Tsinghua University

Deep Reinforcement Learning

Lecture 8: Learning from Demonstrations

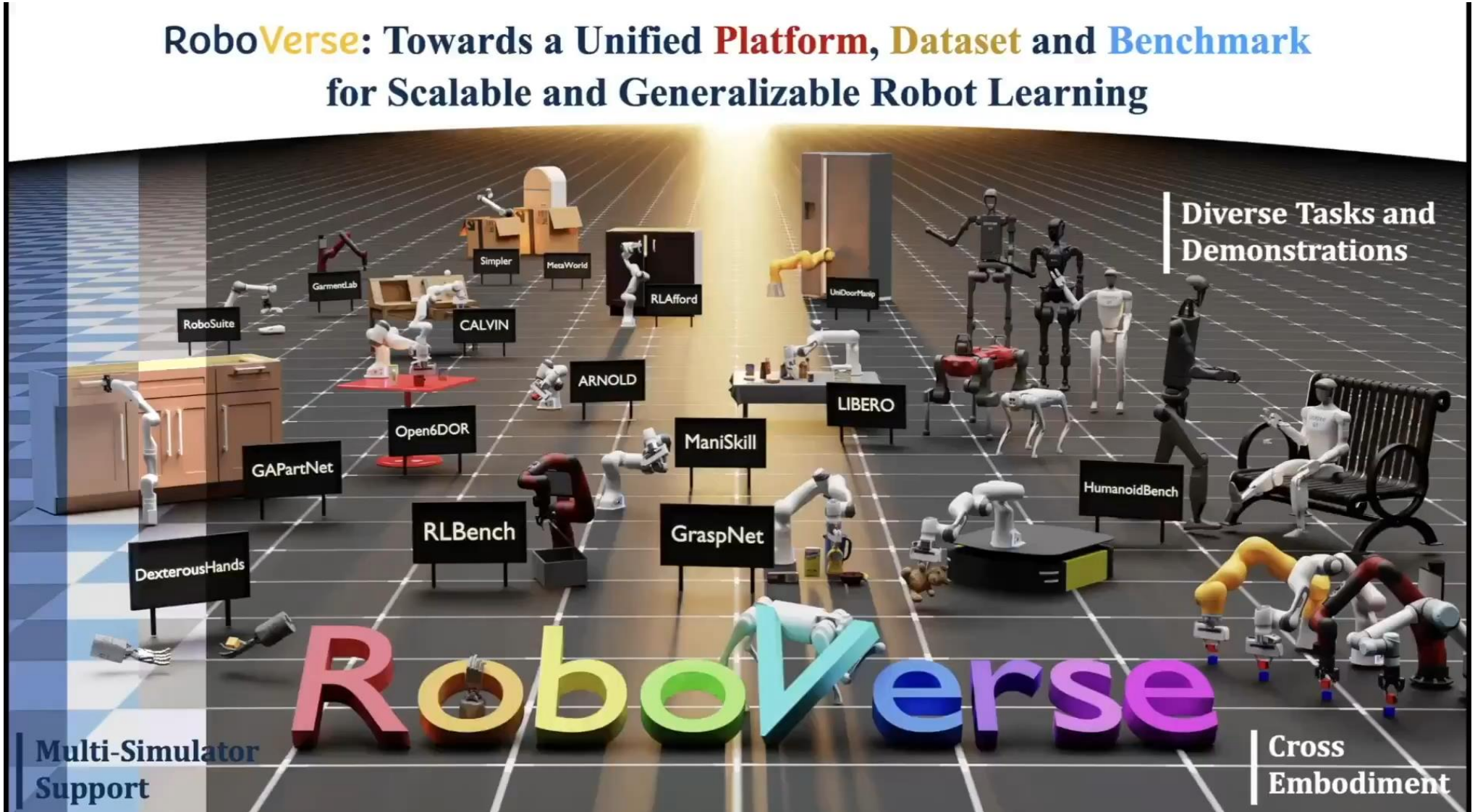
Huazhe Xu

Tsinghua University

Project & Quiz

- This weekend!
- Midterm Quiz in week 10!! In this room! The same time as our lecture.
  - A calculator
  - A pen
  - Student ID Card
  - I'll bring you some chocolate ;p
  - Midterm quiz sample questions will be released next week!

