# Autonomous Robotics

## Robot Learning from Demonstration

Josiah Hanna
University of Wisconsin — Madison

Slide credit: Sonia Chernova

# Announcements

Final project released today.

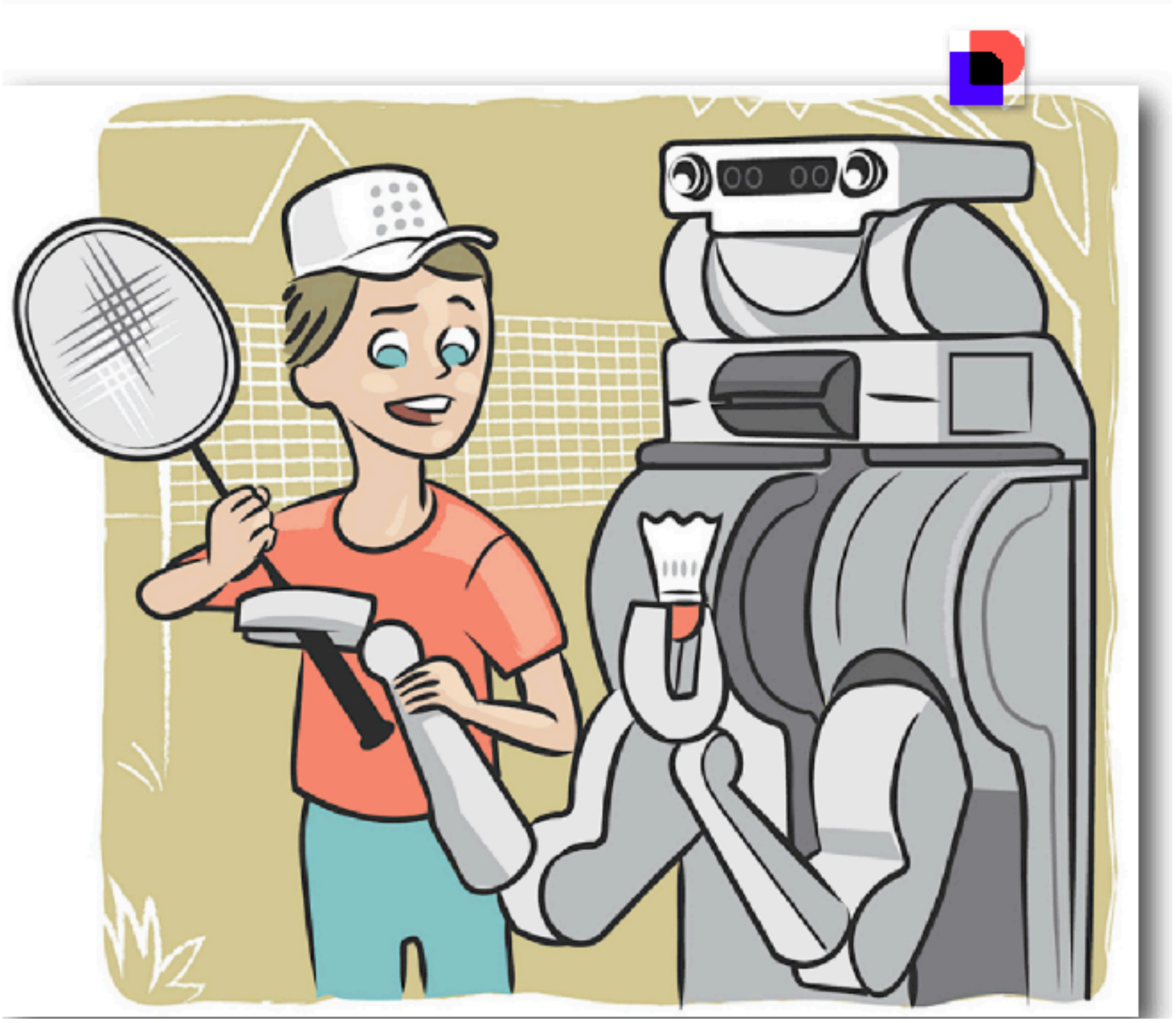
# Learning Outcomes

After today's lecture, you will:

- Understand the imitation learning problem setting.
- Be able to describe key challenges that arise when teaching a robot to perform a task.

# Motivation

- No need for robot experts.
- Natural way to program robot skills.
- Add new skills on the fly.
- We have lots of data recording people doing things!



# Terminology

- Imitation learning (IL)
- Behavior cloning (BC)
- Learning from Demonstration (LfD)
- Programming by Demonstration (PbD)
- Mimicry

# Formalism

- Basic formalism:
  - Given a dataset of the form $\{(s_i, a_i)\}_{i=1}^{m}$.
  - Goal: learn a policy $\pi$ such that $\pi(s_i) \approx a_i$.
- Intuition: copy the behavior in the dataset.
- There are many variations of this basic setup.
  - Example: BC from Observation: given *state-only* trajectories $\{(s_1, \ldots s_T)\}$, learn a policy that reproduces these trajectories. Why is this hard?

