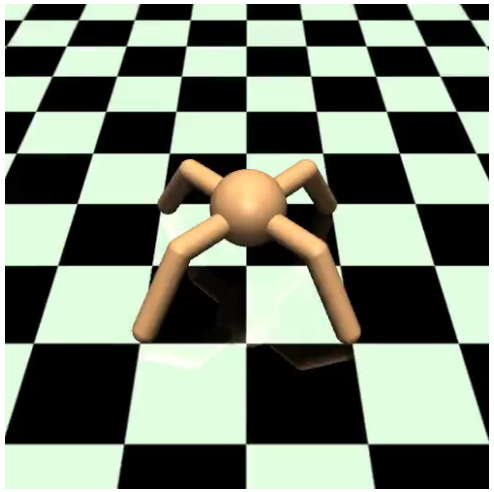
Self-supervision: must have access to reward

Learn quickly and efficiently

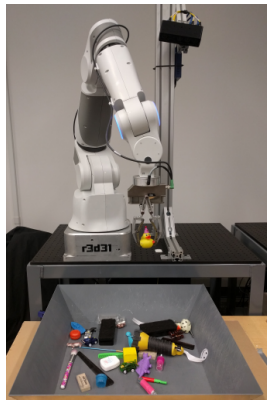
Fix mistakes when they happen

The trouble with RL

- Large-scale
- Emphasizes diversity
- Evaluated on generalization

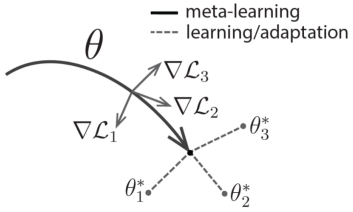


- Small-scale
- Emphasizes mastery
- Evaluated on performance

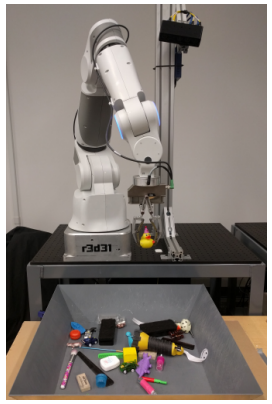


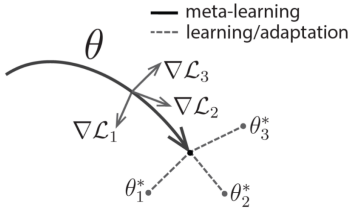
1. Can we self-supervise diverse real-world tasks?

2. Where can we get goal supervision?



3. How can new tasks build on prior tasks?



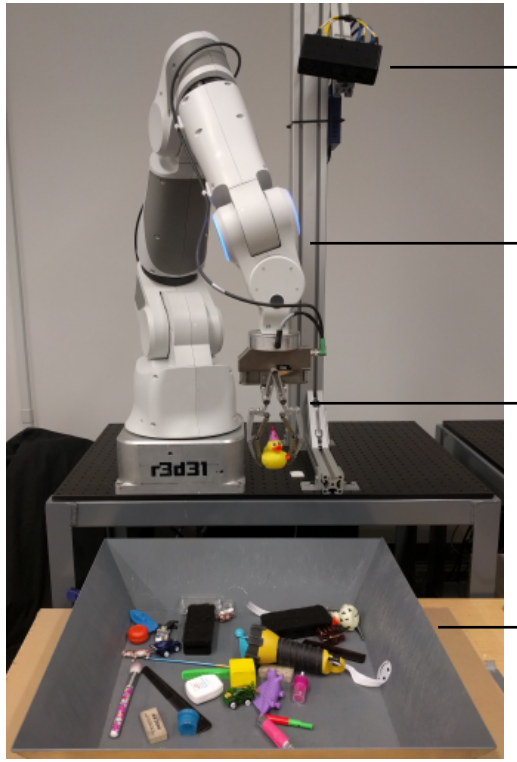


1. Can we self-supervise diverse real-world tasks?
2. Where can we get goal supervision?
3. How can new tasks build on prior tasks?

What can we do with self-supervised learning?

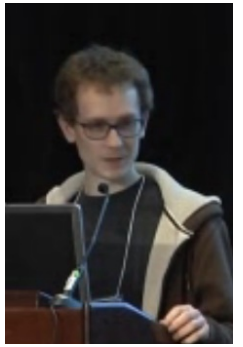
Grasping with Learned Hand-Eye Coordination

- monocular camera (no depth)
  - no camera calibration either
- 2-5 Hz update
  - continuous arm control
  - servo the gripper to target
  - fix mistakes
- no prior knowledge