AI This Week


RoboVerse: Towards a Unified Platform, Dataset and Benchmark
for Scalable and Generalizable Robot Learning
Diverse Tasks and Demonstrations
Simpler
MetaWorld
GarmentLab
RLAfford
UniDoorManip
RoboSuite
CALVIN
ARNOLD
LIBERO
Open6DOR
ManiSkill
GAPartNet
HumanoidBench
RLBench
GraspNet
DexterousHands
RoboVerse
Multi-Simulator Support
Cross Embodiment

Previously, we have learned reinforcement learning…

- Data is collected from
  - the agent itself
  - some past experience of some agent
- However, in many tasks, we might have different quality of data.
  - For example, when you learn to play tennis, your coach will show you the correct motion.
  - Another example is your coach might also demonstrate how you do it wrong!
- In this lecture, we will learn how can we learn from some good data or demonstrations! (And we will leave dealing with bad/suboptimal data to offline RL.)

In Lec8

1. Imitation Learning
2. Improving Imitation Learning
3. Other settings in Learning from Demonstration

In Lec8

1. Imitation Learning
2. Improving Imitation Learning
3. Special settings in Learning from Demonstration

Reward? Interaction steps?

Imitation Learning

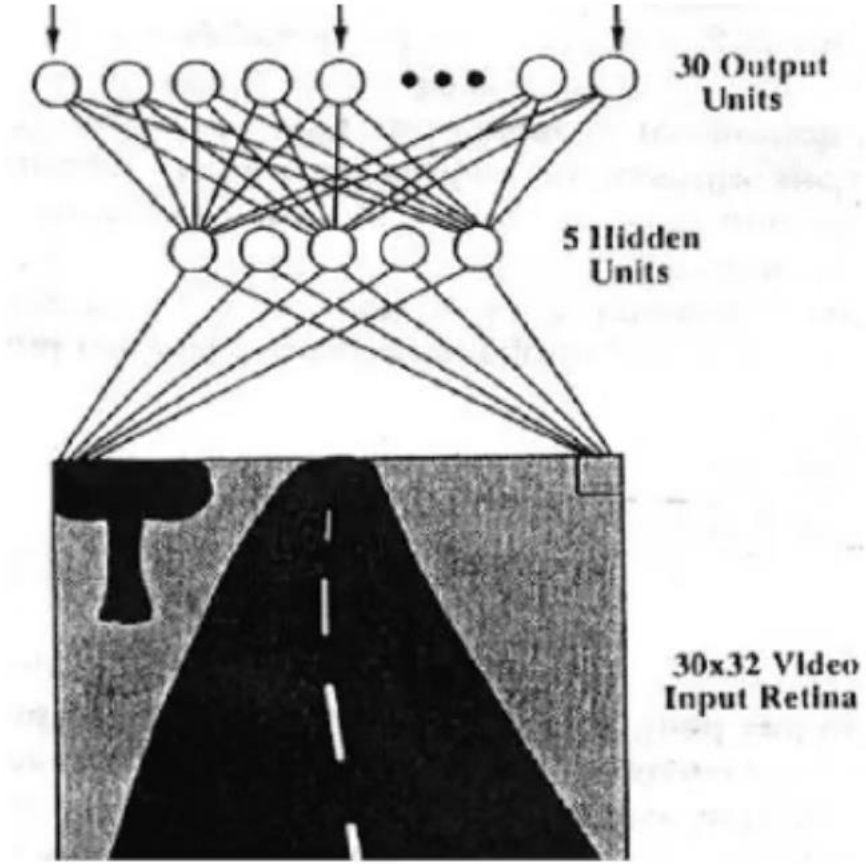
- We have many demonstrations in the format of (o_t, a_t)
  - o_t is the observation
  - a_t is the action
- Our goal is to learn a policy $\pi(a_t|o_t)$ **that maximize the likelihood or some measure of similarity** between the predicted actions and the actions from demonstrations.
- This looks familiar, right?
  - Very similar to other tasks with a supervised learning paradigm
  - E.g., image classification
  - House price regression
- If you directly minimize the loss between predicted actions and demonstration actions, this is also called **behavior cloning (BC).**

Imitation Learning

- Can the agent interact with the environment?
  - Yes! Later, you'll see some RL-based imitation learning.
- Can the expert (data collector) interact with environment?
  - Usually no.
  - Or we call it human(expert)-in-the-loop.