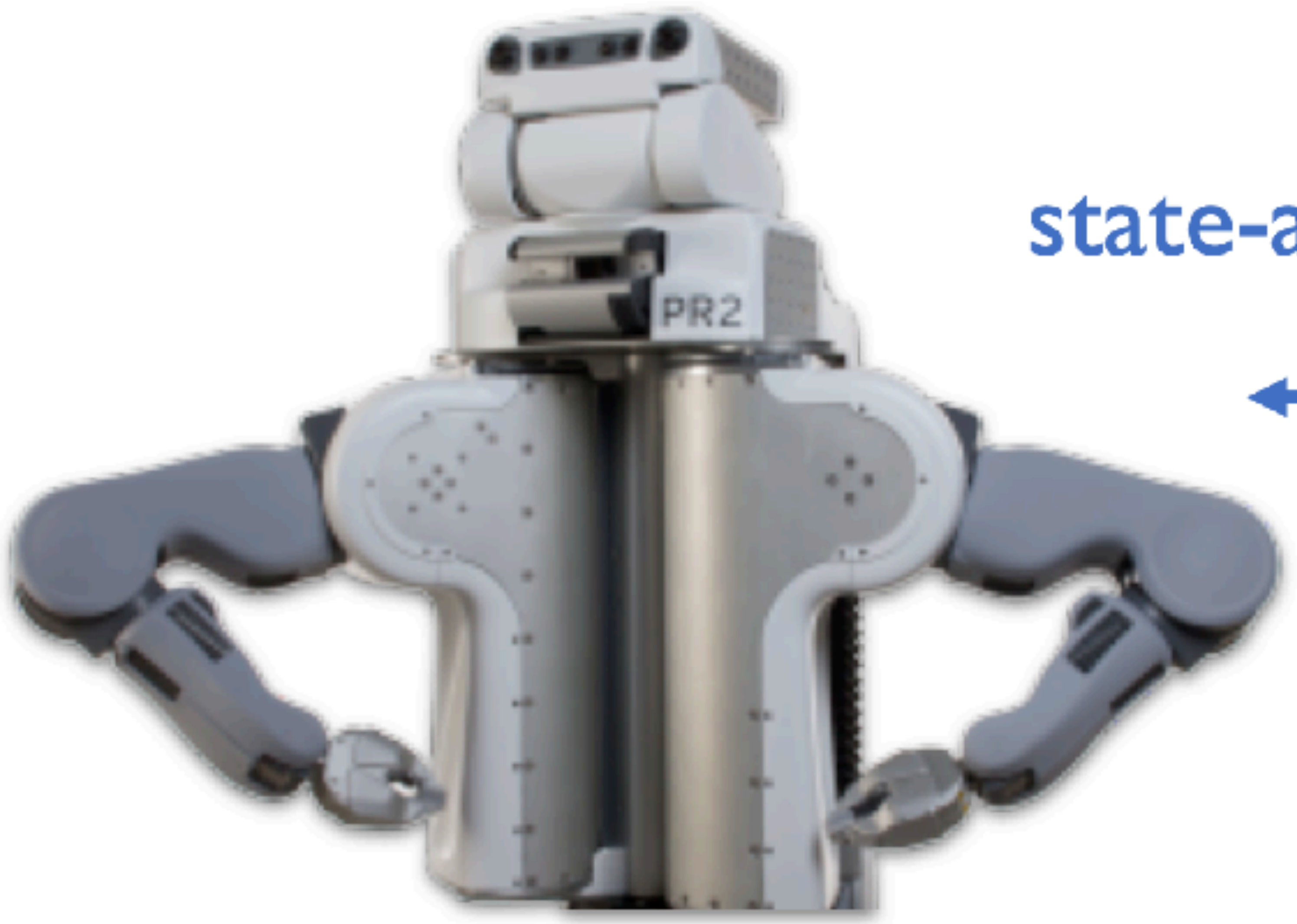
# Basic Approach

- Imitation learning is a *supervised* learning problem.
  - States are inputs, demonstrator's actions are labels.
  - Can be classification or regression depending on control space.
- One issue: many supervised learning algorithms assume inputs are independent of each other during training and testing. Problem?
  - Small mistakes by the learner may compound over time.