Peter Pastor

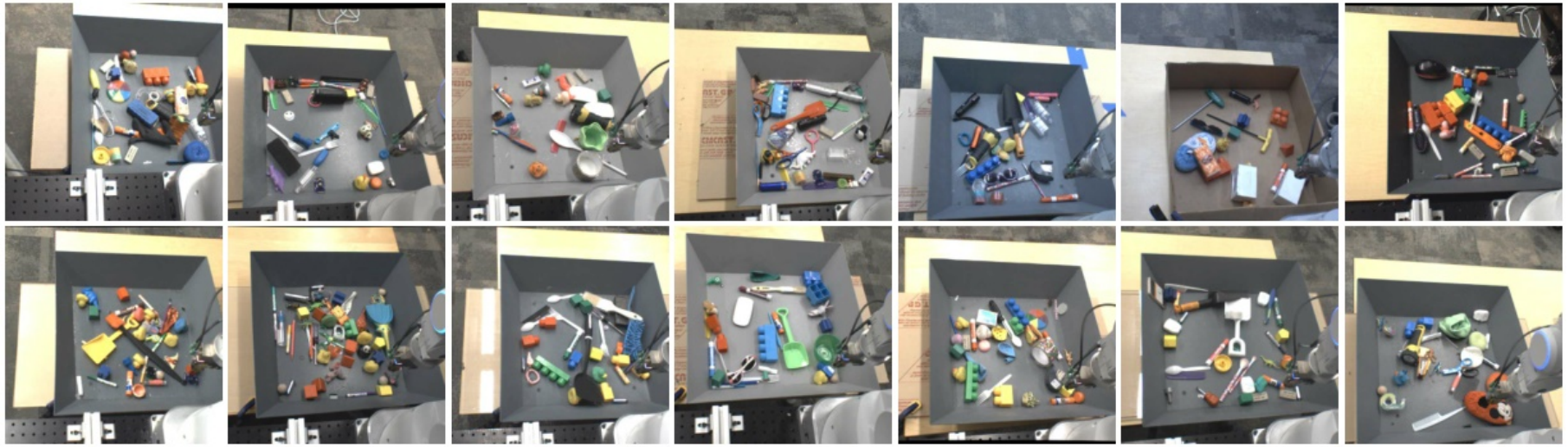
Alex
Krizhevsky



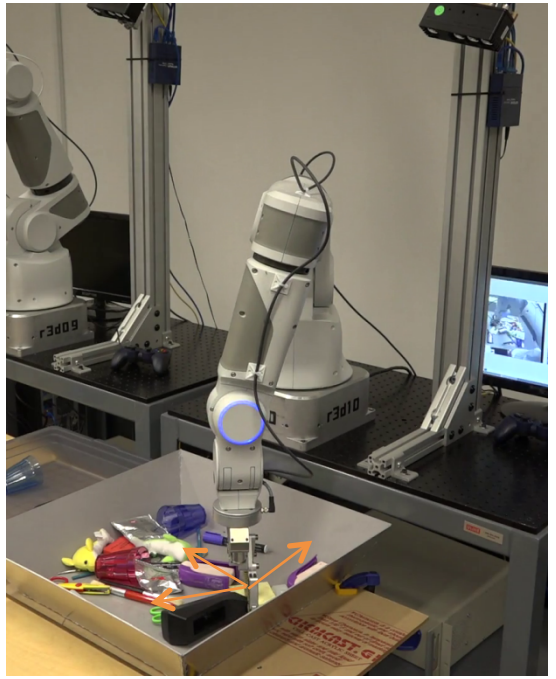
Deirdre Quillen

L., Pastor, Krizhevsky, Quillen ‘16

Using Grasp Success Prediction



training



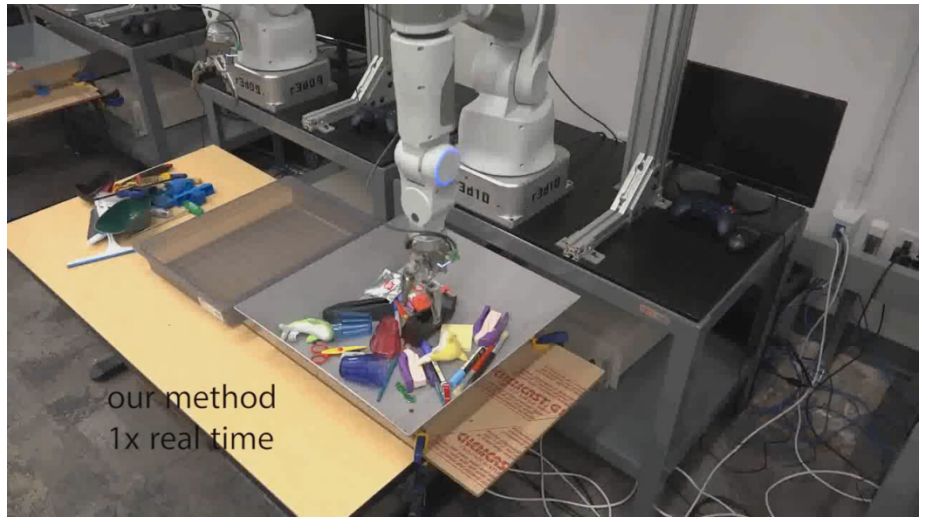
testing

Open-Loop vs. Closed-Loop Grasping

open-loop grasping

failure rate: 33.7%

closed-loop grasping



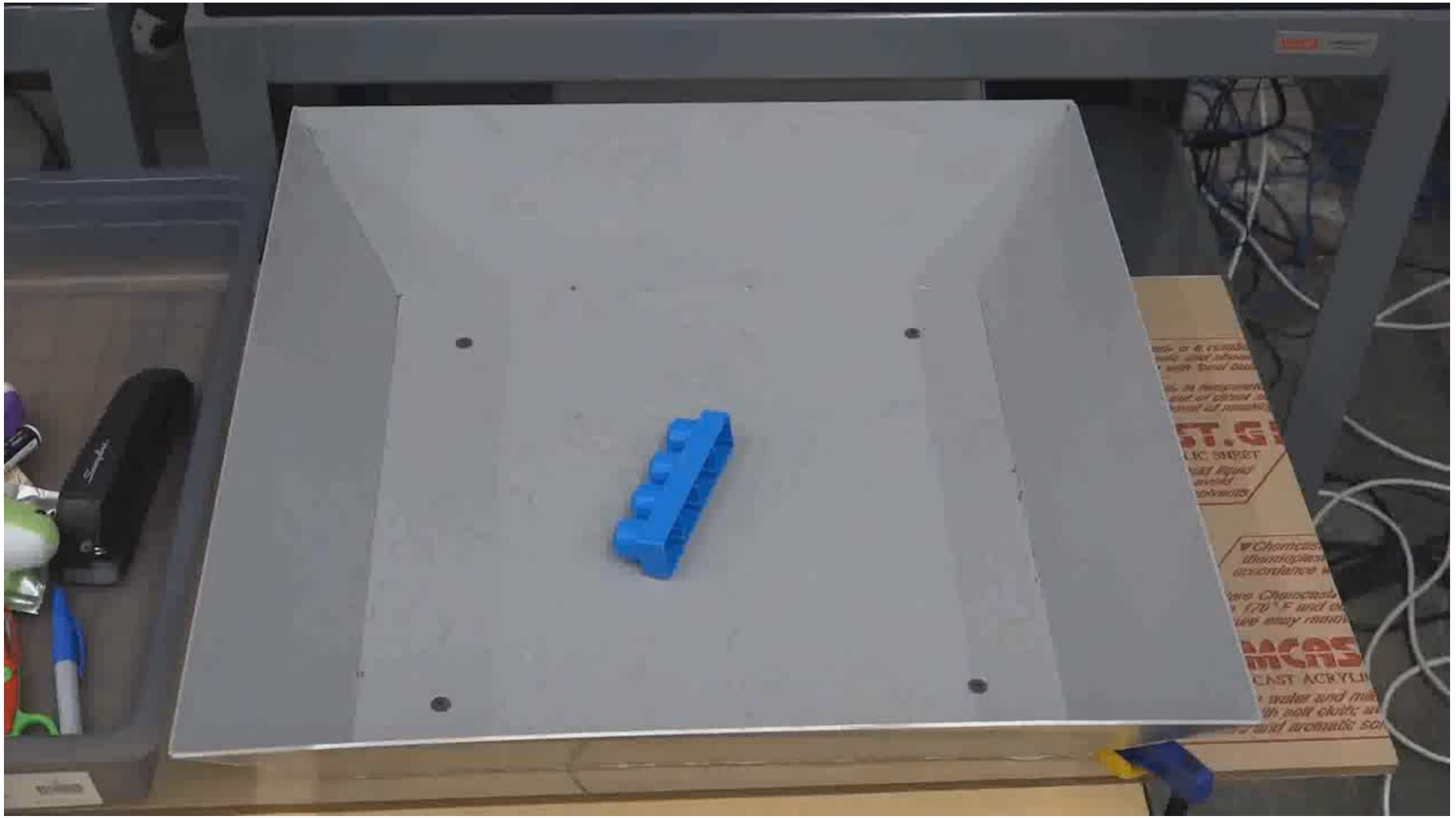
failure rate: 17.5%

Pinto & Gupta, 2015

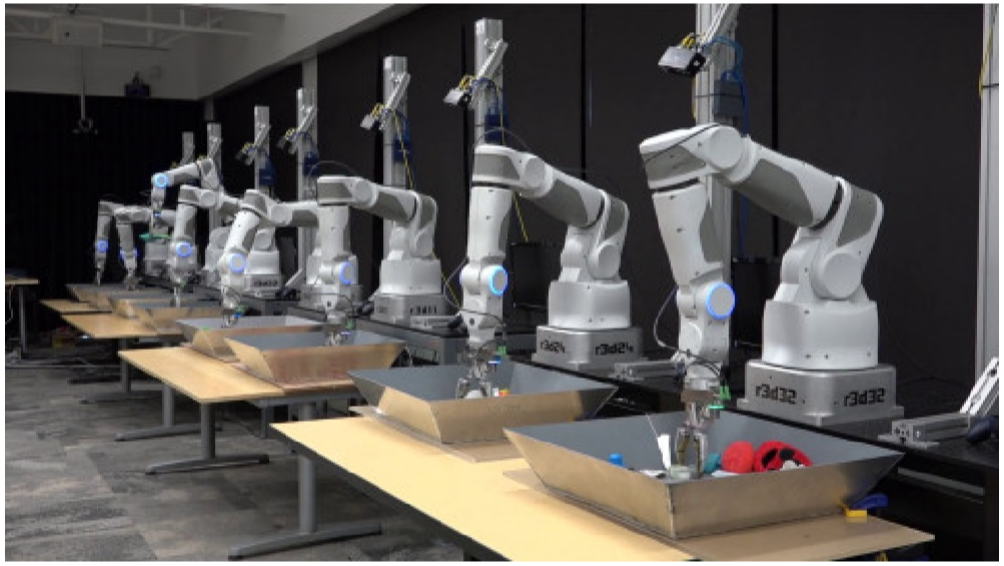
depth + segmentation
failure rate: 35%

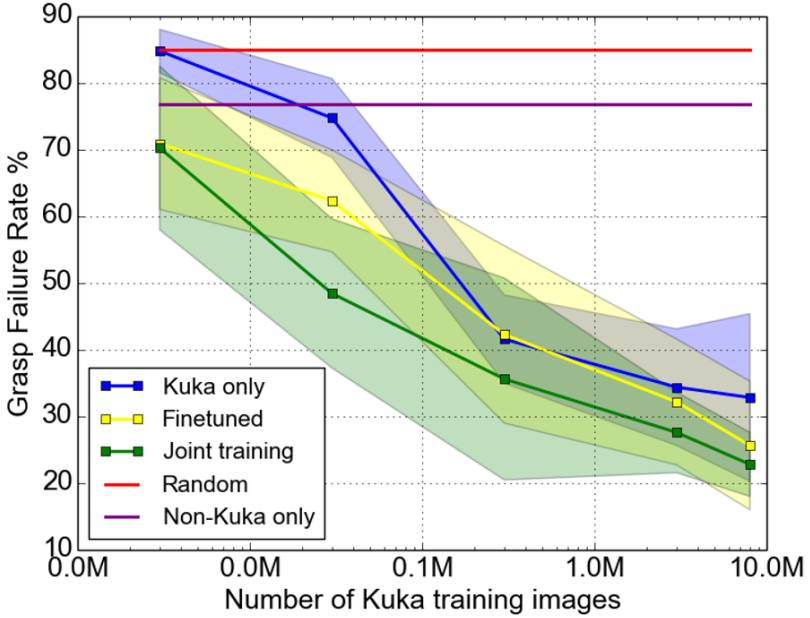
1. Can a learned grasping system generalize?
2. Is there benefit in continuous visual feedback?

Grasping Experiments

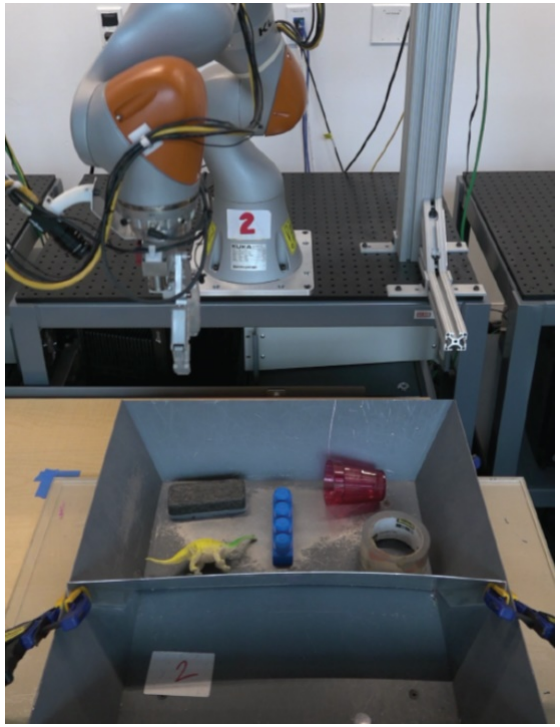


Grasping Experiments

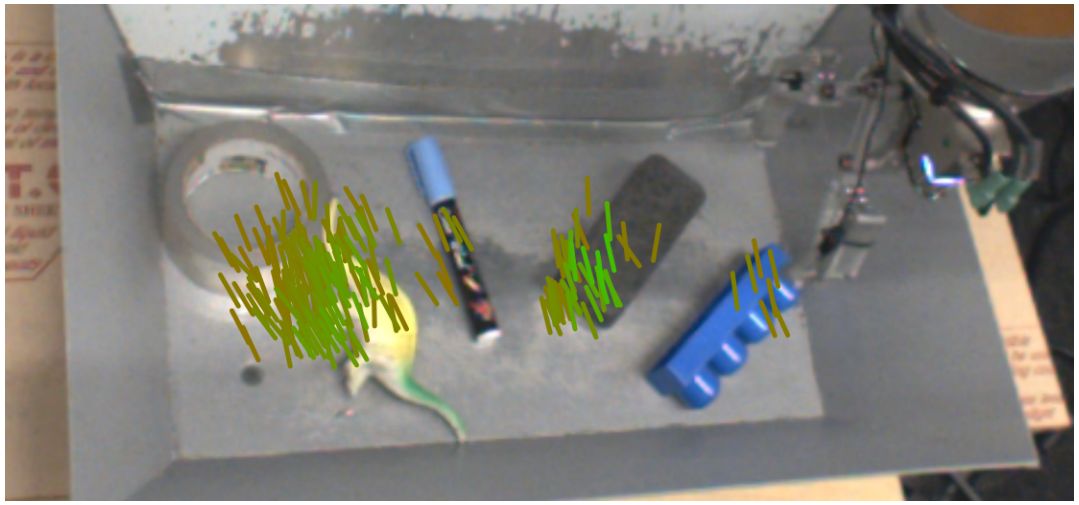


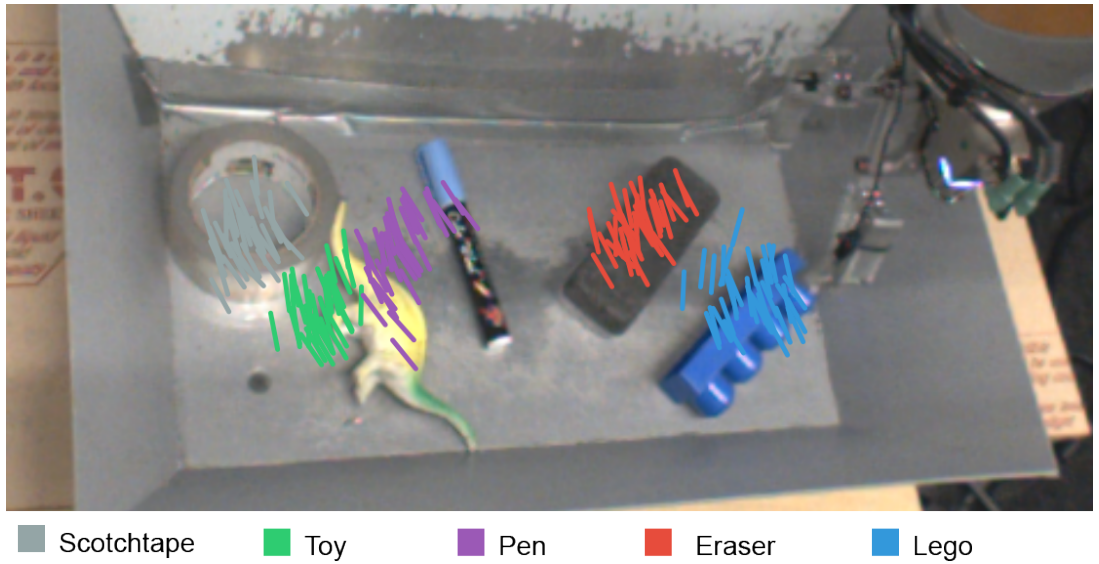


Julian Ibarz



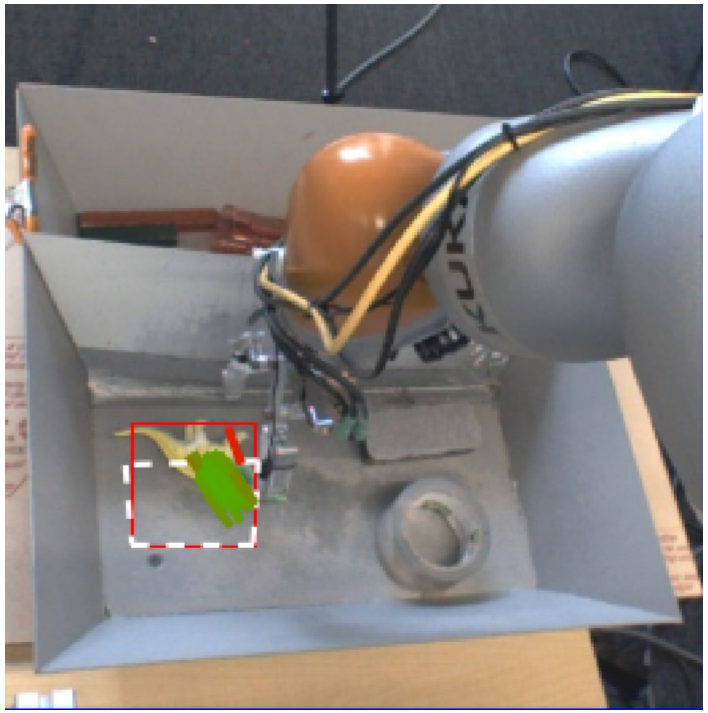
Can we learn to grasp objects from specific categories via self-supervised data collection?





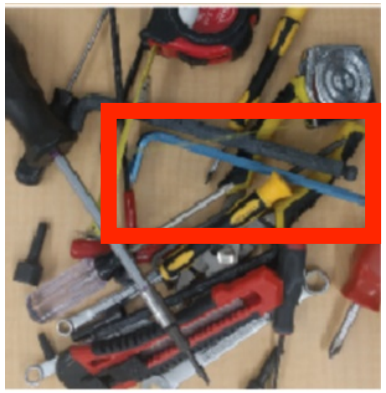
Can we learn to grasp objects from specific categories via self-supervised data collection?

Computer vision approach: detect the object, and then choose grasps only on that object



- Expensive human data (boxes, segmentations) limits dataset size

- Bounding boxes are not enough!

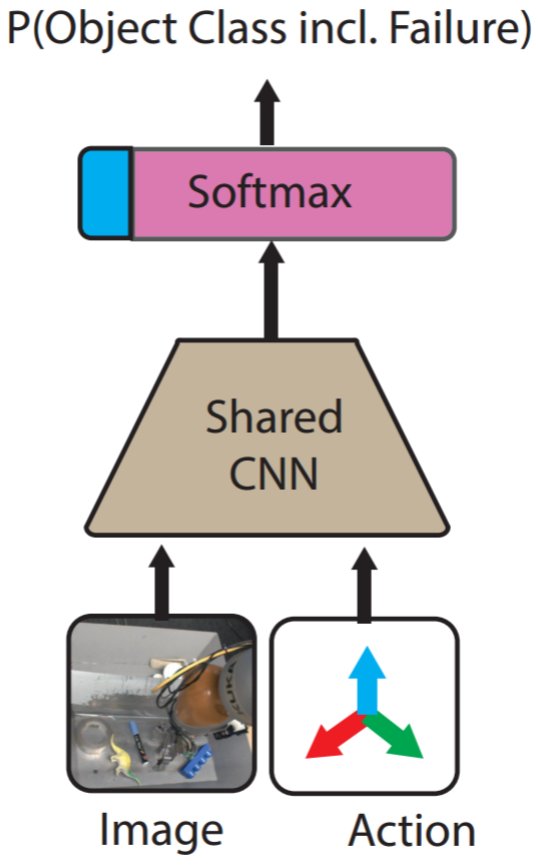


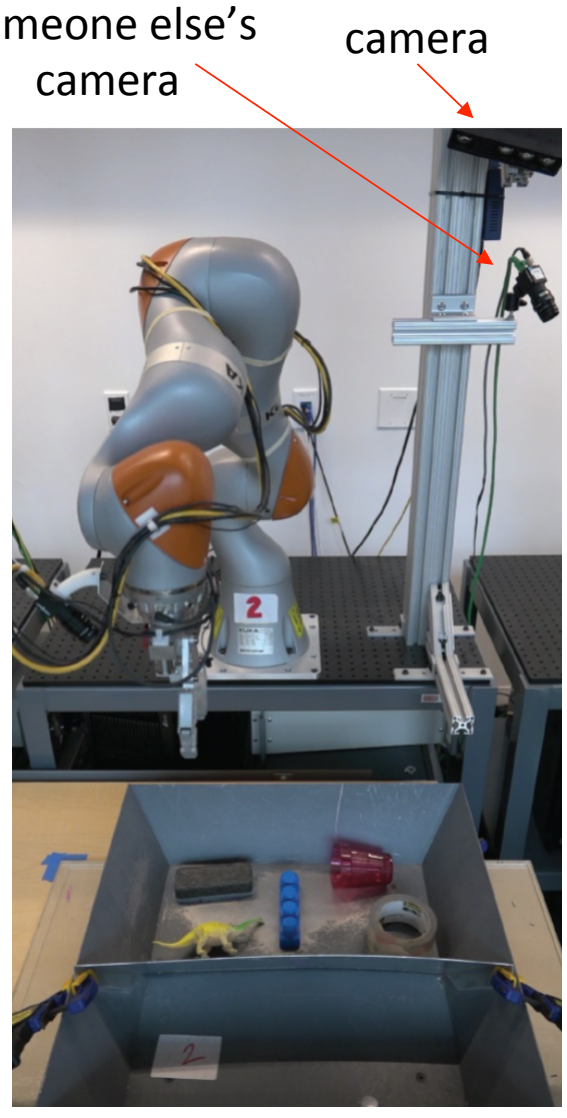
grab the Allen wrench

(good luck)

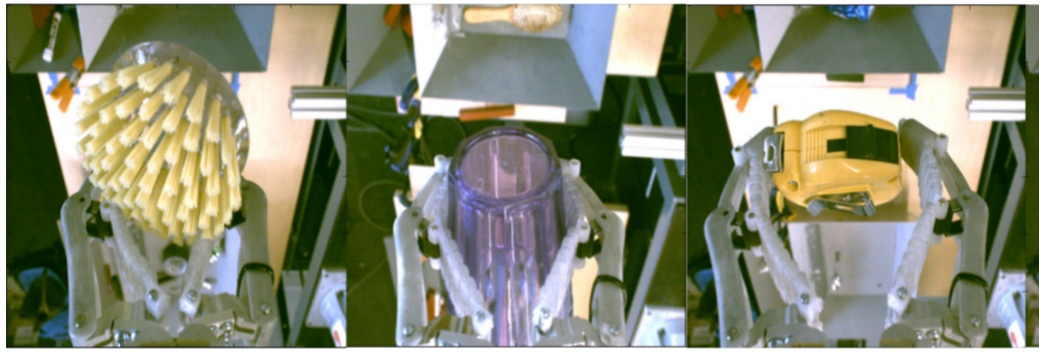
- Too much **and** too little information!