History: Deep Imitation Learning for Driving

- ALVINN: Autonomous Land Vehicle In a Neural Network



History: Deep Imitation Learning for Driving

ALVINN Approach

ALVINN Results

Source: Youtuber Welch Labs

Can imitation learning actually work?

- No, in most of the cases!
  - Self-driving is not solved even today.
- Reason 1: it is similar to how we describe the accumulated error in training dynamics models. The small error will ultimately become large issues.
- Reason 2: what is even worse, once the agent goes into unknown areas, it does not know how to solve the task nor recover from the unknown area to the known area. (covariate shift)
  - More formally, $\boldsymbol{p_{data}(o_t) \neq p_{\pi_\theta}(o_t)}$! Then your policy lose control.

General Imitation Learning

- Objective

$$\pi^* = \arg\min_{\pi} \mathbb{E}_{s \sim \rho_\pi^s} \left[\ell \left(\pi(\cdot \mid s), \pi_E(\cdot \mid s) \right) \right]$$

- l denotes some loss function or some distance metric.
- Distribution of s depends on rollout from π.
- $P(s \mid \pi) \rightarrow \rho_\pi(s)$: distribution of states sampled by a policy

$$\rho_\pi(s) = (1-\gamma) \sum_{t=0}^{\infty} \gamma^t P\left(s_t = s \mid \pi\right)$$

- Problem
- Cannot get access to the expert during sampling!

Credit: Weinan Zhang

As always, can we improve imitation learning?

- Since it is in a supervised learning scheme for now, we might want to go through the whole pipeline and improve IL by
  - Collecting **better dataset** in some smart way
  - Adjust your **data input** so that your model can make better decision
  - Leverage **better models** to produce **useful representations** for downstream tasks
  - Design **better loss function/training methods**
  - Smartly design **downstream tasks** to make the target task look easier
- We will go through this one by one.

In Lec8

1. Imitation Learning
2. Improving Imitation Learning
3. Special settings in Learning from Demonstration

IL improvements

- **Collecting better dataset in some smart way （data）**
- Adjust your data input so that your model can make better decision (inputs)
- Leverage better models to produce useful representations for downstream tasks (model)
- Design better loss function/training methods (training & loss)

IL improvements techniques: DAgger

- Can we make the demonstration data distribution match the policy-induced data distribution?
- DAgger: Dataset Aggregation

1. train $\pi_\theta(\mathbf{a}_t|\mathbf{o}_t)$ from human data $\mathcal{D} = \{\mathbf{o}_1, \mathbf{a}_1, \ldots, \mathbf{o}_N, \mathbf{a}_N\}$
2. run $\pi_\theta(\mathbf{a}_t|\mathbf{o}_t)$ to get dataset $\mathcal{D}_\pi = \{\mathbf{o}_1, \ldots, \mathbf{o}_M\}$
3. Ask human to label $\mathcal{D}_\pi$ with actions $\mathbf{a}_t$
4. Aggregate: $\mathcal{D} \leftarrow \mathcal{D} \cup \mathcal{D}_\pi$

DAgger original paper and its follow-up SMILe

A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning

Stéphane Ross
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213, USA
stephaneross@cmu.edu

Geoffrey J. Gordon
Machine Learning Department
Carnegie Mellon University
Pittsburgh, PA 15213, USA
ggordon@cs.cmu.edu