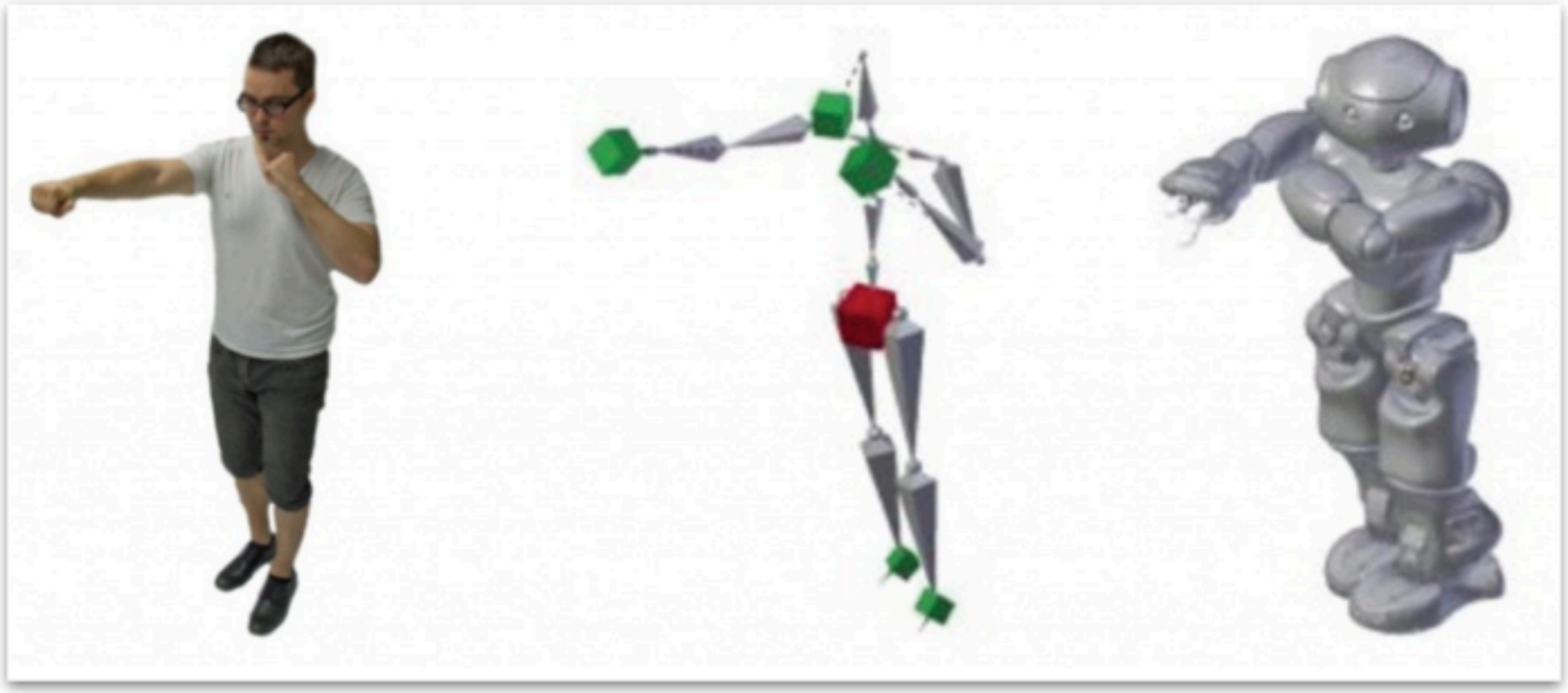
# Solution: The DAgger Approach

1. First, collect demonstrations from the expert.
2. Repeat:
    1. Use supervised learning to imitate the demonstrator $\rightarrow \pi_i$
    2. Collect more demonstrations but sometimes use $\pi_i$ to take actions.
        - Still, record expert actions in every state visited.
    3. Supervised learning on the new dataset $\rightarrow \pi_{i+1}$
3. At each step, increase the proportion of states where $\pi_i$ takes the action instead of the expert.