Can we train end-to-end?

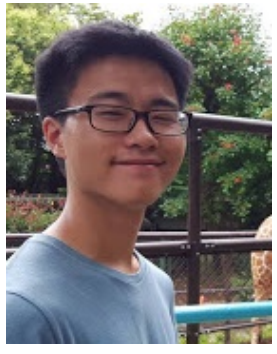




Key idea: label objects **after** they are picked up, and set that label on the entire grasp!



Uses about 20,000 Mechanical Turk labels (about 1.3% of ImageNet/ILSVRC) + 300,000 automatically propagated labels


Eric Jang

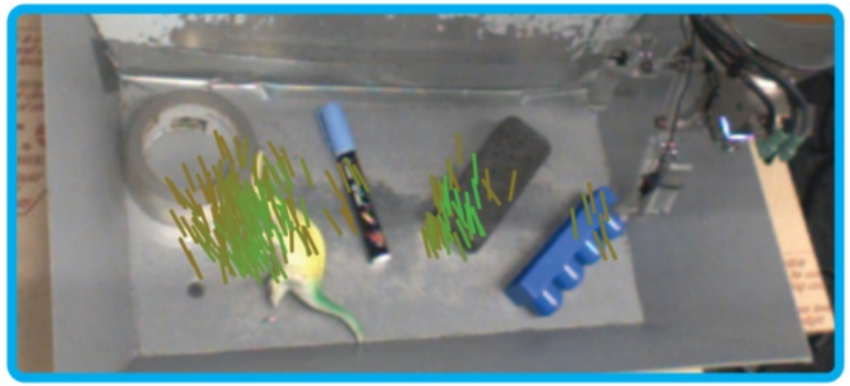

Sudheendra Vijayanarasimhan


Peter Pastor

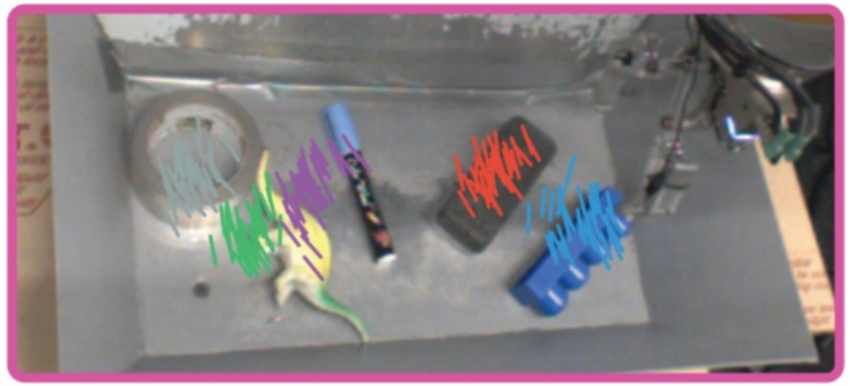

Julian Ibarz

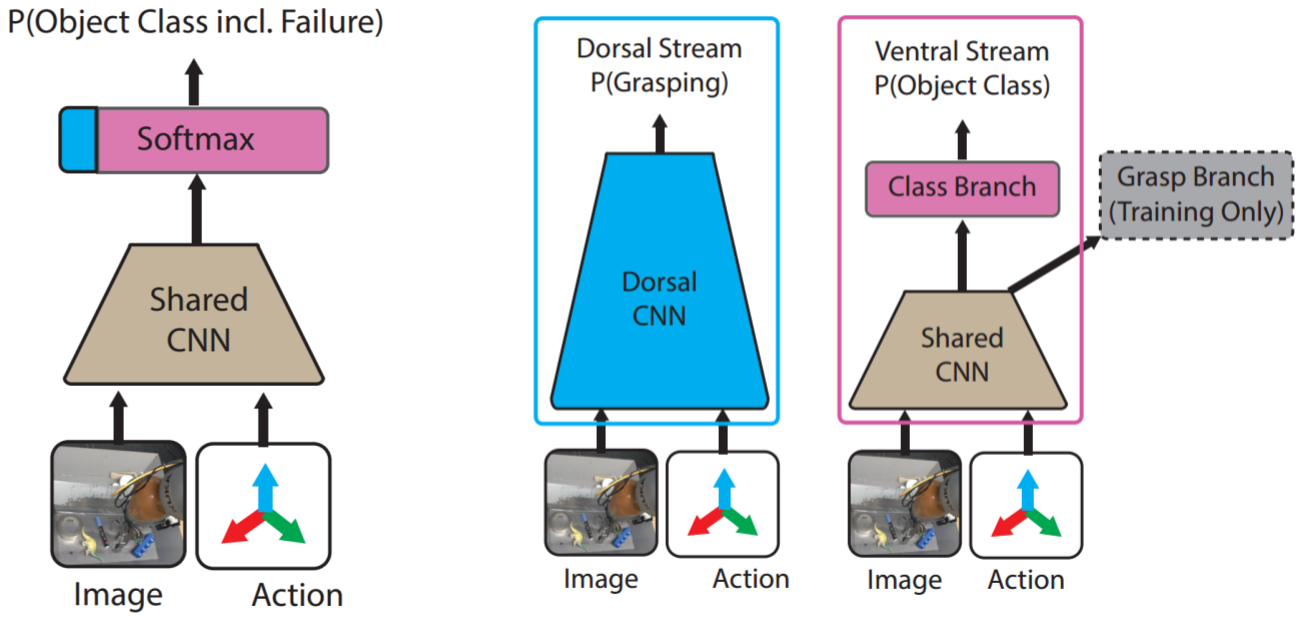
How should we design the network?

P(Grasp success)

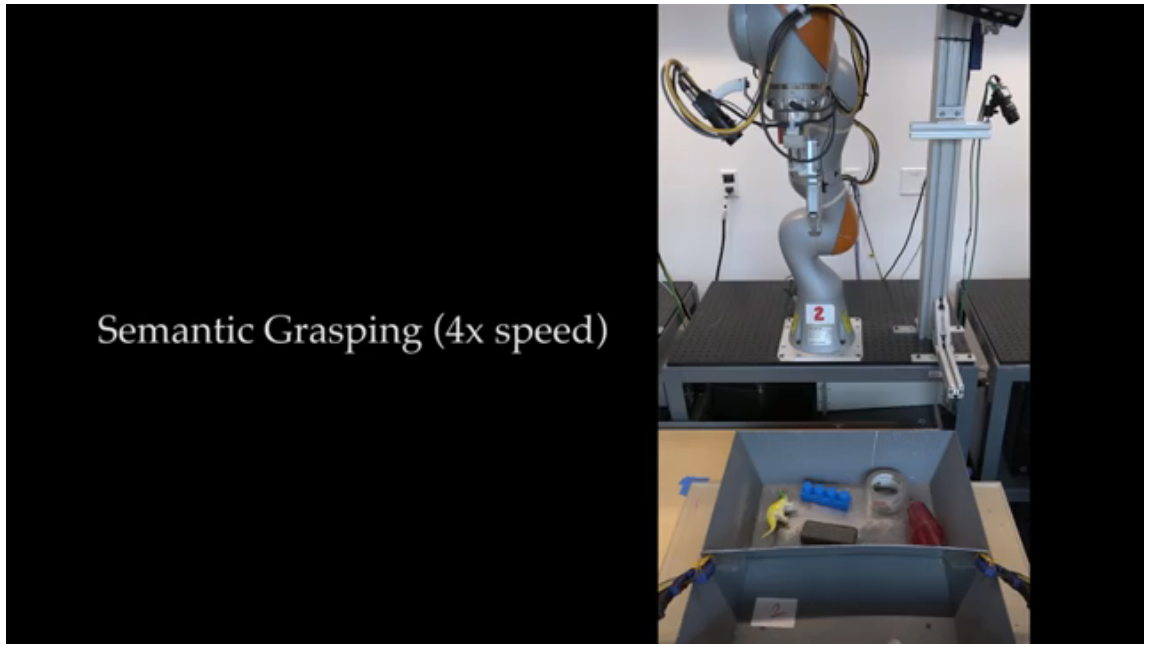


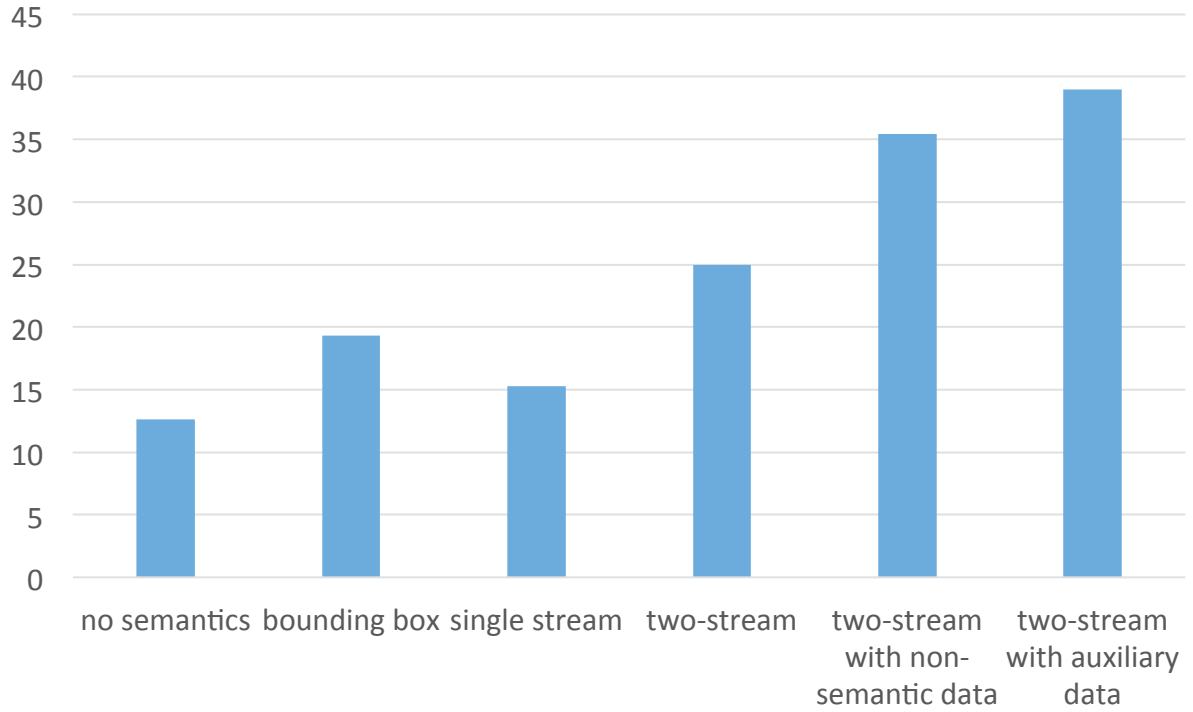
P(Object Class | Grasp success)

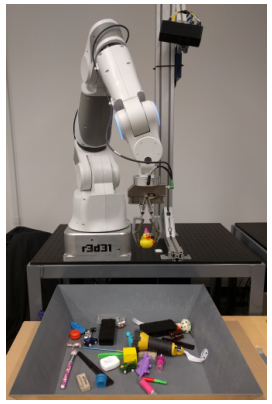




How well does it work?

