UC San Diego

Behavioral Cloning from Observation

Presenter: Qiaojun Feng
05/26/2020

Prerequisite

- Markov decision processes:

- $M = \{S, A, T, r, \gamma\}$
- S: state
- A: action
- T: the transitioning probability from state s_i to s_{i+1} after action a
- r: $S \times A \rightarrow R$ the immediate reward for taking a specific action a in state s
- γ: discount factor

• Markov decision processes:

• $M = \{S, A, T, r, \gamma\}$
• S: state
• A: action
• T: the transitioning probability from state s_i to s_{i+1} after action a
• r: $S \times A \rightarrow R$ the immediate reward for taking a specific action a in state s
• γ: discount factor

Introduction

- Imitation Learning:
- **Goal:** learn by trying to imitate another expert
- **Given data:** observations of other agent (demonstrations) consist of state-action pairs

$$\{(s_0,a_0),(s_1,a_1),\ \ldots,(s_N,a_N)\}$$

Introduction

- Imitation Learning:
- **Goal:** learn by trying to imitate another agent
- **Given data:** observations of other agent (demonstrations) consist of state-action pairs

$$\{(s_0, a_0), (s_1, a_1), \ldots, (s_N, a_N)\}$$

- **Problem:** large amounts of demonstration data ***do not*** come with **actions** (e.g., YouTube videos)

Introduction

- Imitation from Observation (this paper)
- **Goal**: learn a task given *state-only* demonstrations
- **Given data:** observations consist of only *state*

$$\{s_0, s_1, \ldots, s_N\}$$

Introduction

- What's the difference between state s v.s. action a?
- Why one is hard to get than the other?

state: joints position (qpos in hw)
action: joints force (qf in hw)
We know its structure well.

Much complicated! Model unknown

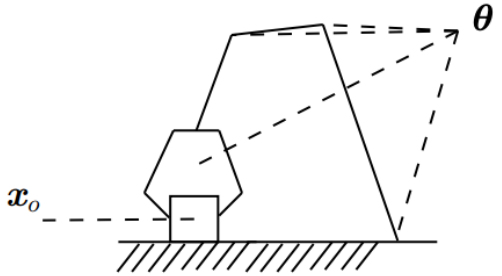
Introduction

- Policy
  $\pi : S \rightarrow A$, which specifies the action the agent should execute in one state.

- State following a policy
  $T_\pi = \{(s_i, s_{i+1})\}$ the set of state transitions experienced by an agent following a policy π.

Introduction

- Agent-specific state
  a given state s can be partitioned into an agent-specific state s^a, and a task-specific state s^t.
  $s = (s^a, s^t)$



- Agent-specific inverse dynamics model
  $M_\theta: S^a \times S^a \rightarrow p(A)$ that maps a pair of agent-specific state transitions, $(s_t^a, s_{t+1}^a) \in T_\pi^a$, to a distribution of agent actions that is likely to cause that transition

Related Works

- Behavioral cloning
  Given: $\{(s_0,a_0),(s_1,a_1),\ ...,(s_N,a_N)\}$
  Target: Learn the expert policy $\pi: S \rightarrow A$

- Inverse reinforcement learning
  Given: $\{(s_0,a_0),(s_1,a_1),\ ...,(s_N,a_N)\}$
  Target: Learn the reward function r: $S \times A \rightarrow R$

- Model-based reinforcement learning
  Given: $\{(s_0,a_0),(s_1,a_1),\ ...,(s_N,a_N)\}$
  Target: Learn the transition model T: $S \times A \rightarrow S$

Method

imitation learning $\mathcal{D}_{\text{demo}} = ((s_0, a_0), (s_1, a_1), \ldots)$

imitation from observation $\mathcal{D}_{\text{demo}} = ((s_0, \ ? \ ), (s_1, \ ? \ ), \ldots)$

our solution:

(1) estimate actions using a learned ***inverse dynamics model***, i.e., guess the missing $\{a_0, a_1, \ldots, a_N\}$

(2) perform ***behavioral cloning*** using state-action pairs

Method

policy initialization: initialize π randomly

data collection: use π to gather experience

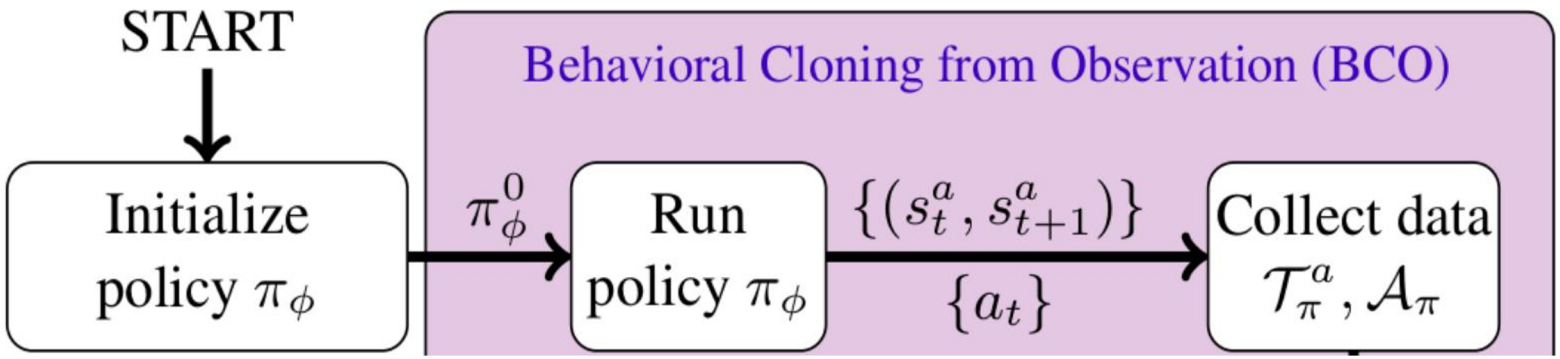
model learning: learn inverse dynamics model $P(a_t|s_t, s_{t+1})$