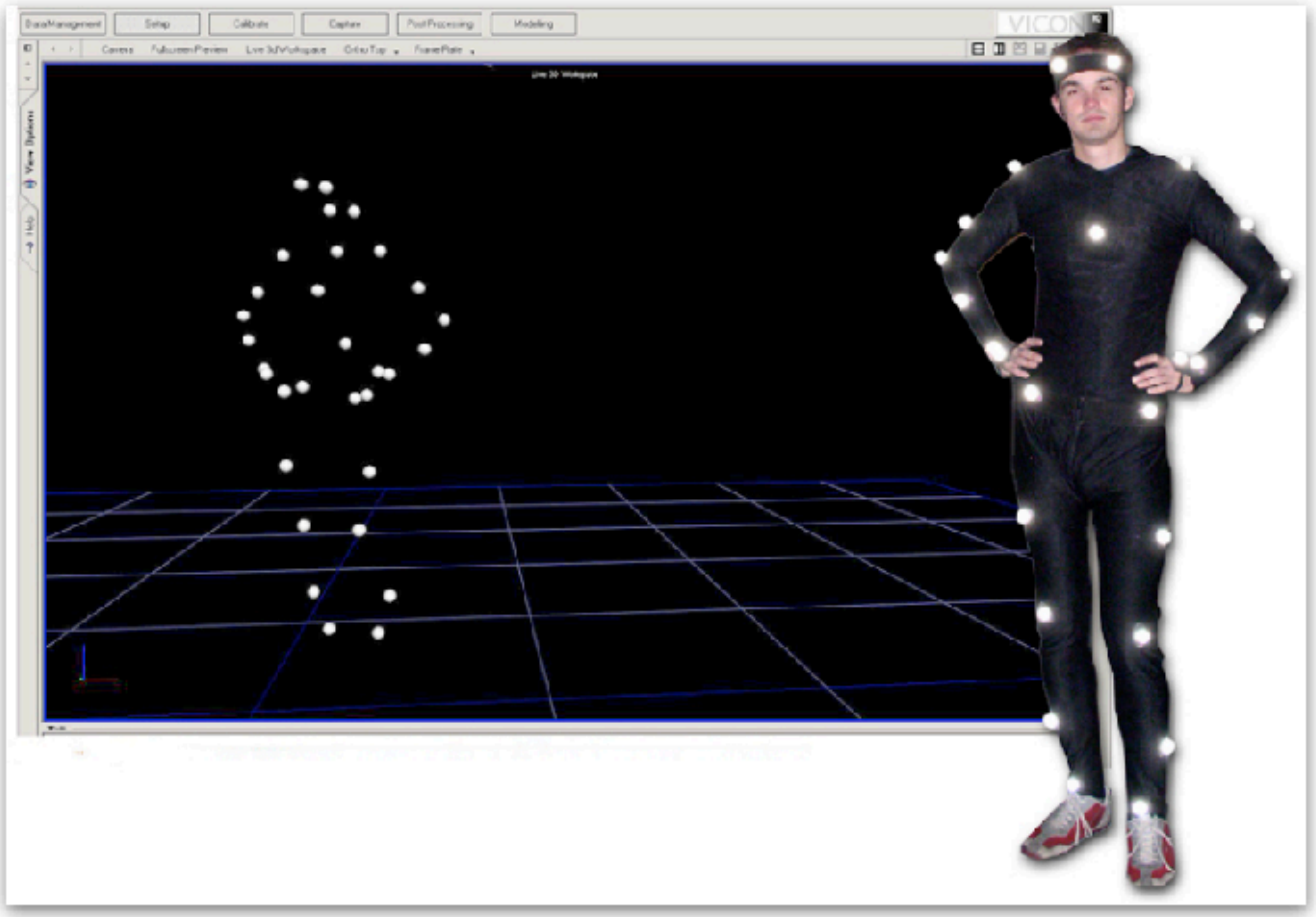
# Correspondence Problem



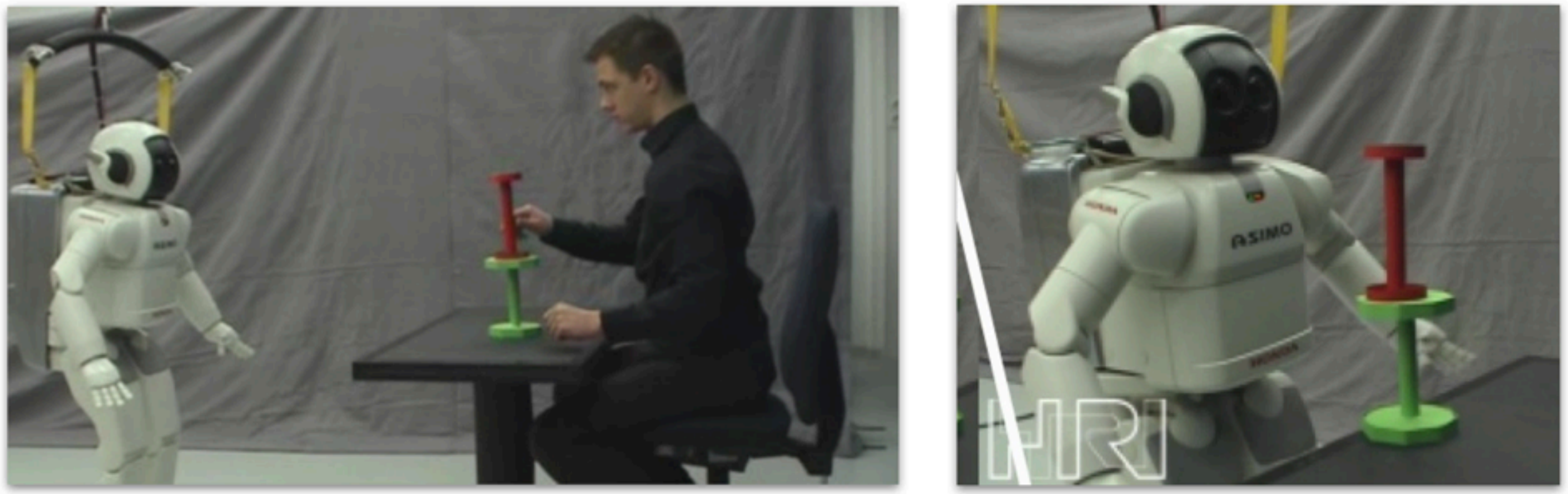

state-action mapping?