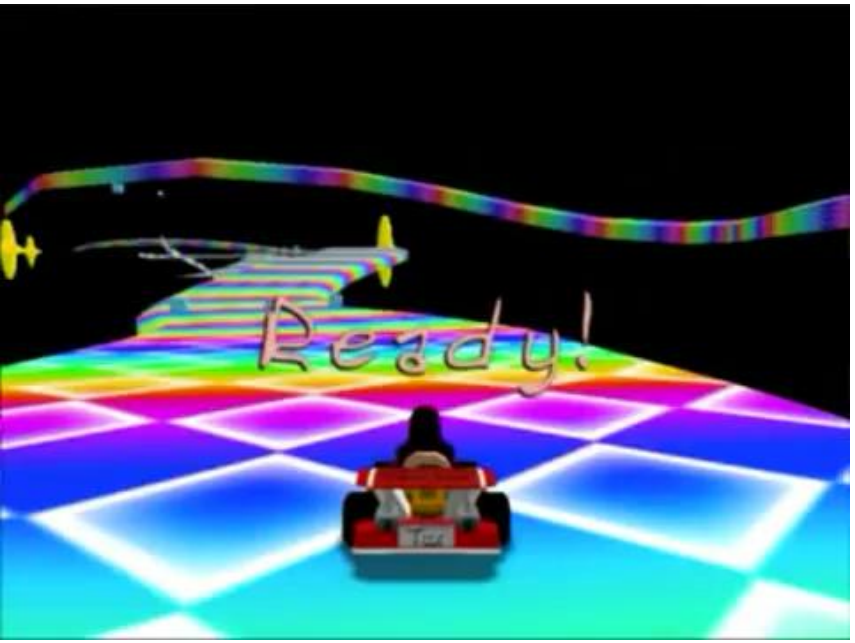
J. Andrew Bagnell
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213, USA
dbagnell@ri.cmu.edu

Efficient Reductions for Imitation Learning

Stéphane Ross
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA, USA

J. Andrew Bagnell
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA, USA

A comparison



Expert

w/o DAgger

w/ DAgger

What is not practical in DAgger?

- It might be hard for human to get the actions by looking at the observations. Human might need feedback after executing an tentative action.
- To collect these new data, we need to execute a suboptimal policy first. That is dangerous.
- In the Dagger in Carla example, everything is on-policy. That means your oracle/autopilot has to be ready to takeover anytime.

IL improvements techniques: Dataset Resampling/Reweighting

- Your dataset can be imbalanced.
  - For example, in a driving scene, 95% of the data are just driving straight down the road.
- Sample more of the rare data can be helpful.
- Another way is to put more weight on it.
- How do you determine what are the rare samples?
  - A naïve way: use prediction error!

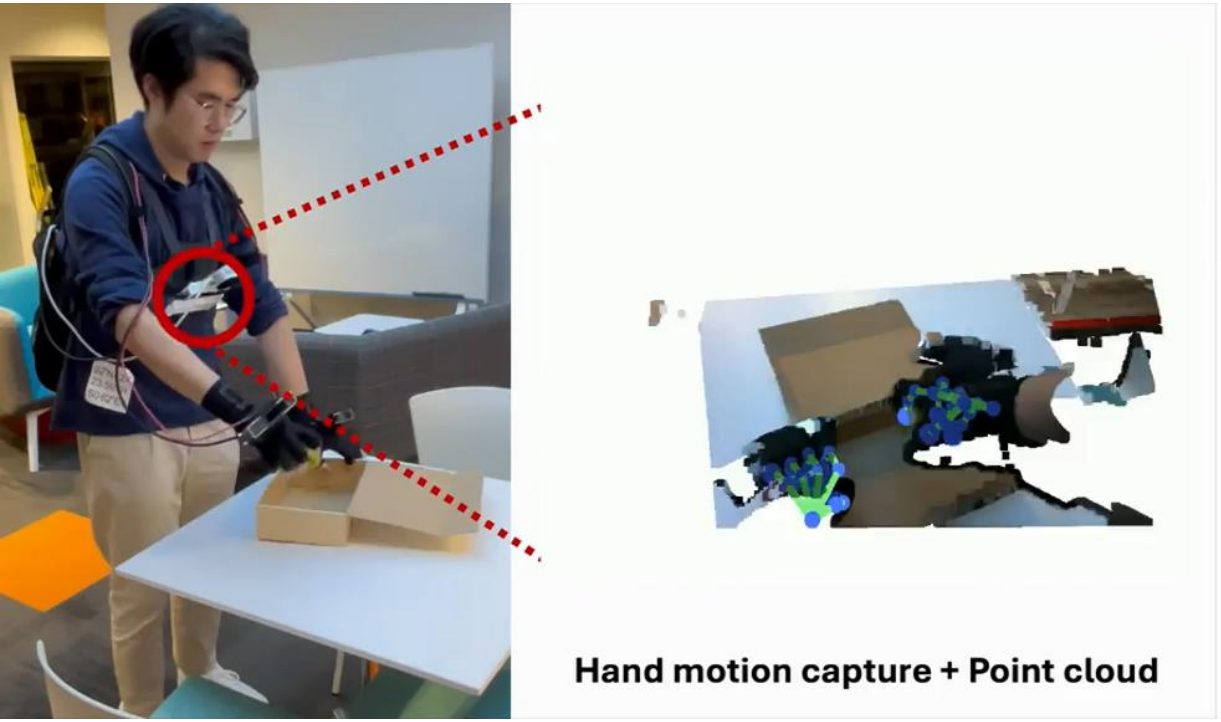
IL improvements techniques: Data Augmentation