action estimation: infer demonstrator actions using model

policy learning: update π using *behavioral cloning* over state-(action estimate) pairs

Method

Before demonstrations



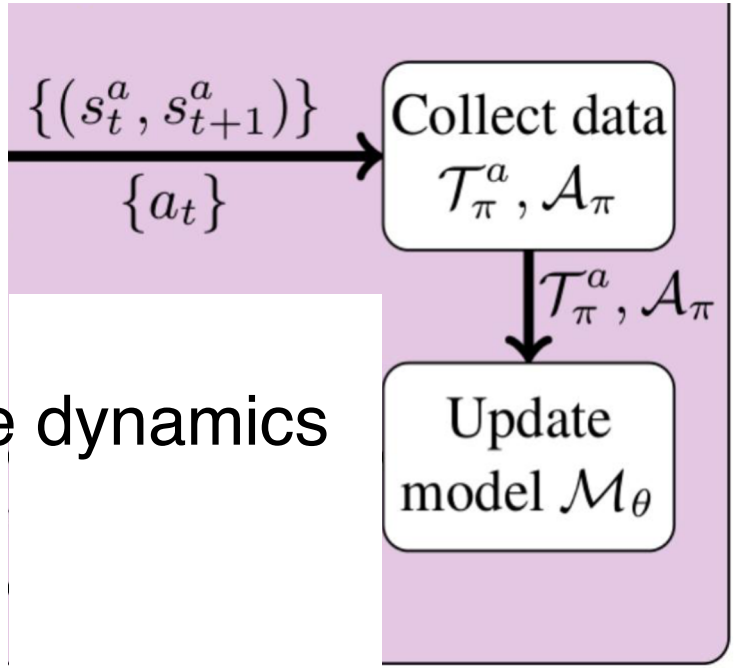
policy initialization: initialize with a random policy

data collection: gather experience of the agent (state-action pairs)

(Same agent but not necessarily performing the desired task. Assume known actions.)

Method

Before demonstrations

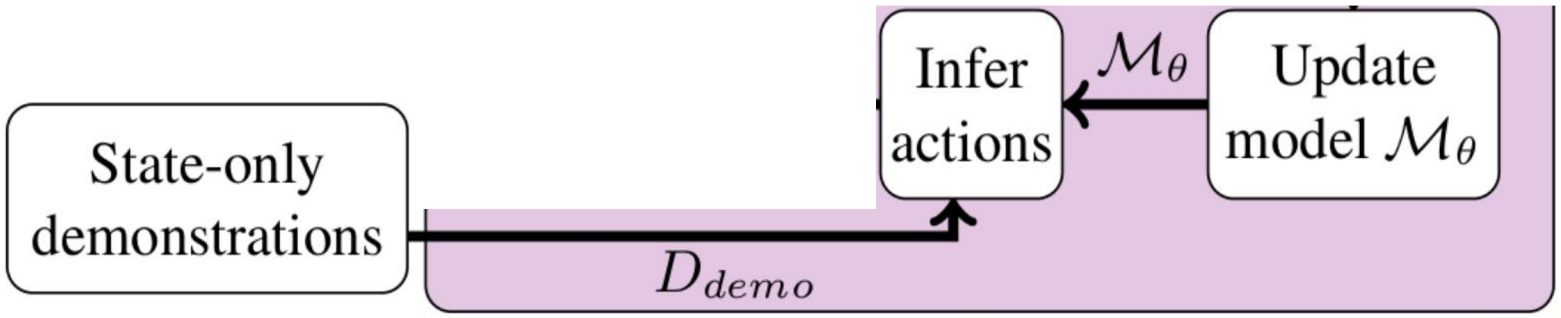


model learning: learn inverse dynamics model $P(a_t|s_t, s_{t+1})$

Method

After demonstrations

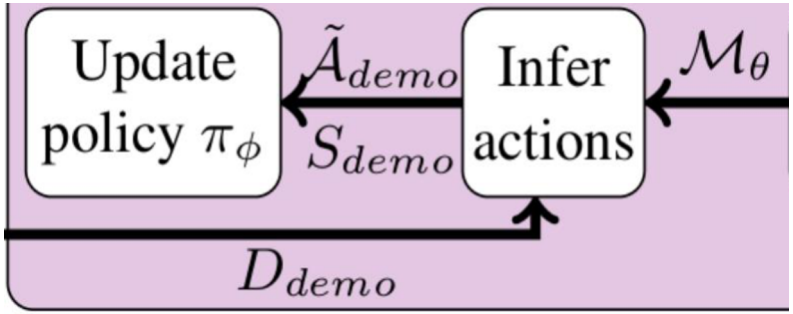
action estimation: infer demonstrator actions using model $\mathcal{M}_\theta(s_t, s_{t+1}) = \arg\max_a P(a|s_t, s_{t+1})$