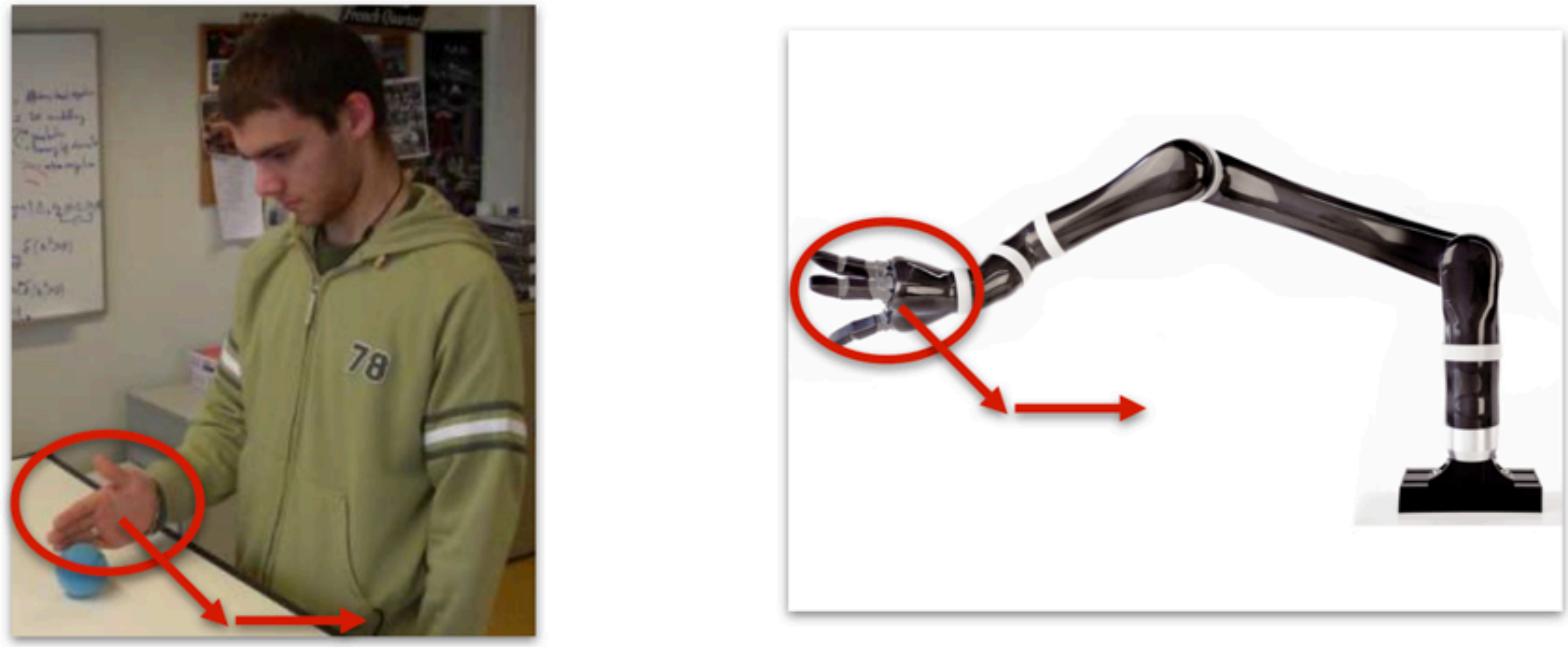
# Motion Capture



# Mapping between Humans and Robots



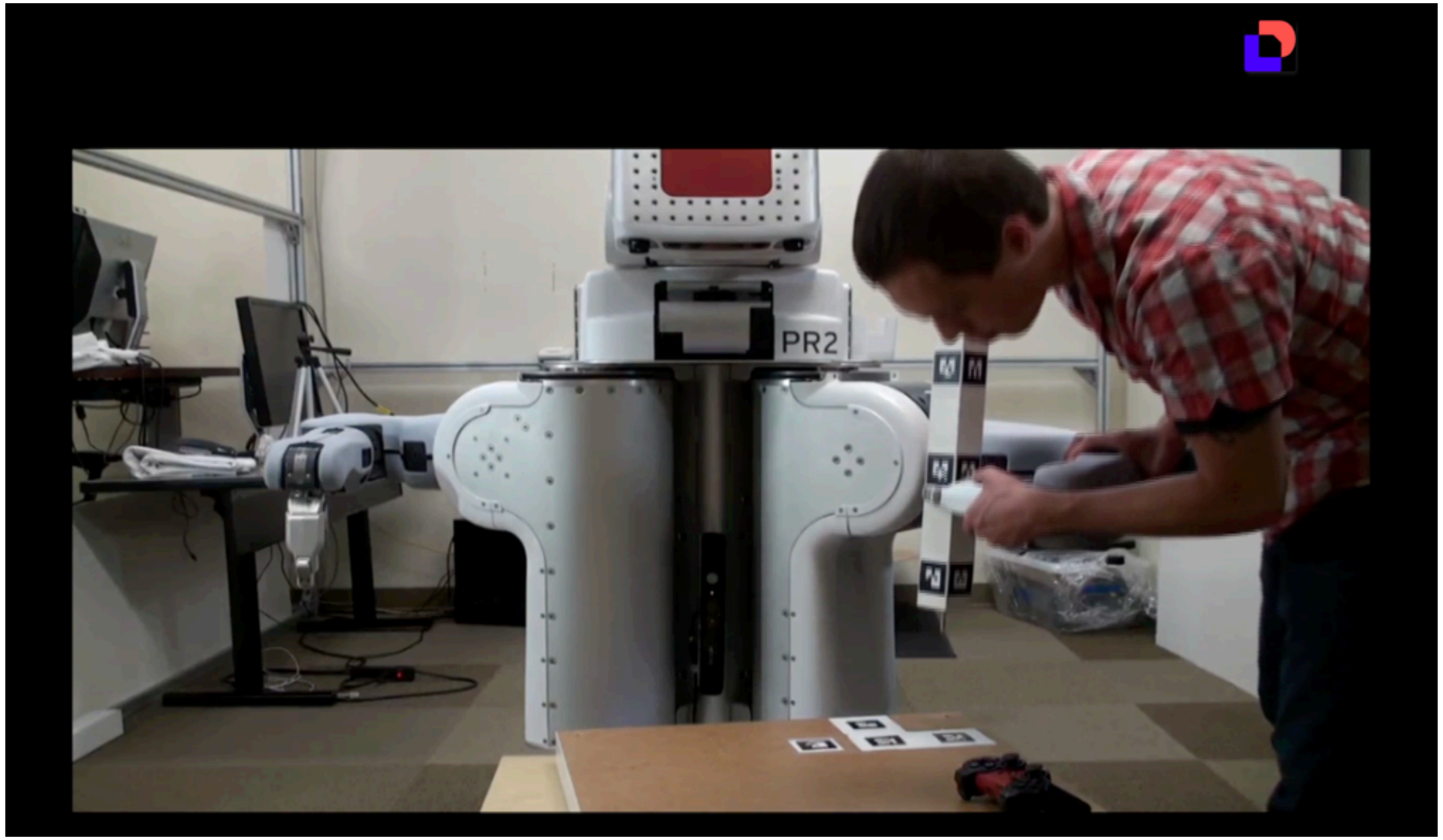

Object-based



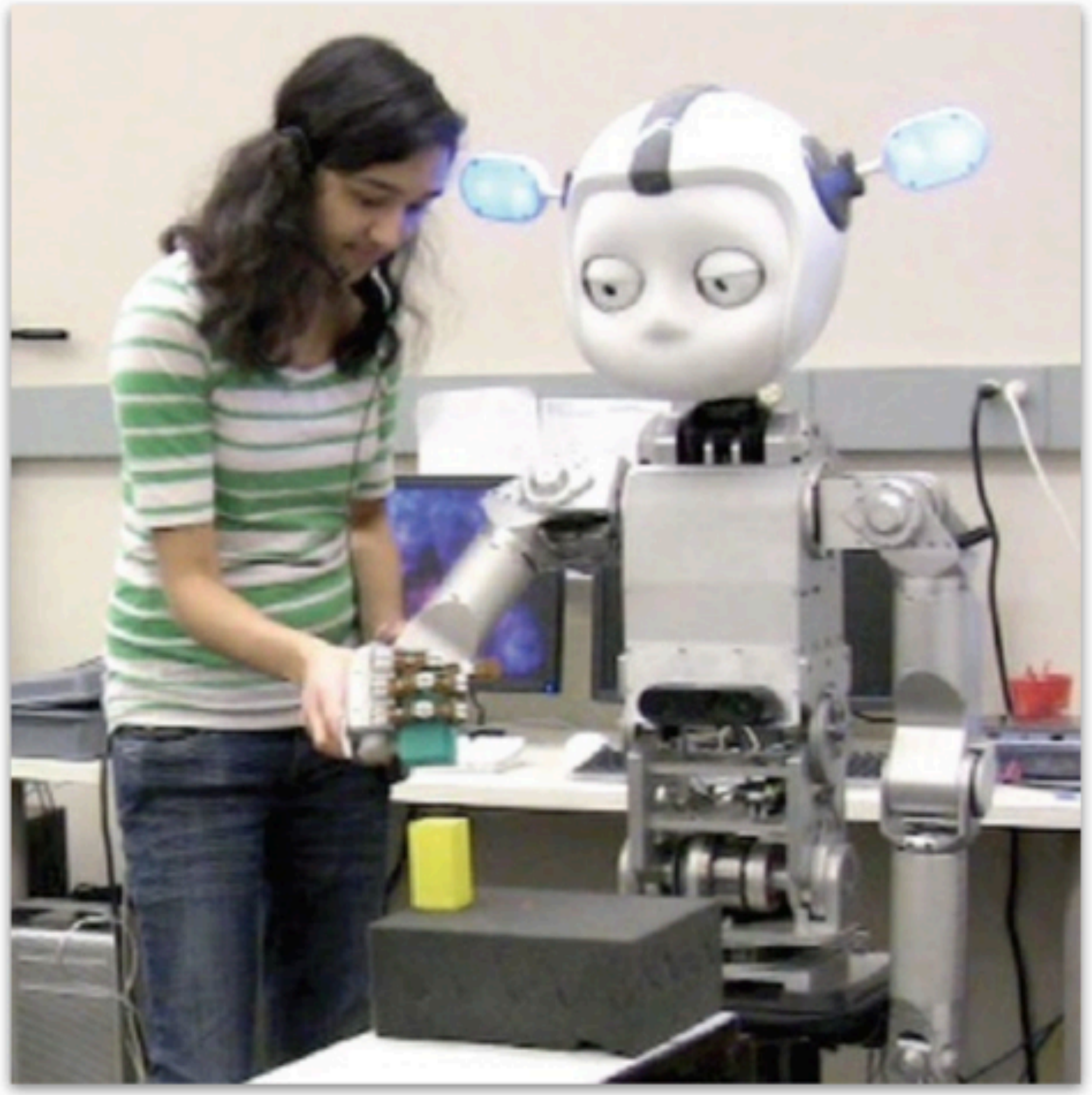
End effector-based

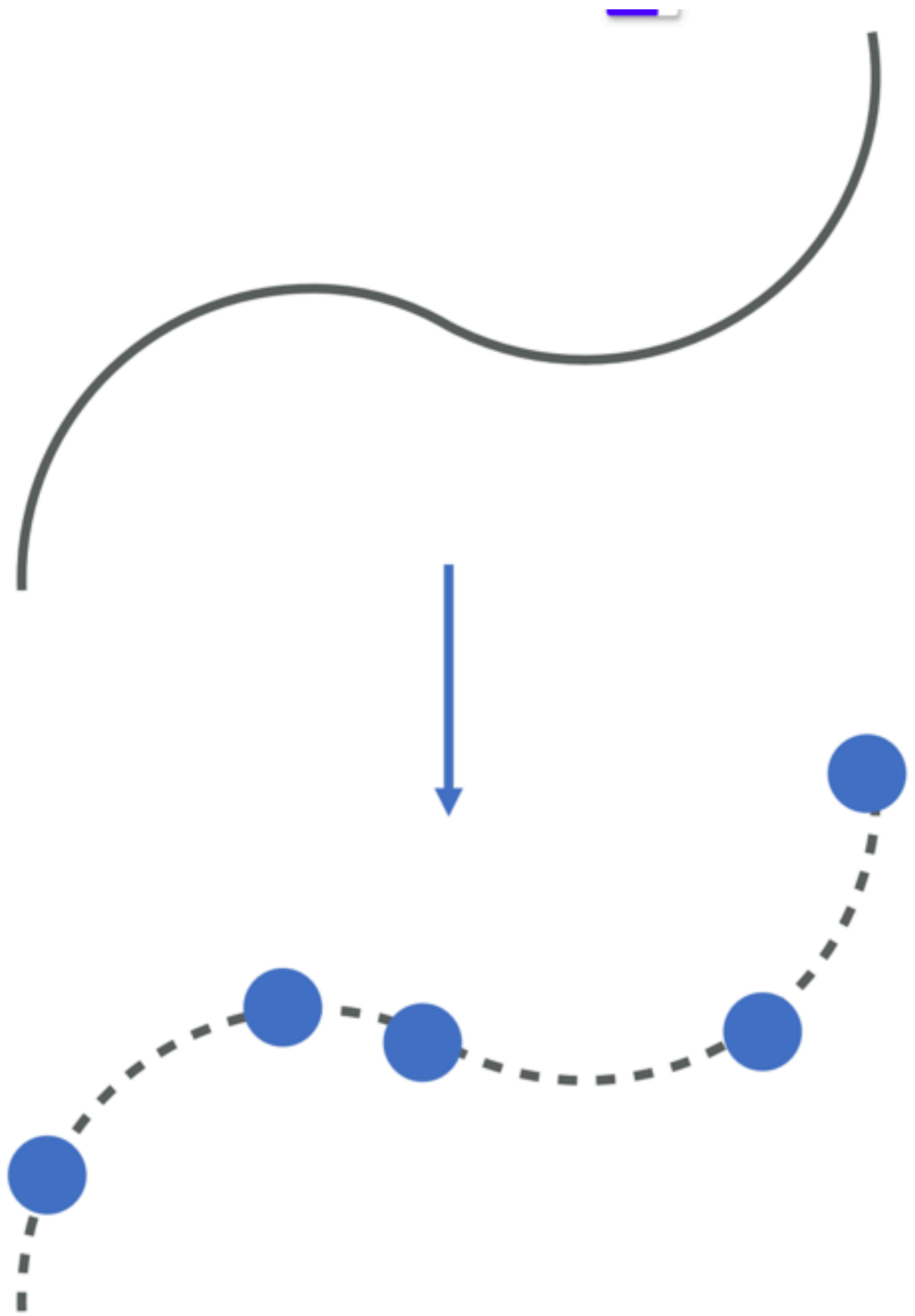
# Teleoperation

# Kinesthetic Teaching



# Keyframe Demonstrations

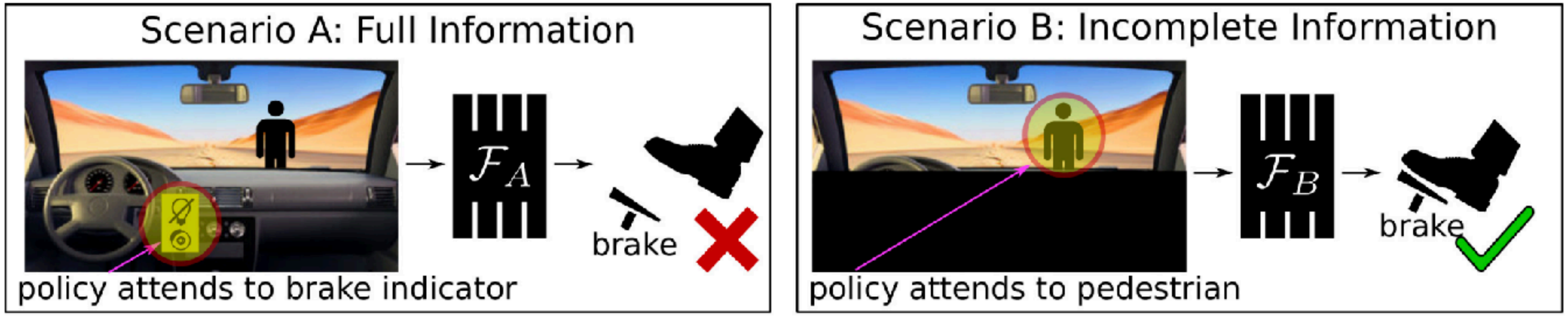