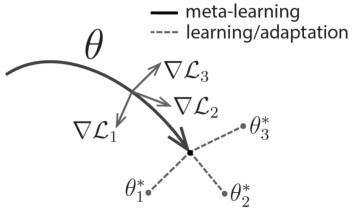






1. Can we self-supervise diverse real-world tasks?

2. Where can we get goal supervision?

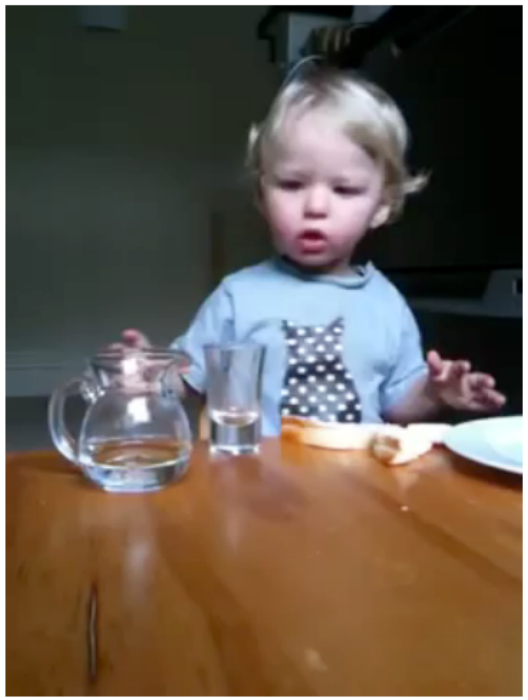


3. How can new tasks build on prior tasks?

reward

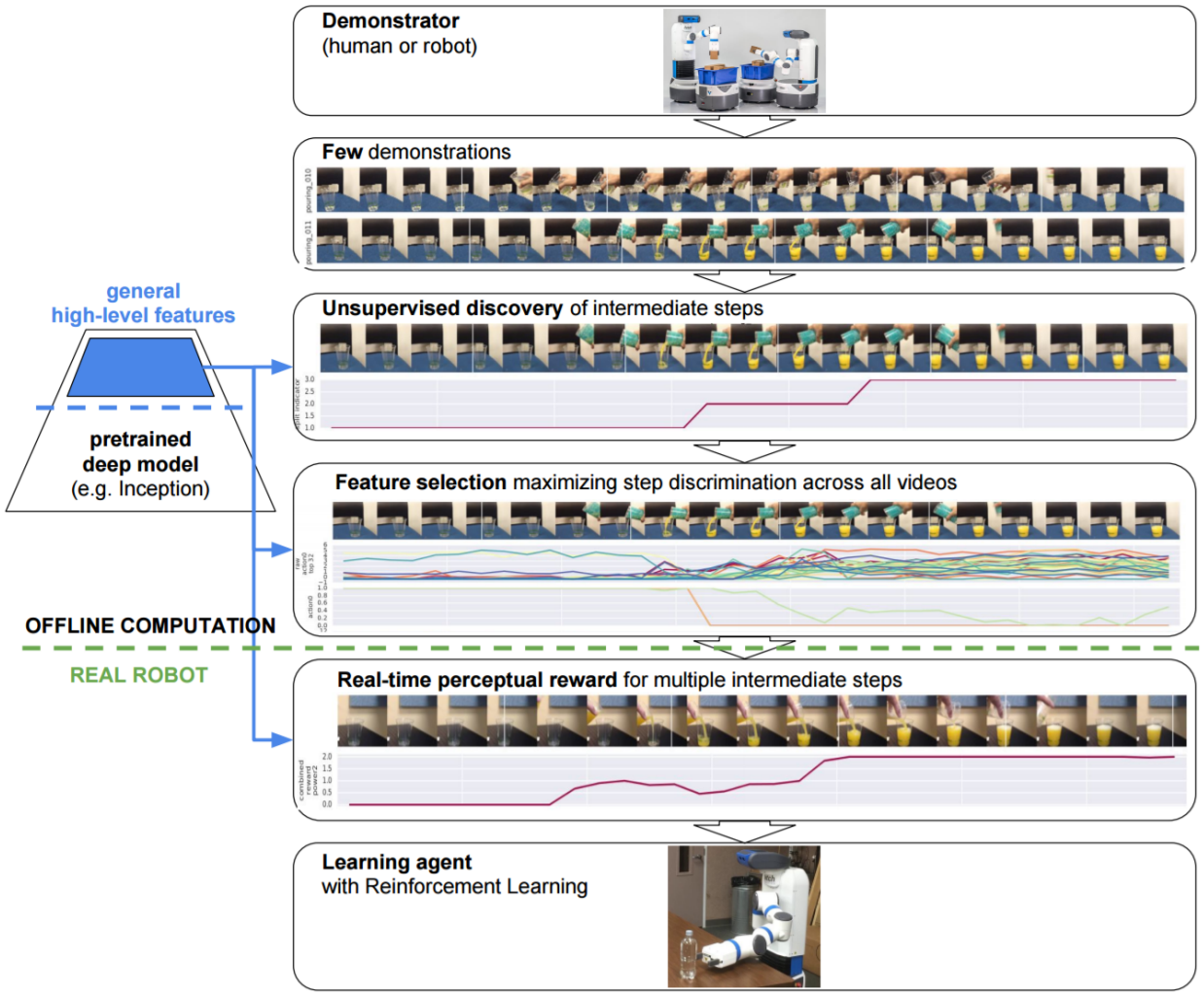
Mnih et al. '15

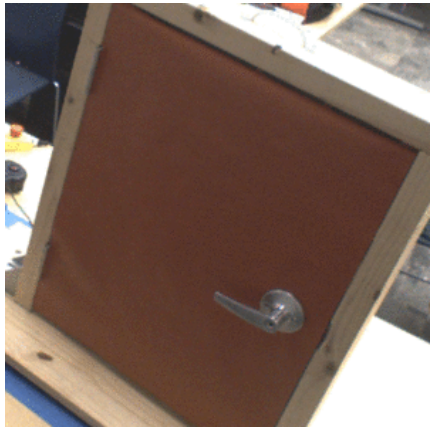
reinforcement learning agent

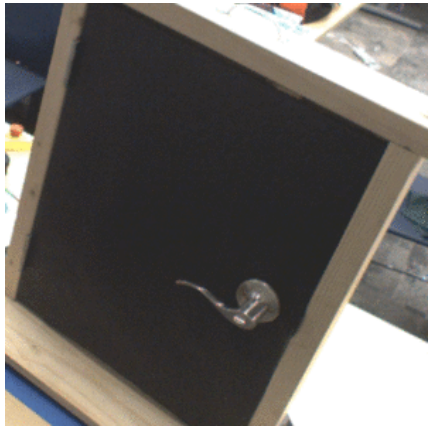


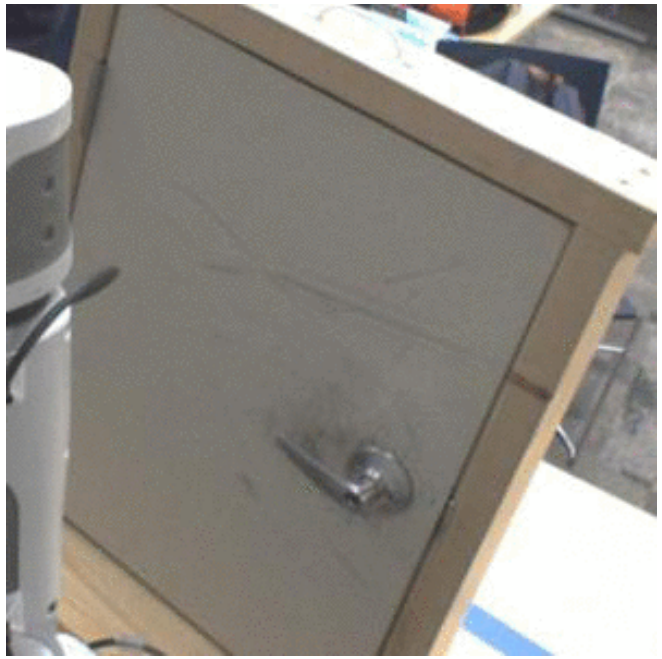
what is the reward?

Learning with Invariant Features



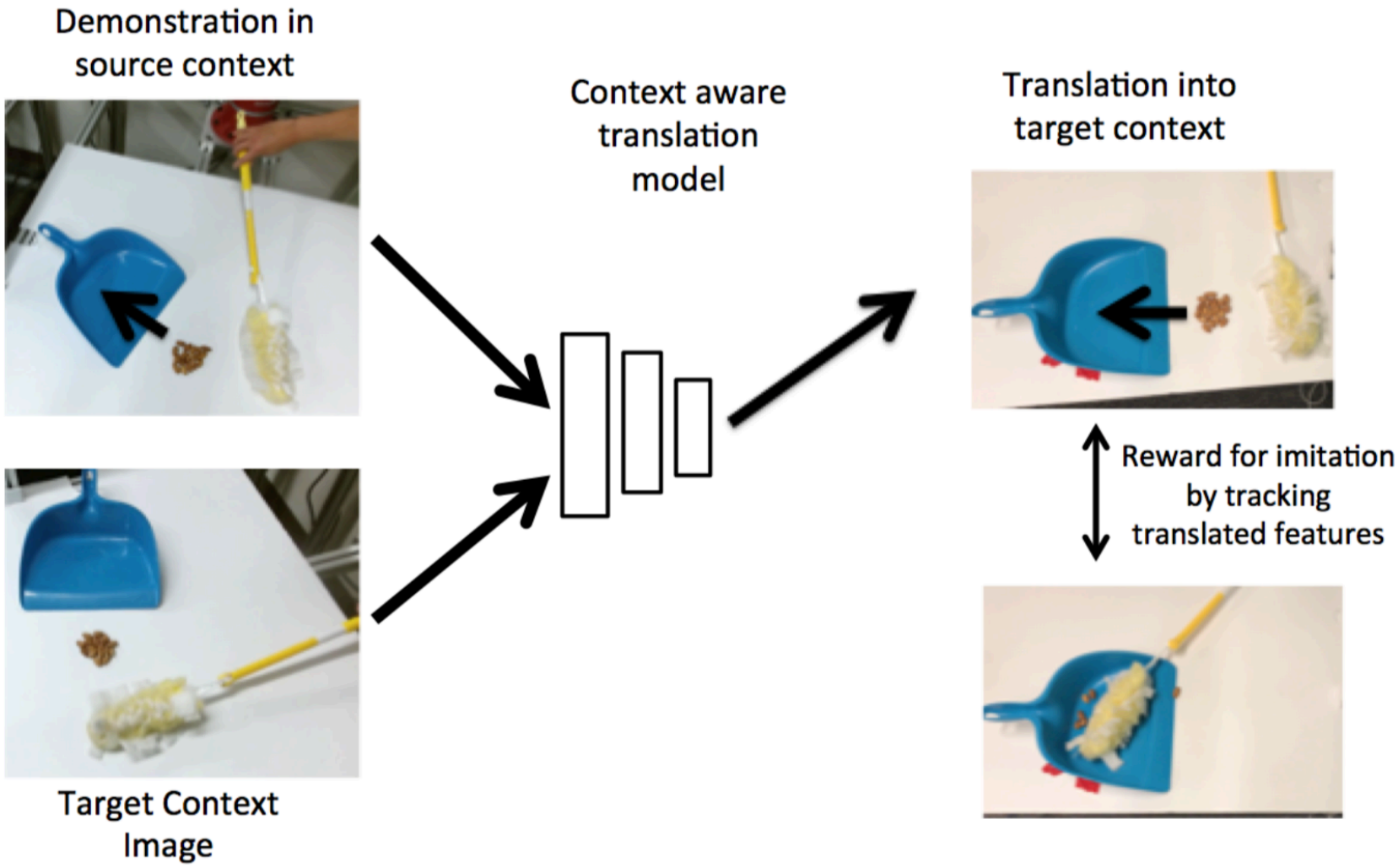






Sermanet, Xu, L. ‘16

Video Translation for Learning Objectives

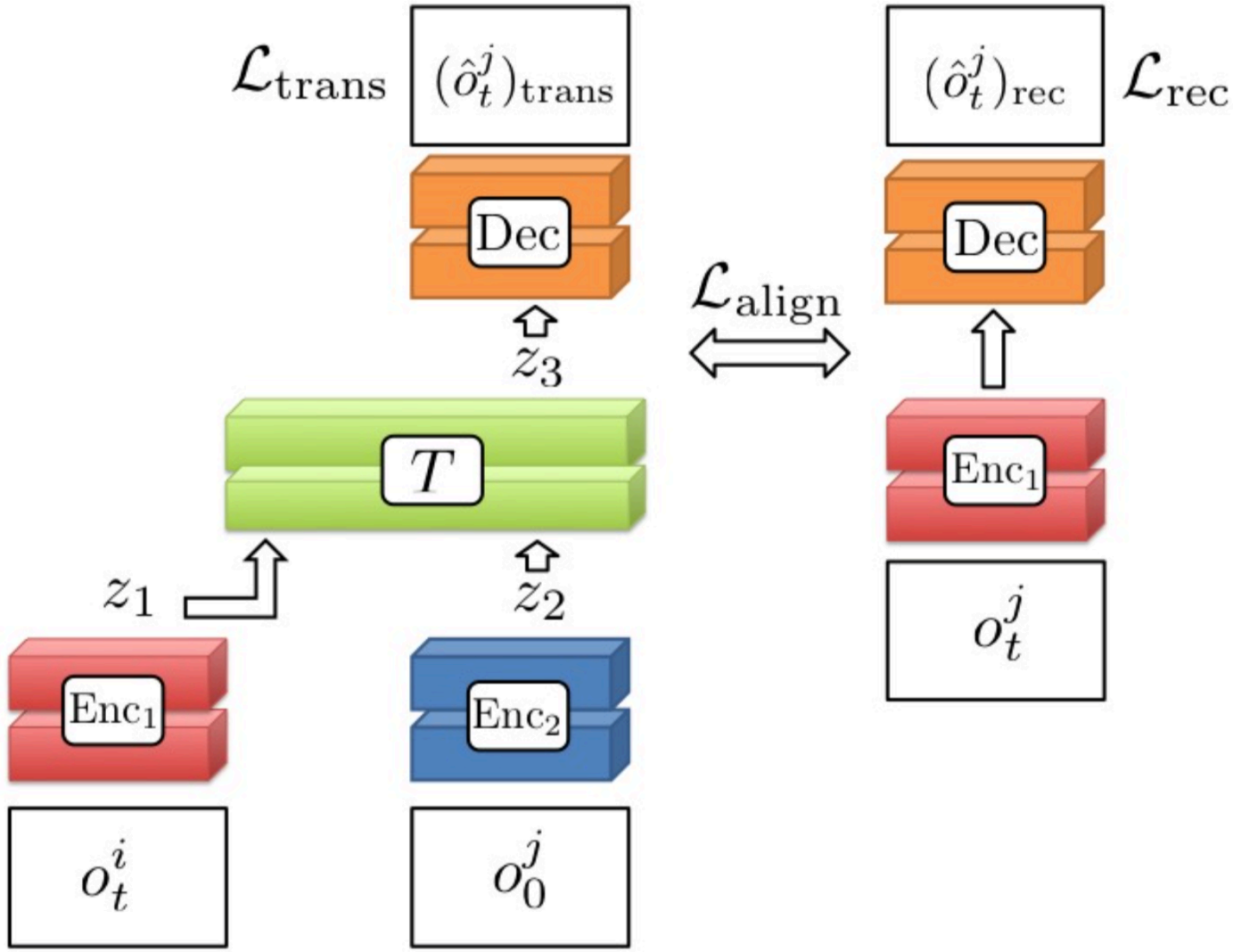


[Liu*, Gupta* et al, 2017]