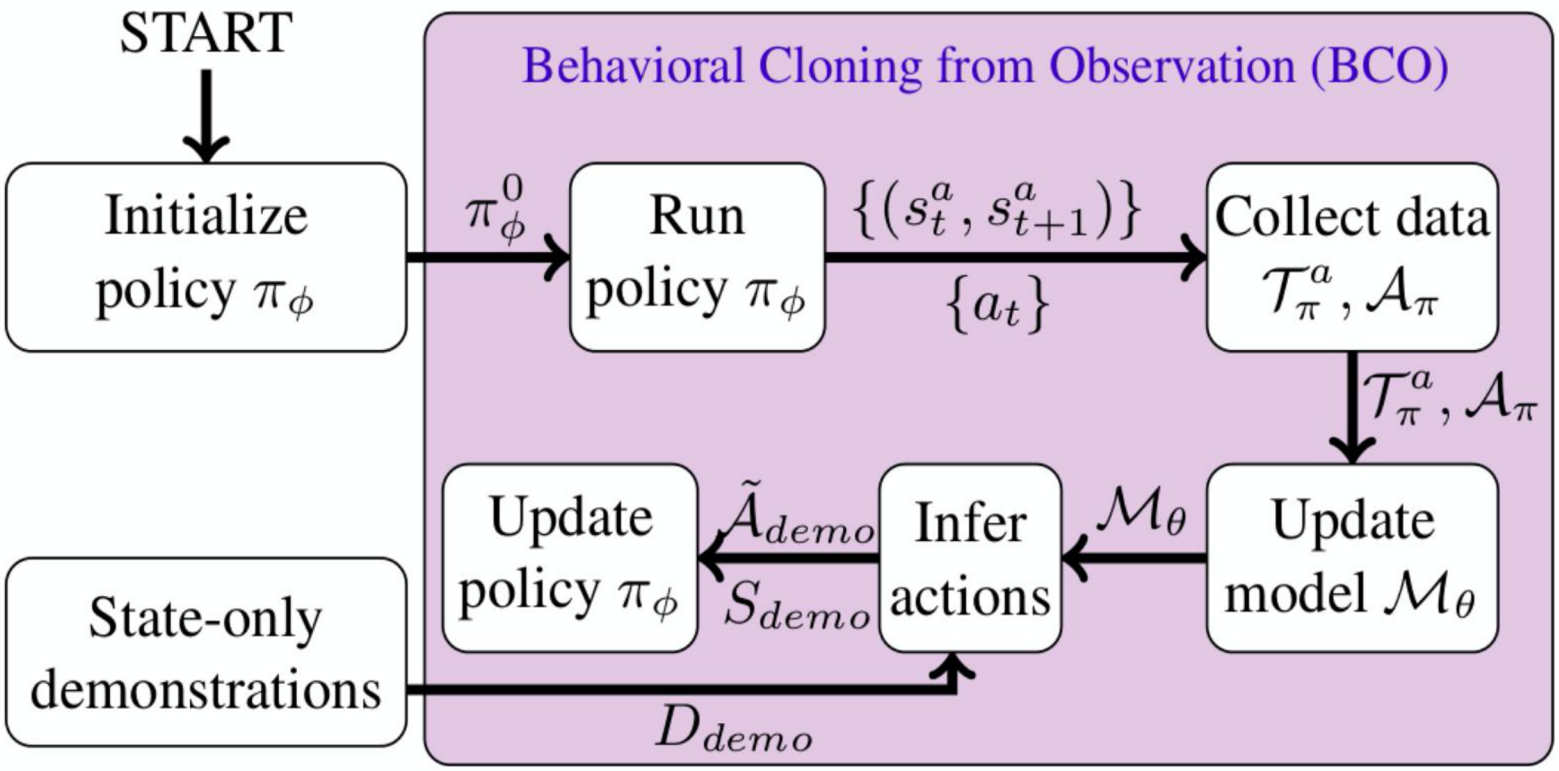
Method

After demonstrations

policy learning: update policy using *behavioral cloning* over state-(action estimate) pairs