# Causal Confusion

- The demonstrator is running some policy that depends on some state variables that the observer can observe in the world.

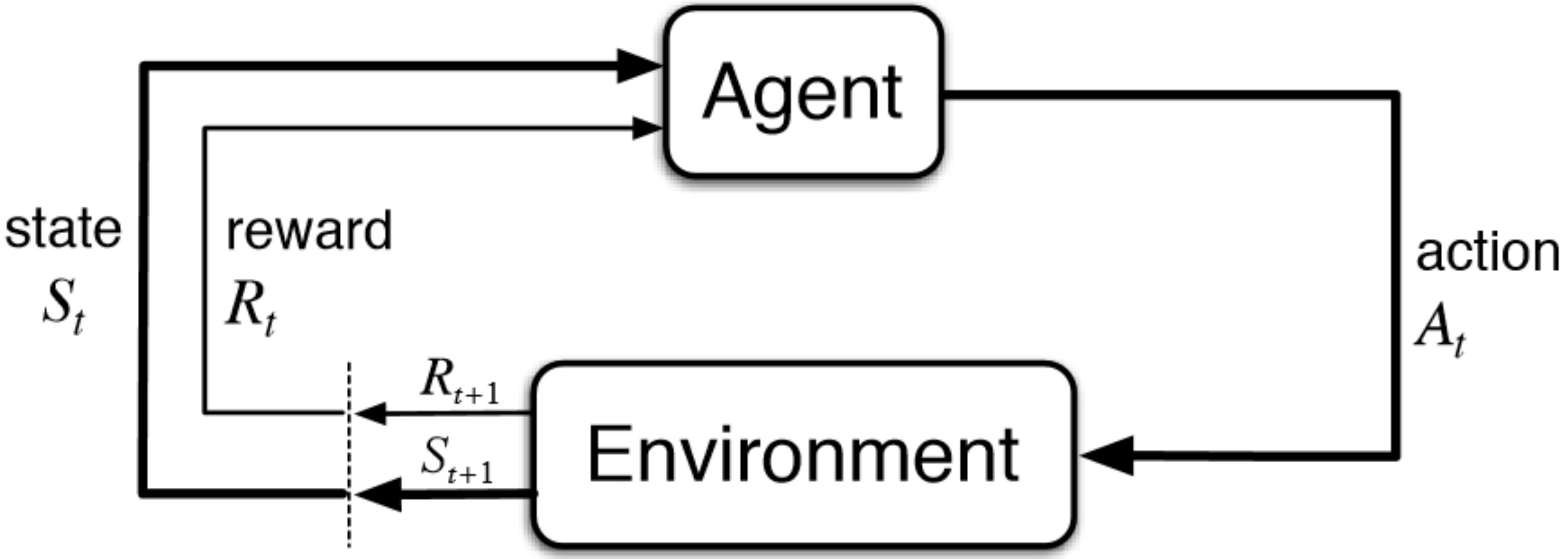
- Depending on its sensors, the robot can observe / estimate some set of variables that may be different than the demonstrators.



# Inverse Reinforcement Learning

- Basic formalism:
  - Given a dataset of the form $\{(s_i, a_i)\}_{i=1}^m$.
  - Goal: learn a reward function such that $\pi^\star(s) \approx a$, where $\pi^\star$ is the optimal policy for the learned reward function.
- Intuition: learn demonstrator's intention rather than simply mimicking their actions. Why useful?
  - Potentially generalizes better to new scenarios.
  - Can attempt to surpass the demonstrator.

# Inverse Reinforcement Learning




$$\ldots S_t, A_t, S_{t+1}, A_{t+1}, \ldots$$

$$S_{t+1} \sim p(\cdot \mid S_t, A_t)$$

$$A_{t+1} \leftarrow \pi(S_{t+1})$$

# Inverse Reinforcement Learning

Helicopter tricks
[Abbeel et al. 2007]

# Summary

Today we covered:

1. The imitation learning problem, issues, and approaches.

2. Introduced and discussed inverse reinforcement learning.

# Action Items

Human-robot interaction readings posted.

Begin final project.