Abhishek Gupta

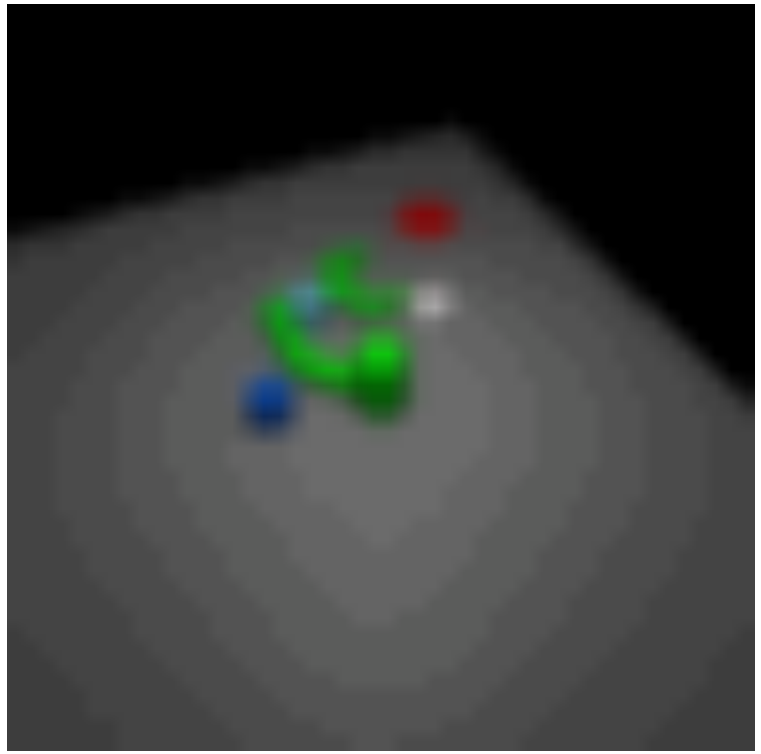
YuXuan Liu

Video Translation Architecture

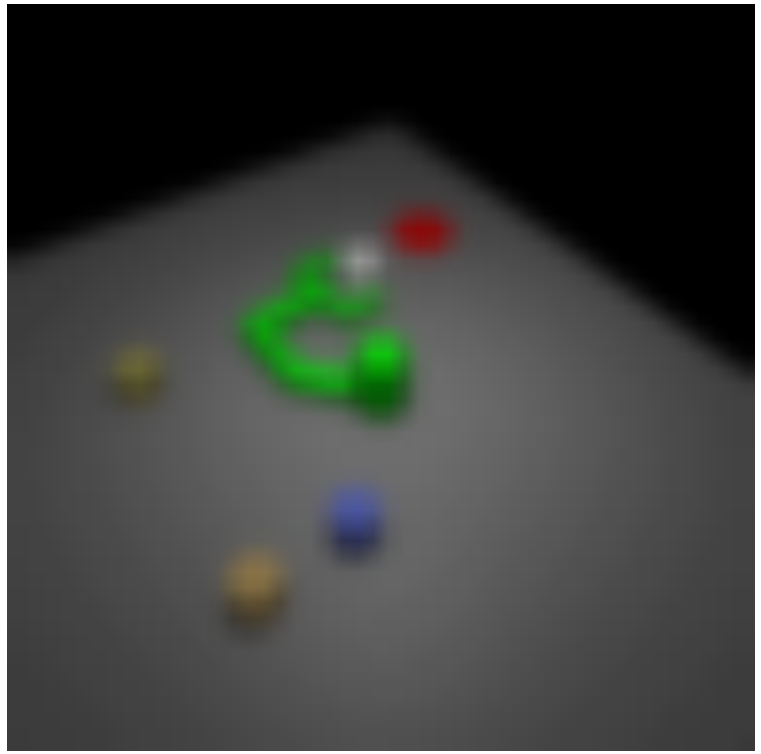


Context Translation Video Prediction Results

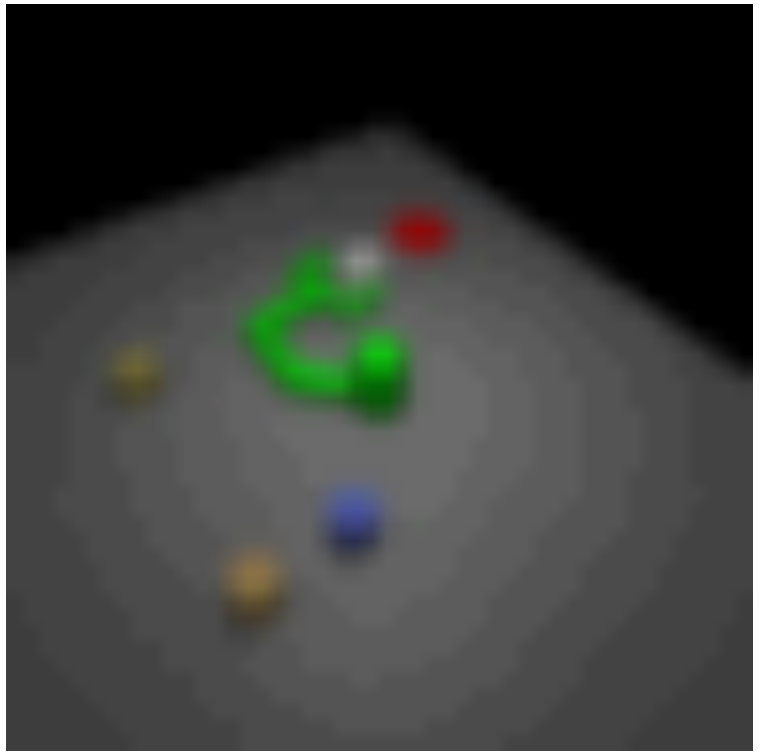
Source Video



Context at time 0



Translation Prediction



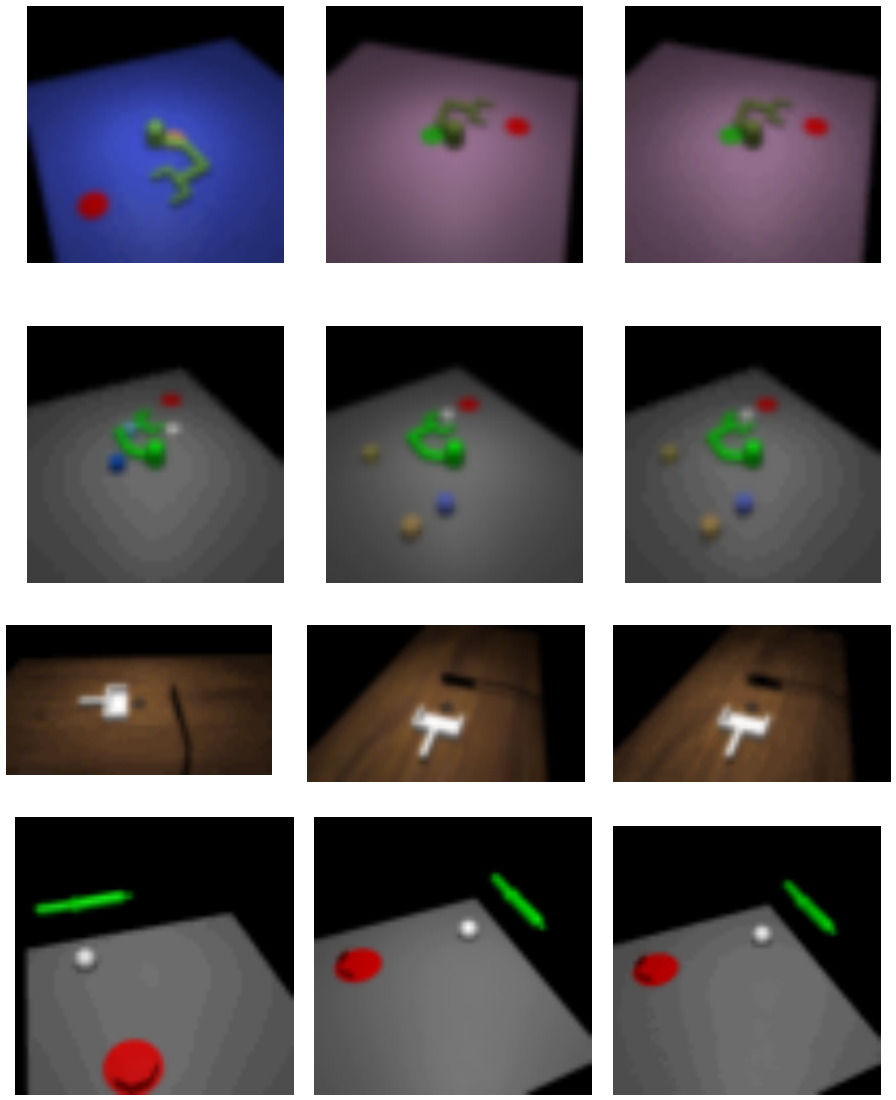
Context Translation Video Prediction Results

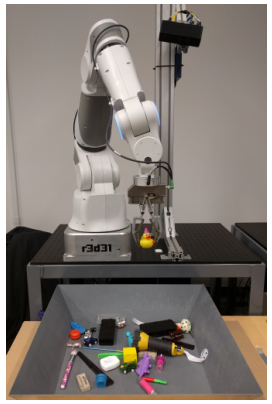
Source Video

Context at time 0

Translation Prediction

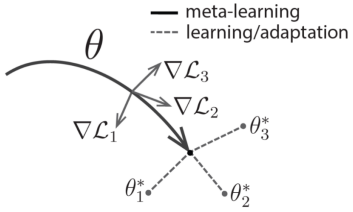
Context Translation Video Prediction Results



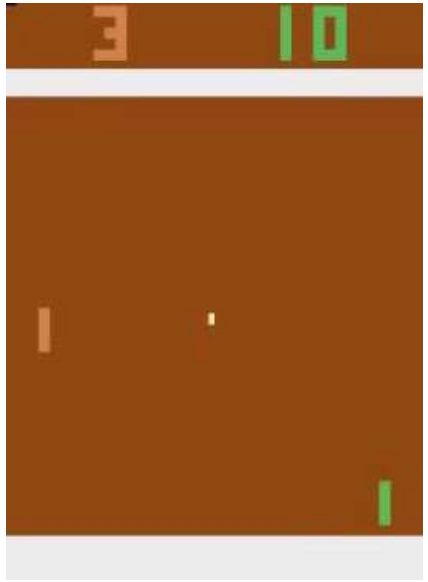


1. Can we self-supervise diverse real-world tasks?

2. Where can we get goal supervision?



3. How can new tasks build on prior tasks?



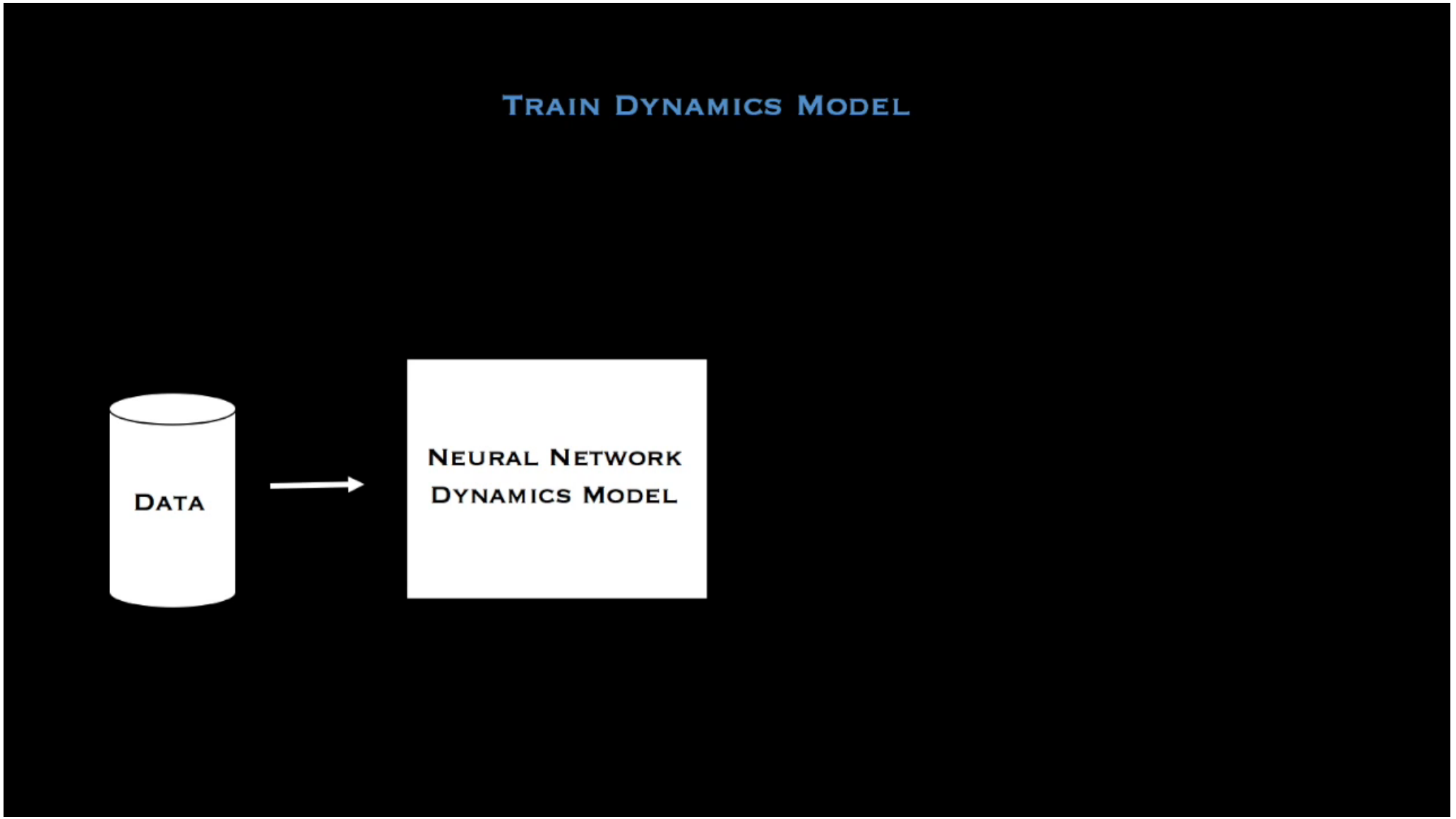
2,000,000 – 10,000,000 frames to learn Pong

How can humans learn so fast?