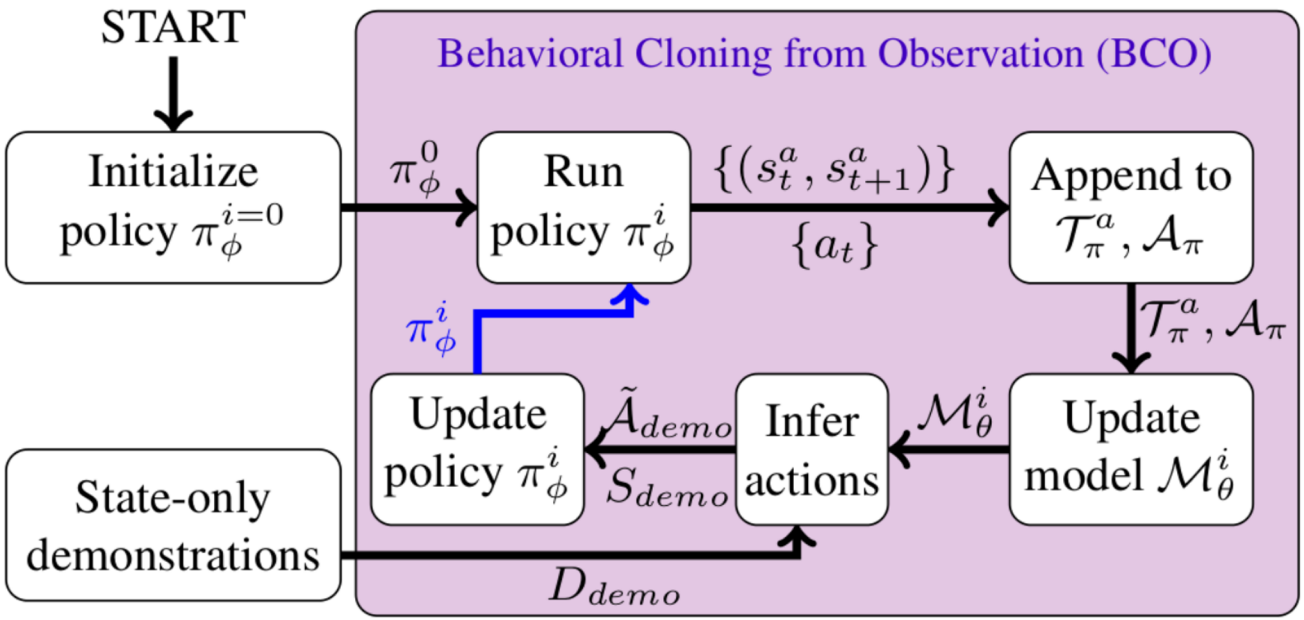


Method


START
Behavioral Cloning from Observation (BCO)
Initialize policy π_ϕ
π_ϕ^0
Run policy π_ϕ
$\{(s_t^a, s_{t+1}^a)\}$
$\{a_t\}$
Collect data $\mathcal{T}_\pi^a, \mathcal{A}_\pi$
$\mathcal{T}_\pi^a, \mathcal{A}_\pi$
Update model $\mathcal{M}_\theta$
$\mathcal{M}_\theta$
Infer actions
$\tilde{\mathcal{A}}_{demo}$
S_{demo}
Update policy π_ϕ
State-only demonstrations
D_{demo}

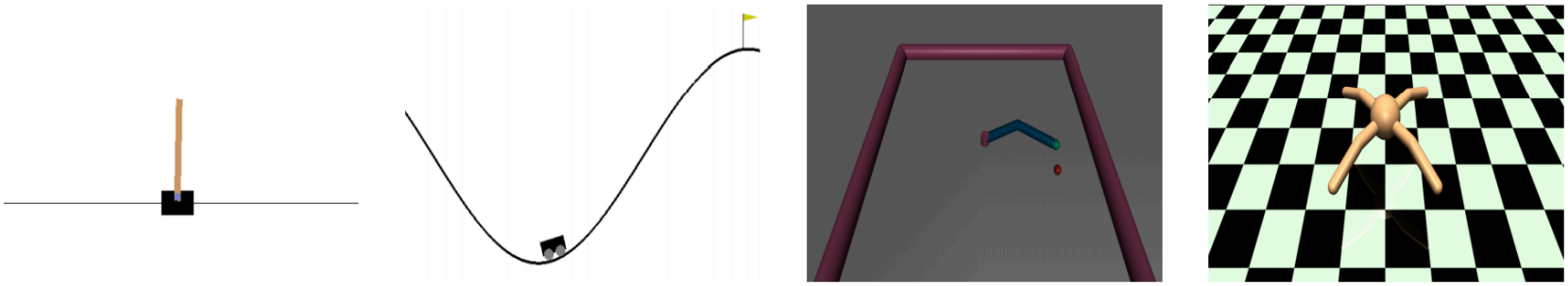
Method

Idea: learn a *better* model by gathering more experience



Generate more experience using latest policy.