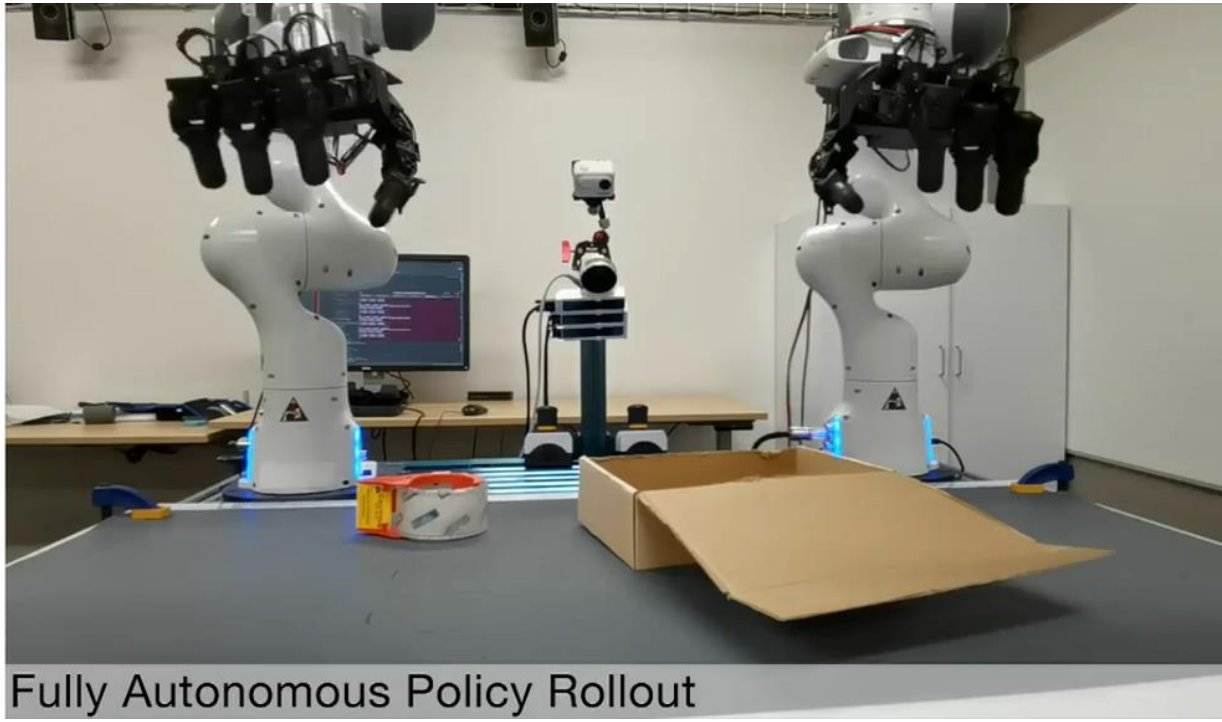
- Augment your data to scale up the dataset! But note there is a difference between IL and computer vision tasks.
- Commonly used data augmentation techniques include:
  - Translation
  - Color jittering
  - Image cropping
  - Black/White pepper noise
- Some data augmentation you should be careful:
  - Image left-right flip: sometimes this is useful if you action can also be flipped.
  - Image masking: you have to mask out pixels smartly.
- Rarely used data augmentation
  - Image rotation