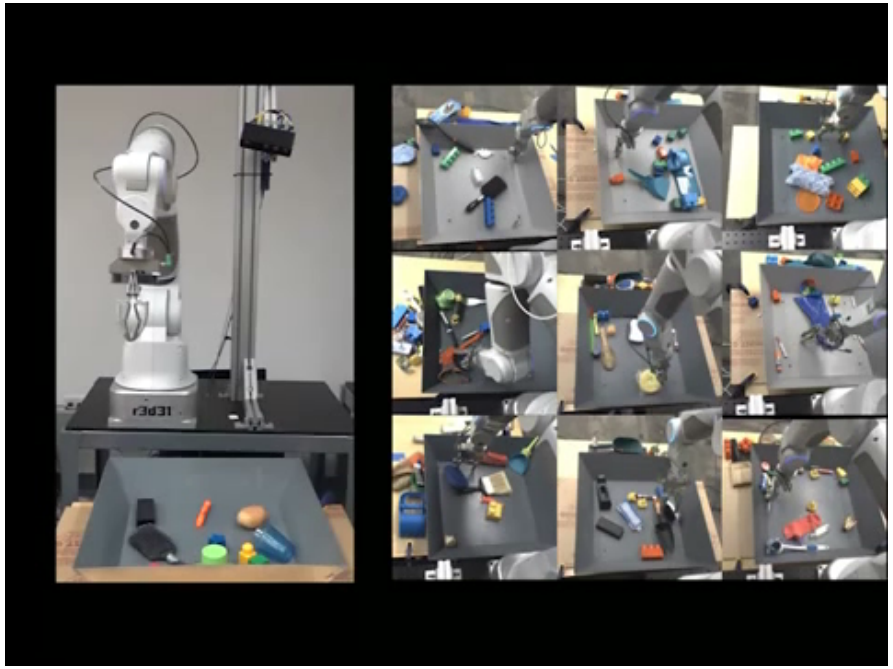
We don’t learn from scratch!

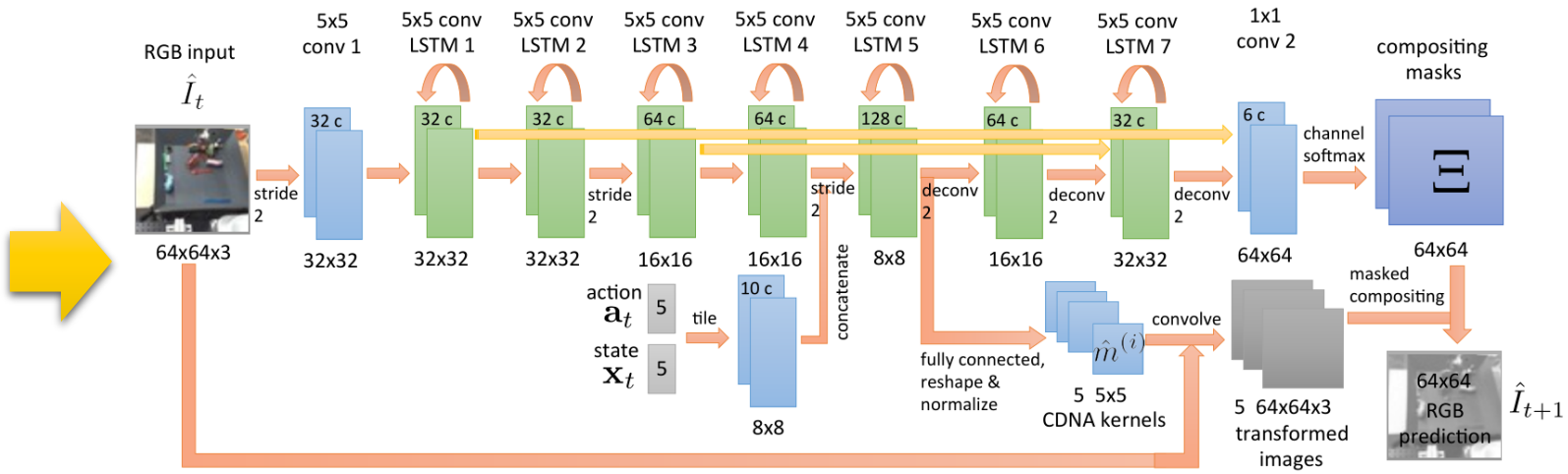
What can we reuse from past experience to learn new tasks?

Reusing models: model-based RL and prediction



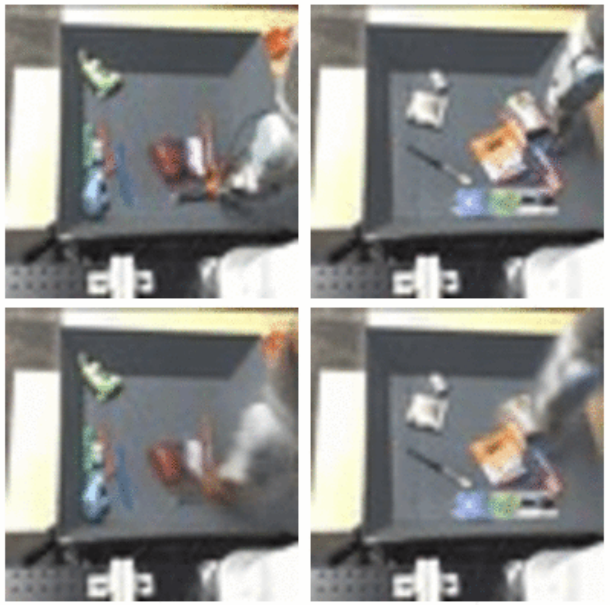
Learning to Reason about Physical Interactions

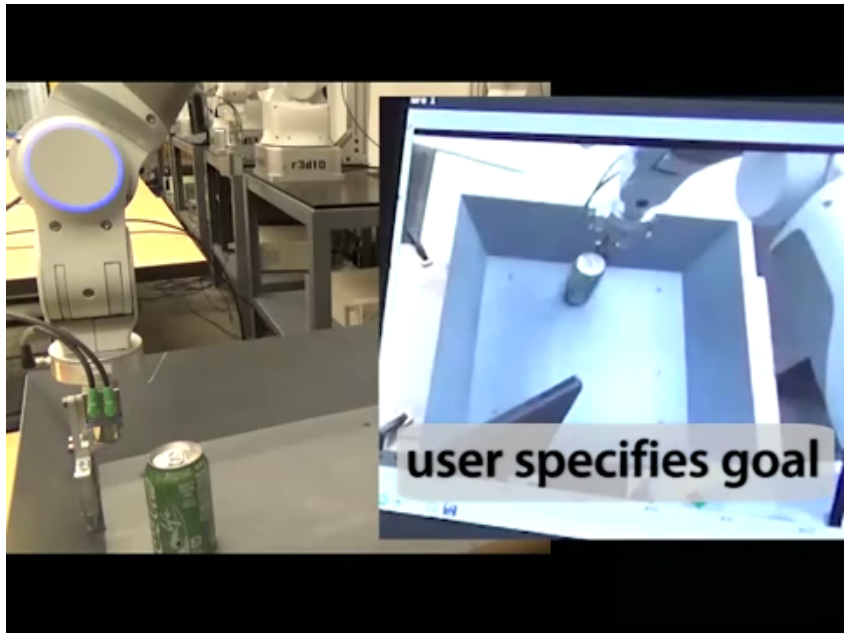




original video

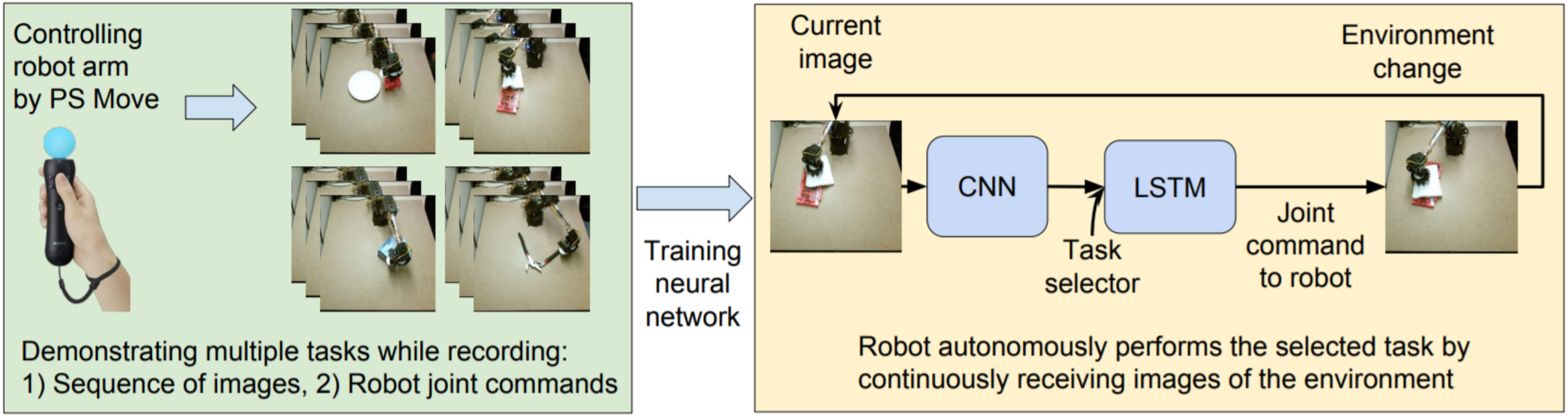
predictions



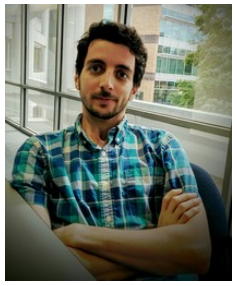


Chelsea Finn

Reusing skills with multi-task learning

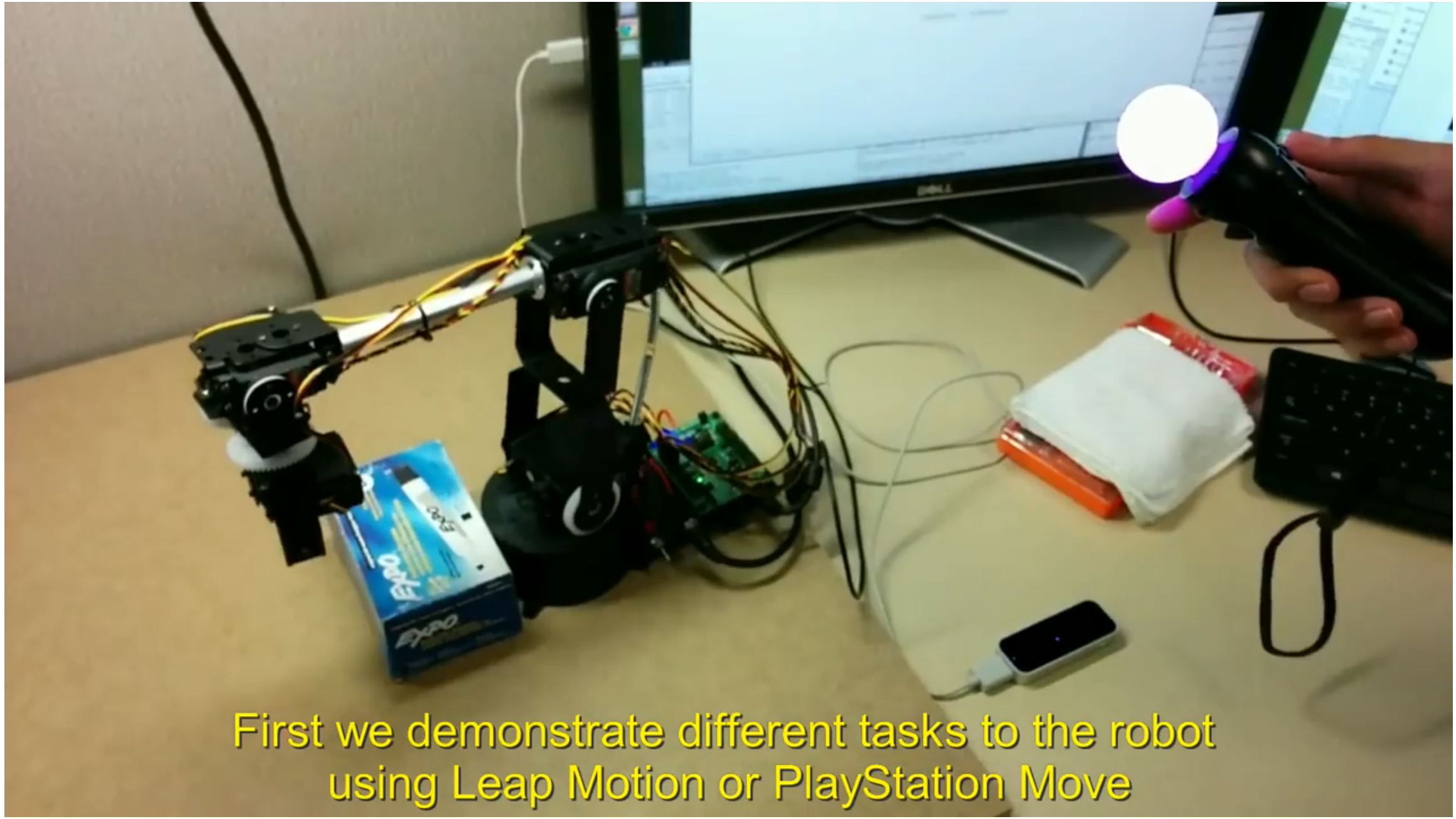


Rouhollah Rahmatizadeh



Pooya Abolghasemi

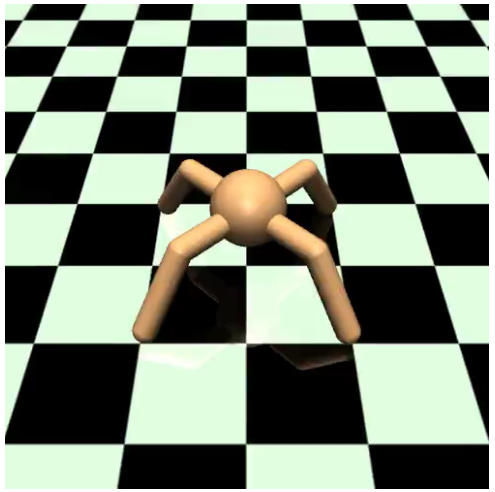
Rahmatizadeh, Abolghasemi, Bölöni, L. ‘17