Experiments

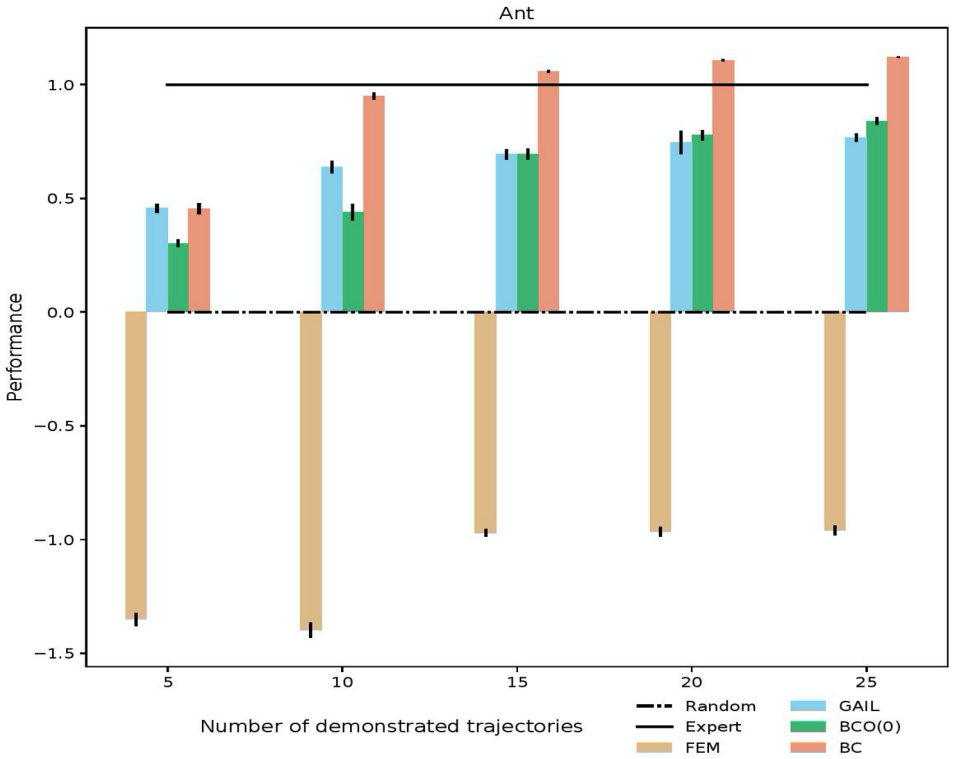


(a) CartPole (b) MountainCar (c) Reacher (d) Ant

- (a) discrete action space (left, right, still)
- (b) discrete action space (left, right, still)
- (c) continuous action space (force on two joints)
- (d) 8 dimensional continuous action space

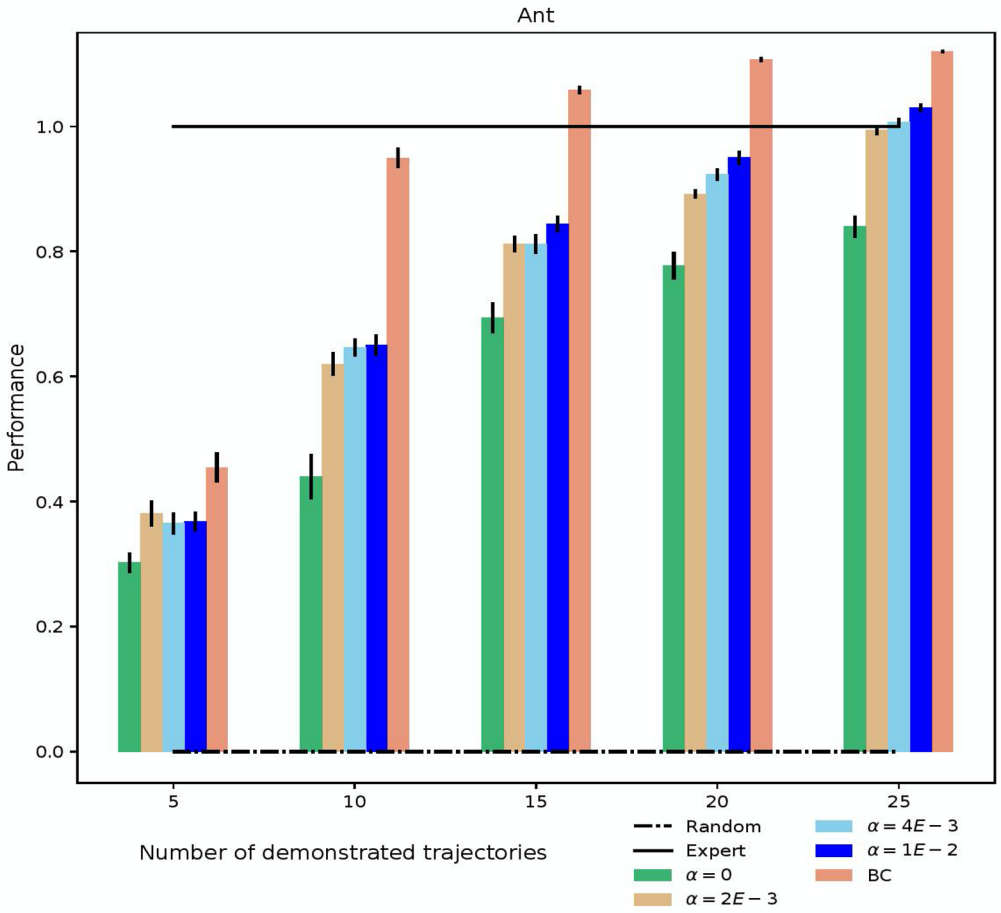
Experiments

BCO v. **imitation learning** (has actions) (Ant)



Experiments

Varying α (number of interactions after demonstrations)

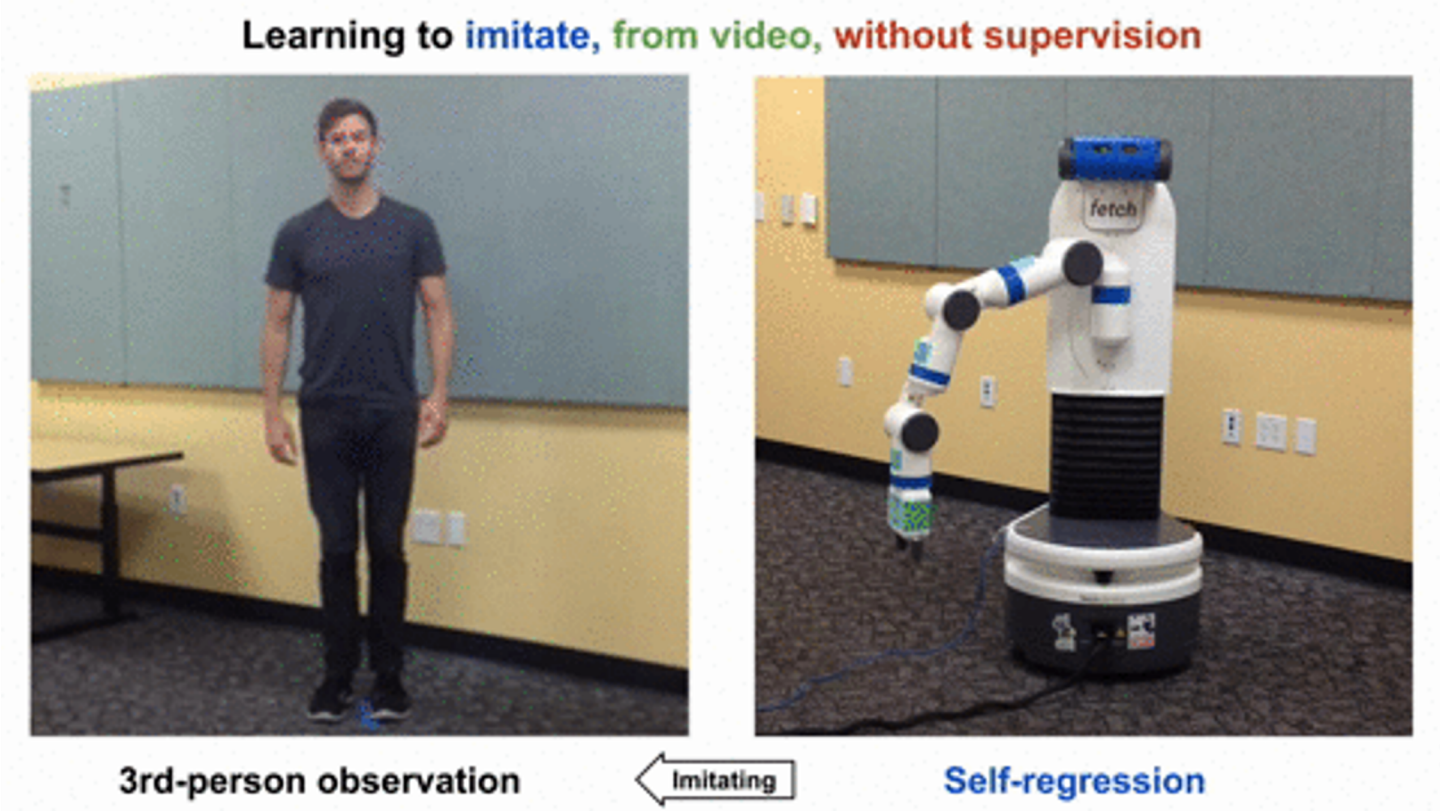


Conclusion

- Can we bring the benefits of behavioral cloning to imitation from observation?
- Yes! By
  - Learn an inverse dynamics model using available state-action pairs (not demonstrations).
  - Apply the model to predict the missing actions in the demonstrations. Then do behavioral cloning.