First we demonstrate different tasks to the robot
using Leap Motion or PlayStation Move

Method	Task 1	Task 2	Task 3	Task 4	Task 5
Single-task	36%	16%	44%	16%	8%
Multi-task	16%	20%	52%	64%	20%
Multi-task autoregressive (no reconstruction)	12%	72%	56%	48%	16%
Multi-task autoregressive	**76%**	**80%**	**88%**	**76%**	**88%**

Reusing prior *learning* experience: meta-learning

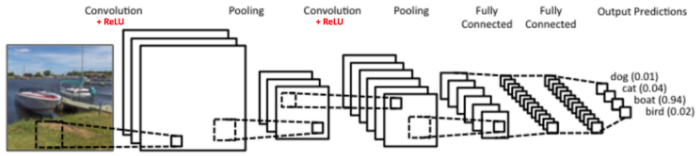
- Large-scale
- Emphasizes diversity
- Evaluated on generalization

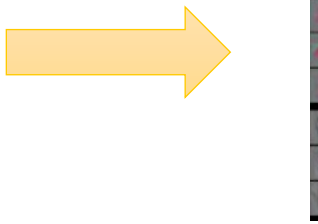


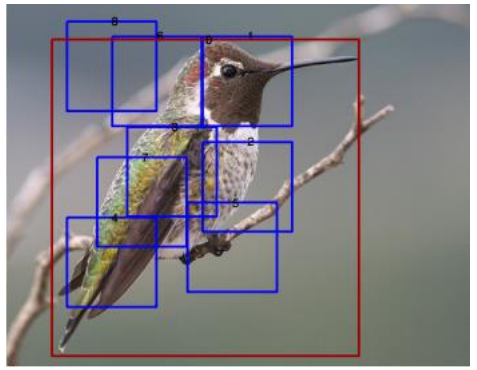
- Small-scale
- Emphasizes mastery
- Evaluated on performance
- Can we force generalization?

Learning useful representations with deep learning

IMAGENET

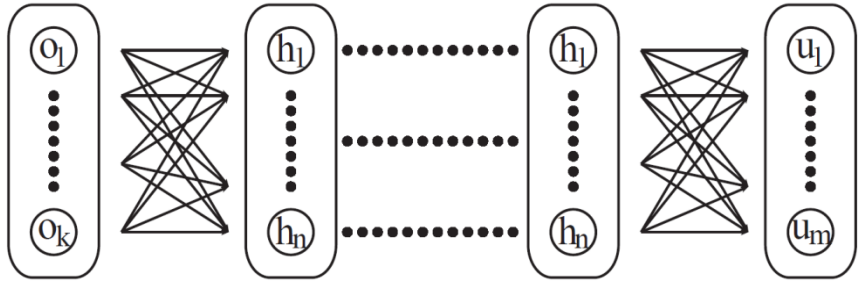






Where are the “ImageNet” features of decision making?

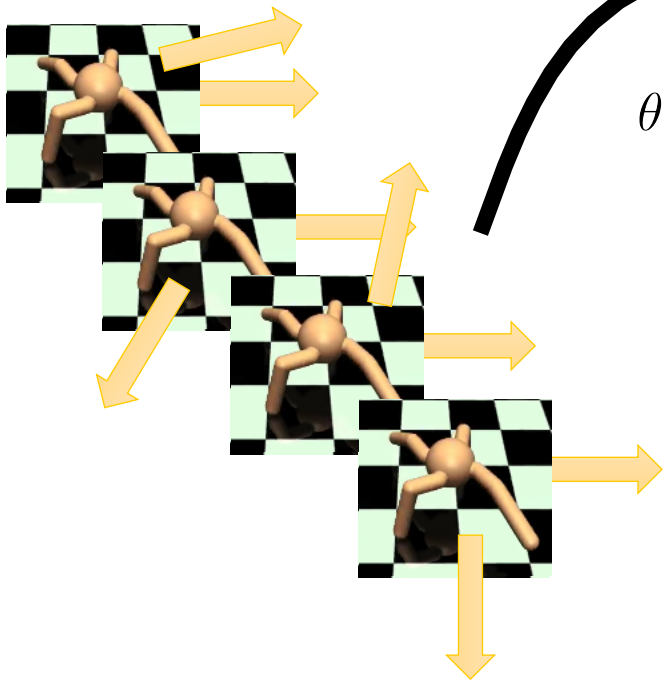
Multi-task training for adaptability

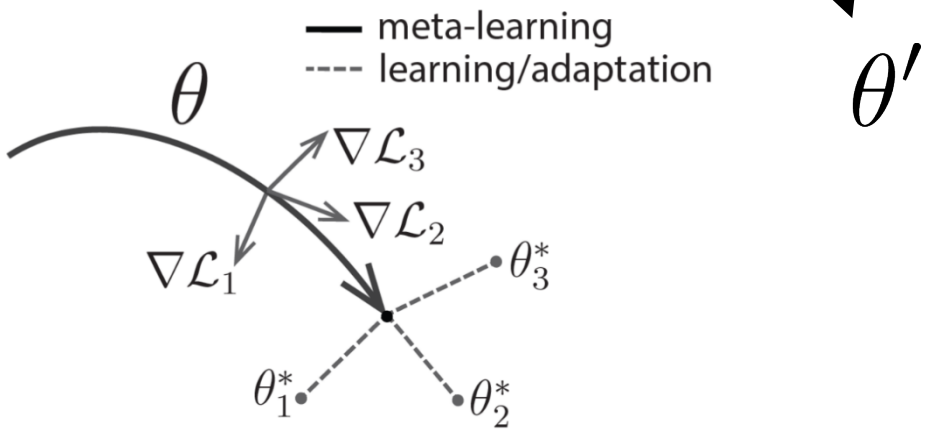


θ

$\theta \leftarrow \theta + \alpha \nabla_\theta R(\theta)$

$$\theta \leftarrow \theta + \alpha \sum_i \nabla_\theta [\theta + \alpha \nabla_\theta R_i(\theta)]$$



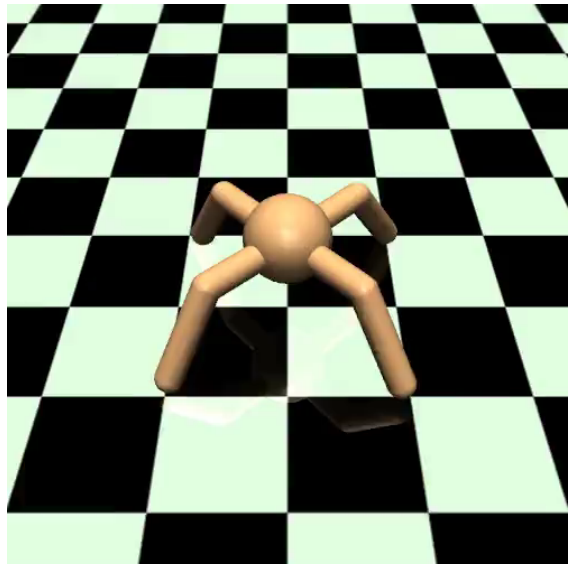


θ'

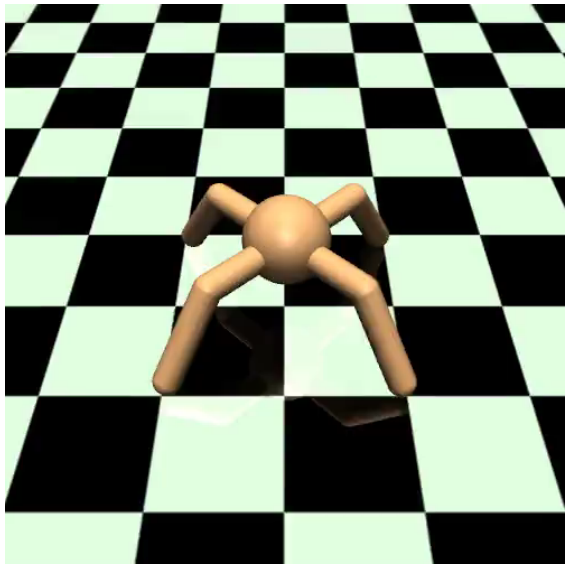
Chelsea Finn

Model-agnostic meta-learning: forward/backward locomotion

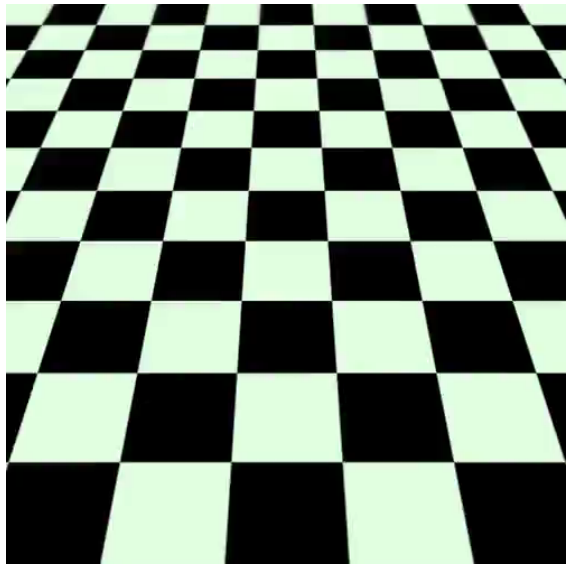
after MAML training

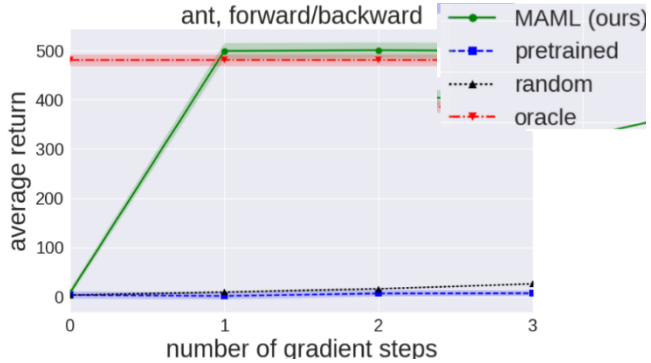


after 1 gradient step (forward reward)



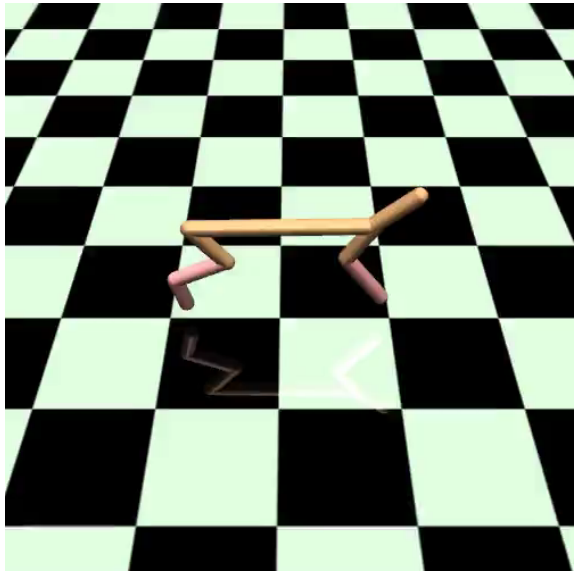
after 1 gradient step (backward reward)



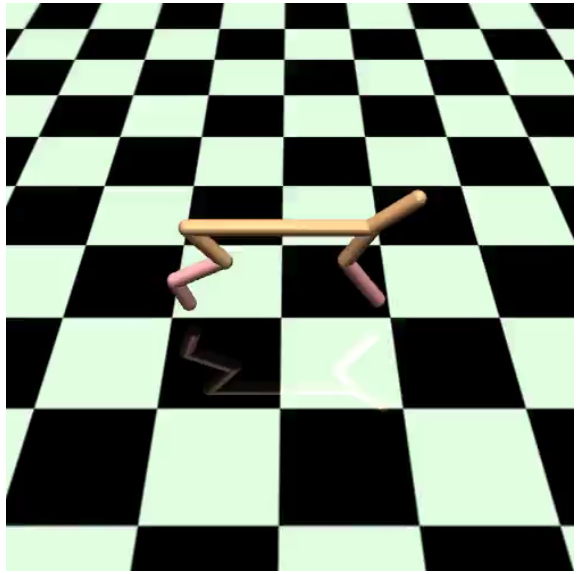


Model-agnostic meta-learning: forward/backward locomotion

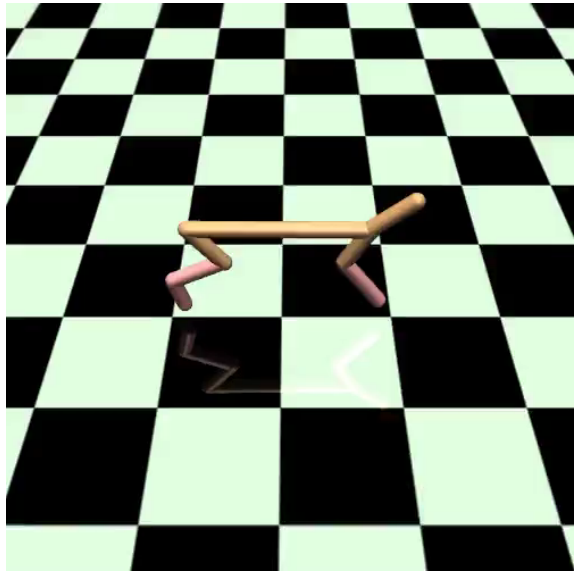
after MAML training



after 1 gradient step (backward reward)



after 1 gradient step (forward reward)