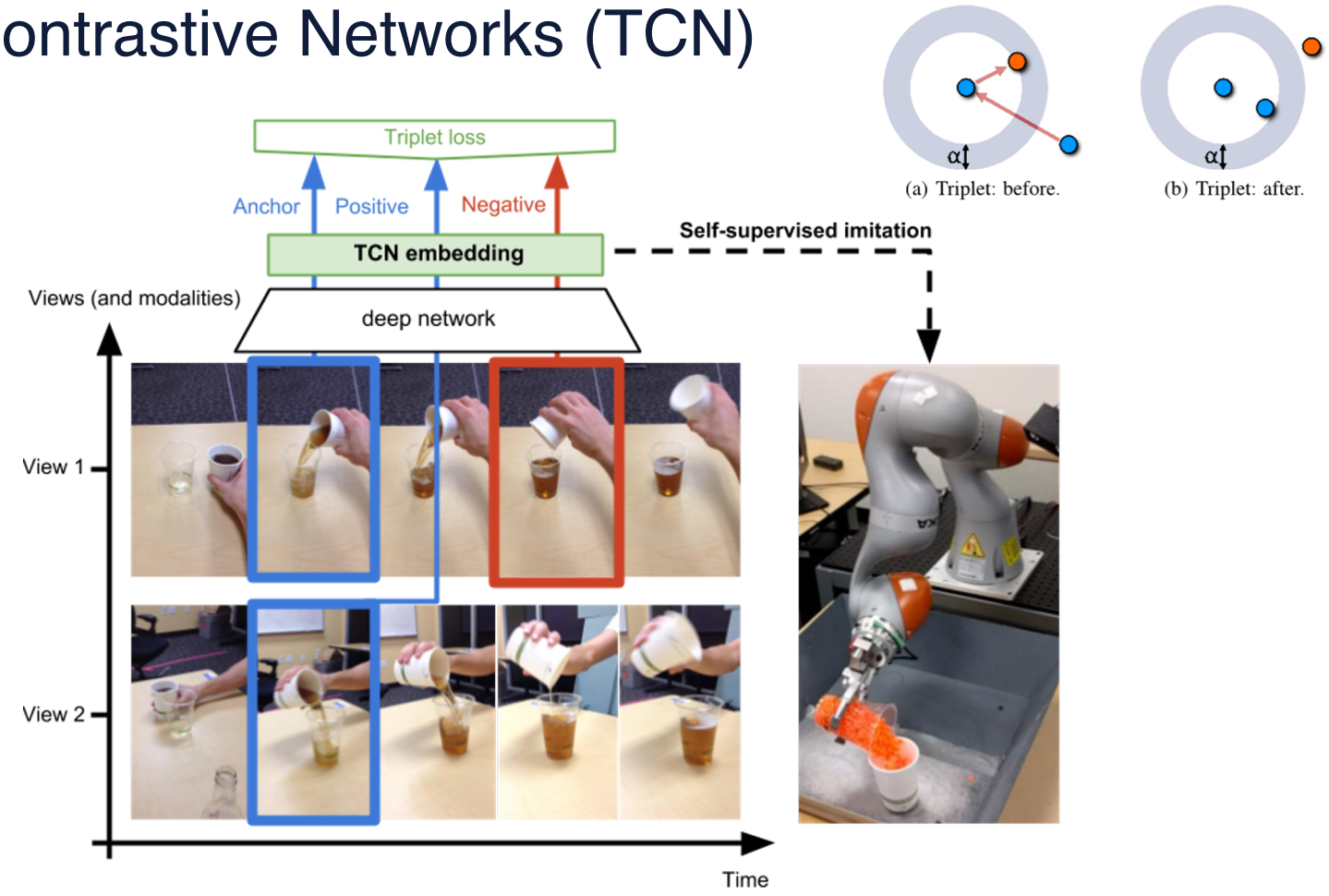
Related Works

This work cheat! There are action records!
Can we imitate purely from video?



Related Works

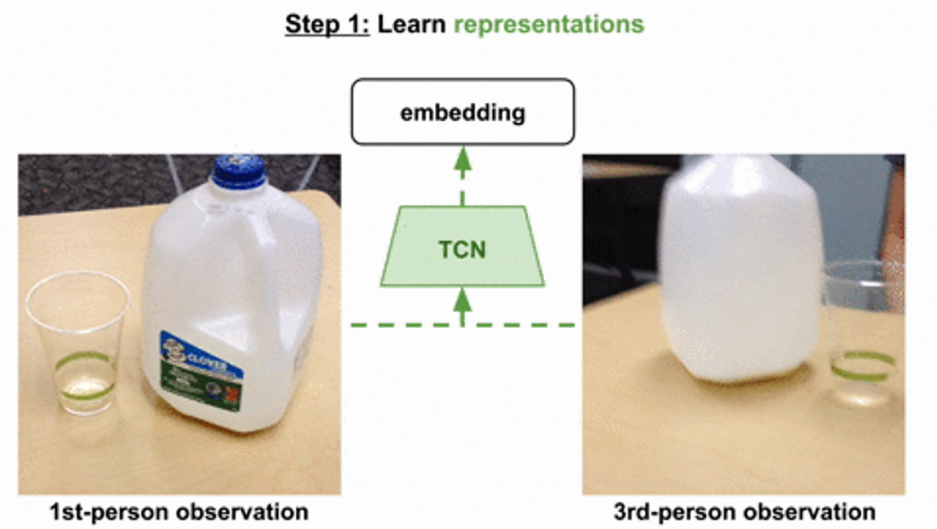
- Time-Contrastive Networks (TCN)

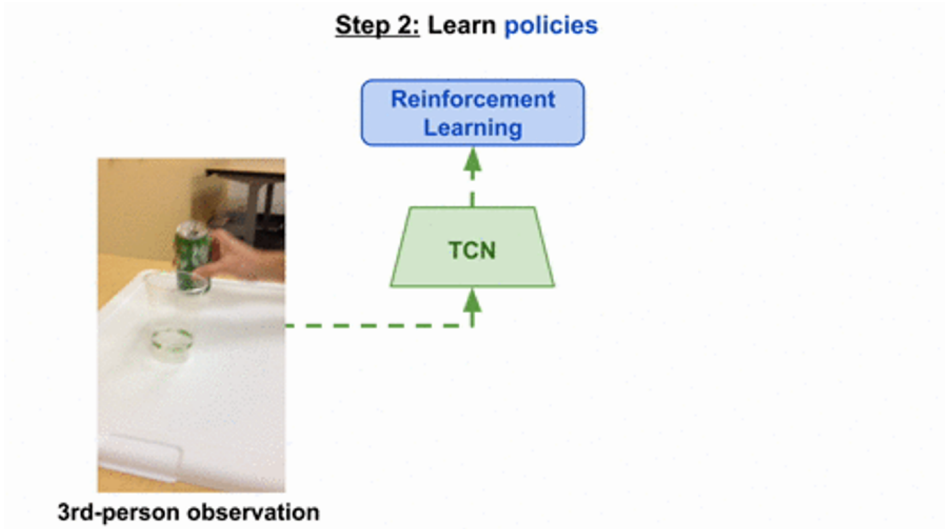


Positive: images taken from simultaneous viewpoints
Negative: images taken from a different time in the same sequence