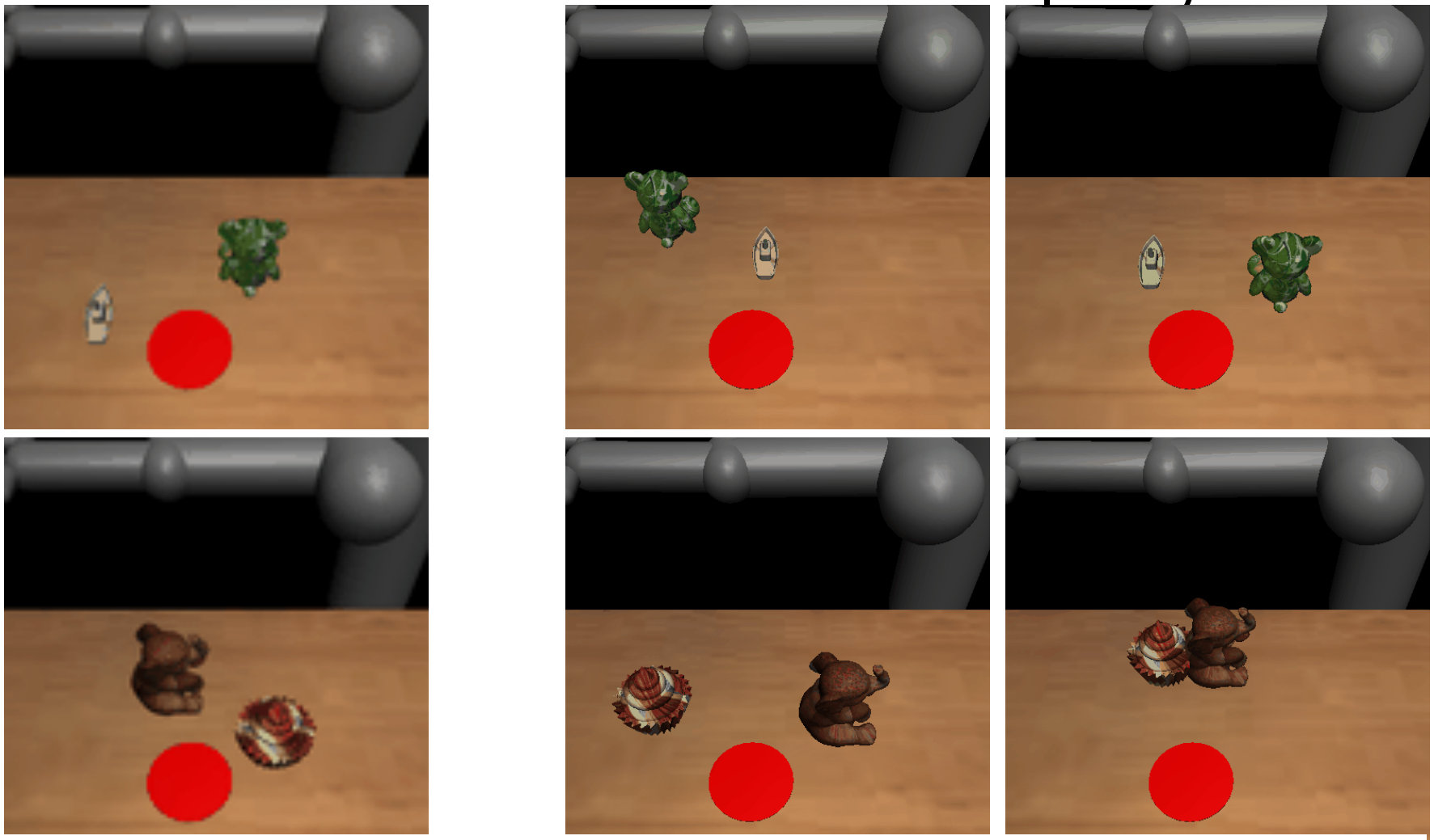


One-shot *visual* imitation

input demonstration

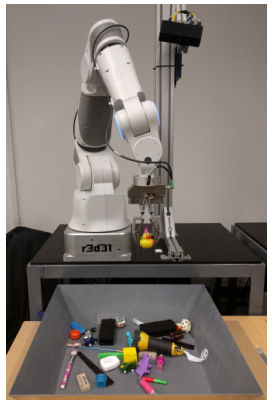
learned policy

115 random objects
with random textures,
masses, frictions, etc.



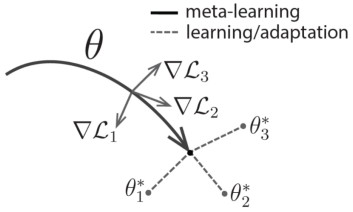
Takeaway: reuse experience across objects when learning to interact with new objects

Finn*, Yu*, Zhang, Abbeel, Levine ‘17



1. Can we self-supervise diverse real-world tasks?

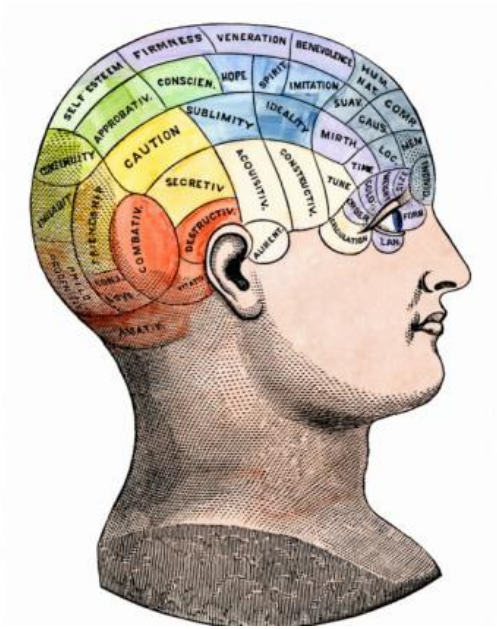
2. Where can we get goal supervision?

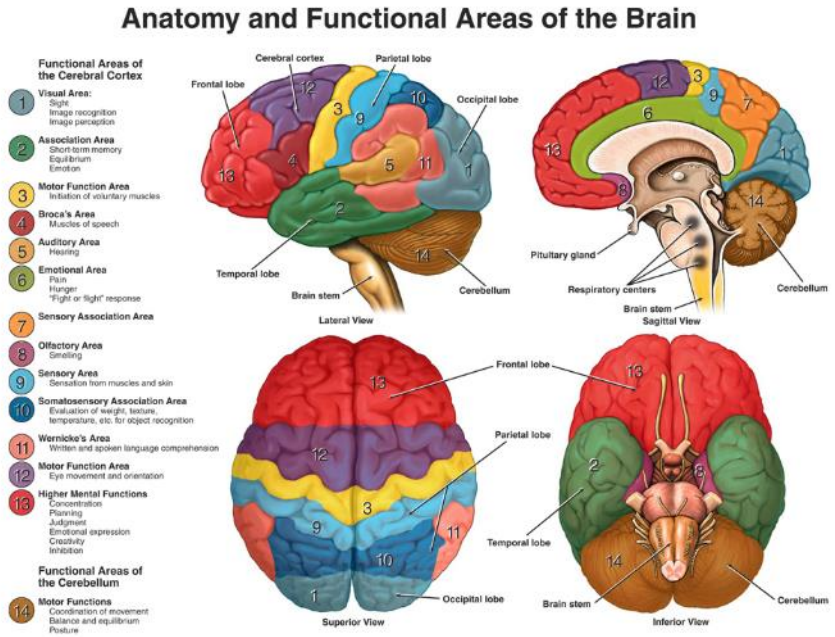


3. How can new tasks build on prior tasks?

How do we building intelligent machines?

- Imagine you have to build an intelligent machine, where do you start?



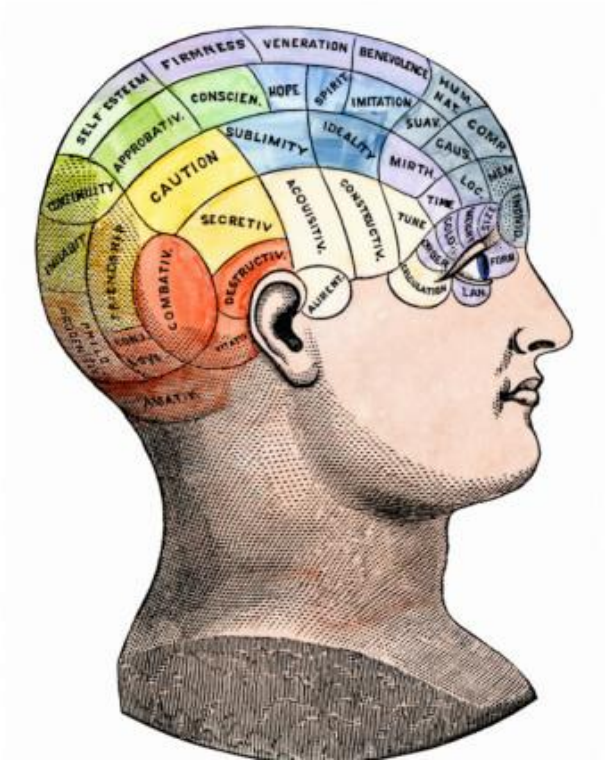


Learning as the basis of intelligence

- Some things we can all do (e.g. walking)
- Some things we can only learn (e.g. driving a car)
- We can learn a huge variety of things, including very difficult things
- Therefore our learning mechanism(s) are likely powerful enough to do everything we associate with intelligence
  - Though it may still be very convenient to “hard-code” a few really important things

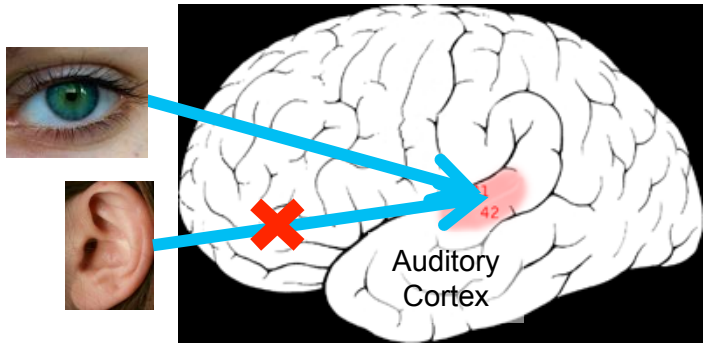
A single algorithm?

- An algorithm for each “module”?
- Or a single flexible algorithm?



Seeing with your tongue

Human echolocation (sonar)



[BrainPort; Martinez et al; Roe et al.]

adapted from A. Ng

What must that single algorithm do?

- Interpret rich sensory inputs

- Choose complex actions

Why deep reinforcement learning?

- Reinforcement learning = can reason about decision making
- Deep models = allows RL algorithms to learn and represent complex input-output mappings

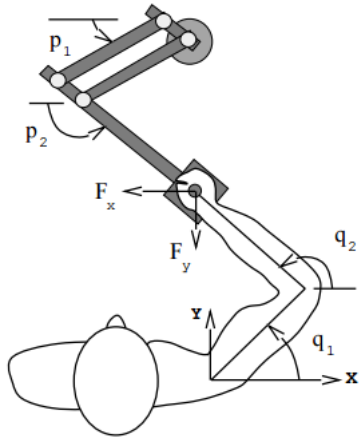
Deep models are what allow reinforcement learning algorithms to solve complex problems end to end!

What can deep learning & RL do well now?

- Acquire high degree of proficiency in domains governed by simple, known rules
- Learn simple skills with raw sensory inputs, given enough experience
- Learn from imitating enough human-provided expert behavior

What has proven challenging so far?

- Humans can learn incredibly quickly
  - Deep RL methods are usually slow
- Humans can reuse past knowledge
  - Transfer learning in deep RL is an open problem
- Not clear what the reward function should be
- Not clear what the role of prediction should be



What is missing?

How Much Information Does the Machine Need to Predict?

Y LeCun

- "Pure" Reinforcement Learning (cherry)
  - The machine predicts a scalar reward given once in a while.
  - **A few bits for some samples**
- Supervised Learning (icing)
  - The machine predicts a category or a few numbers for each input
  - Predicting human-supplied data
  - **10→10,000 bits per sample**
- Unsupervised/Predictive Learning (cake)
  - The machine predicts any part of its input for any observed part.
  - Predicts future frames in videos
  - **Millions of bits per sample**

- (Yes, I know, this picture is slightly offensive to RL folks. But I'll make it up)

Where does the supervision come from?

- Yann LeCun's cake
  - Unsupervised or self-supervised learning
  - Model learning (predict the future)
  - Generative modeling of the world
  - Lots to do even before you accomplish your goal!
- Imitation & understanding other agents
  - We are social animals, and we have culture – for a reason!
- The giant value backup
  - All it takes is one +1
- All of the above

How should we answer these questions?

- Pick the right problems!
- Pay attention to generative models, prediction
- Carefully understand the relationship between RL and other ML fields