Related Works

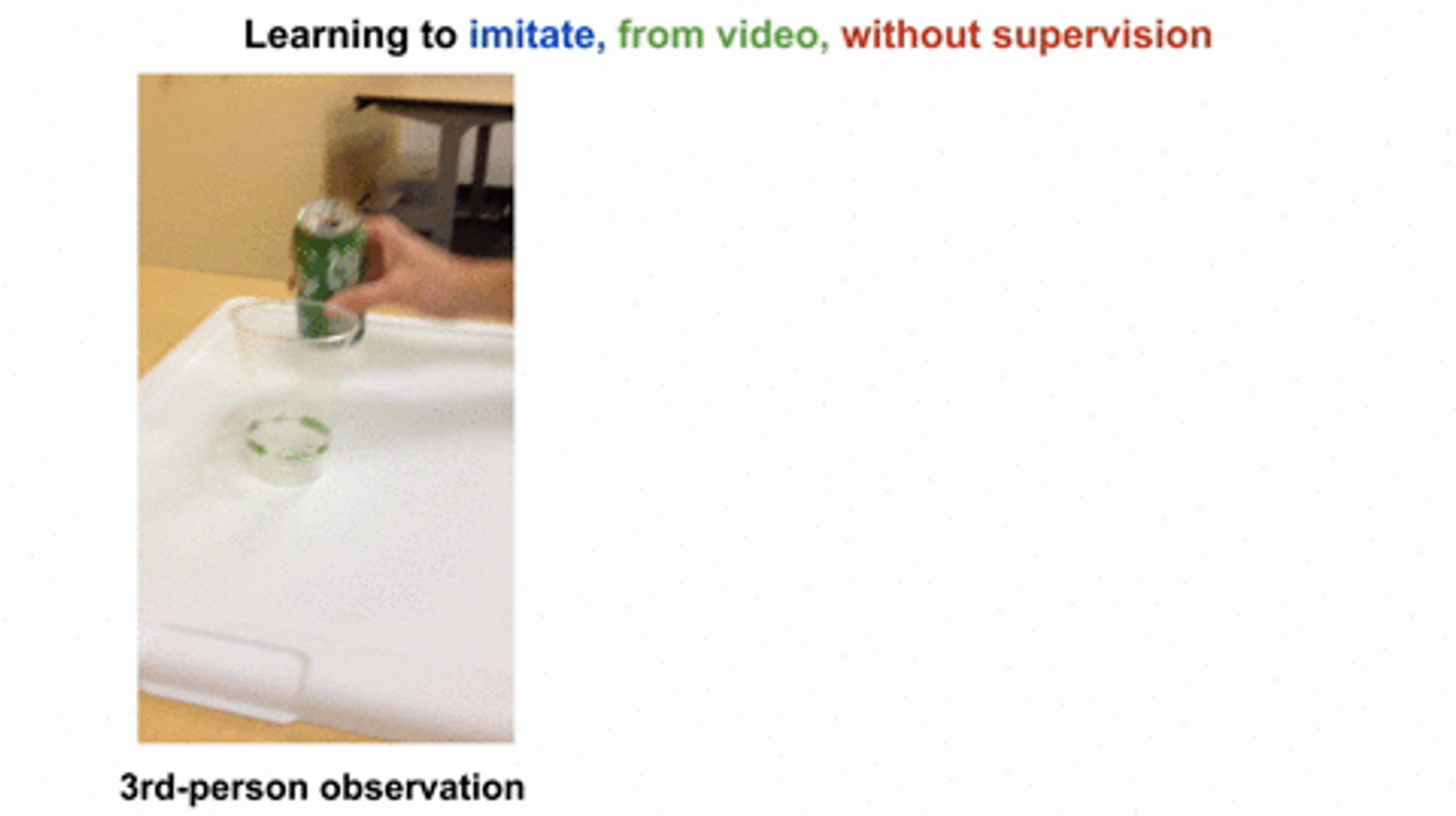
- Step 1: learn a universal embedding for each frame (ideally same state across viewpoints, agents should be close)
- Step 2: imitation learning on given demonstrations (design the reward to make embeddings close)





Related Works

Resulting policies



Reference

- Torabi, Faraz, Garrett Warnell, and Peter Stone. "Behavioral cloning from observation."
- Torabi, Faraz, Garrett Warnell, and Peter Stone. "Recent Advances in Imitation Learning from Observation."
- Sermanet, Pierre, et al. "Time-contrastive networks: Self-supervised learning from video."