Better Data Collection Techniques

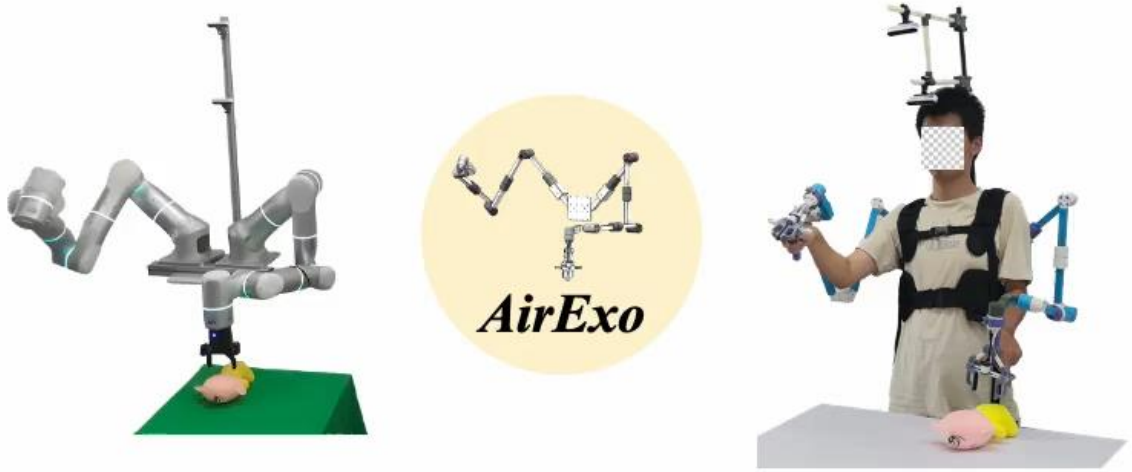
Mobile Aloha from Chelsea Lab

Better Data Collection Techniques

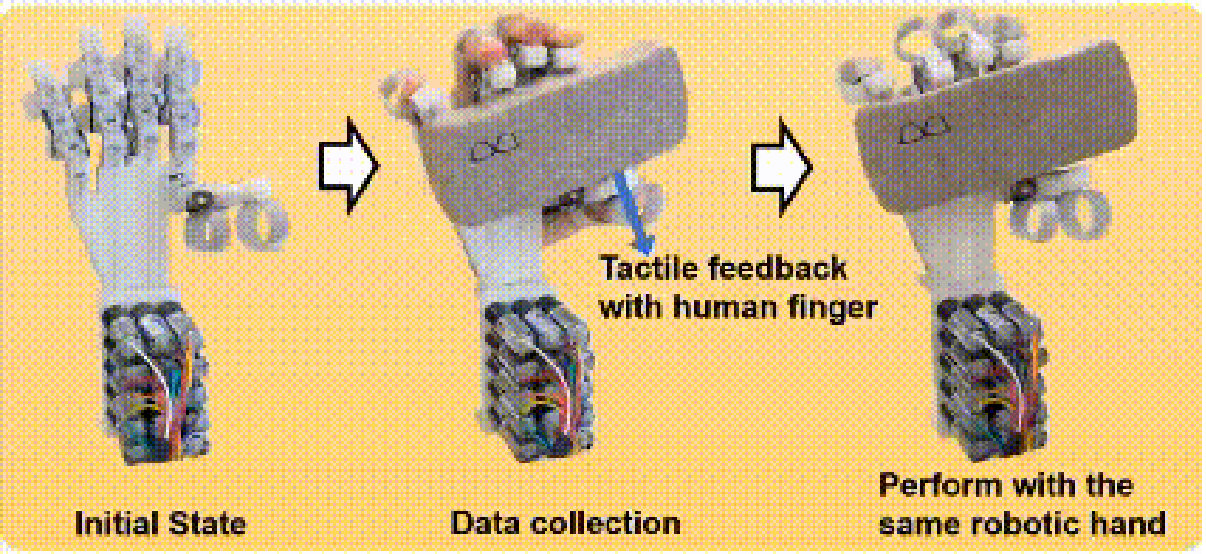




DexCap from Stanford Fei-Fei lab



AirExo from SJTU Cewu's Lab



HiroHand from TEA lab

UMI from Stanford Shuran Lab

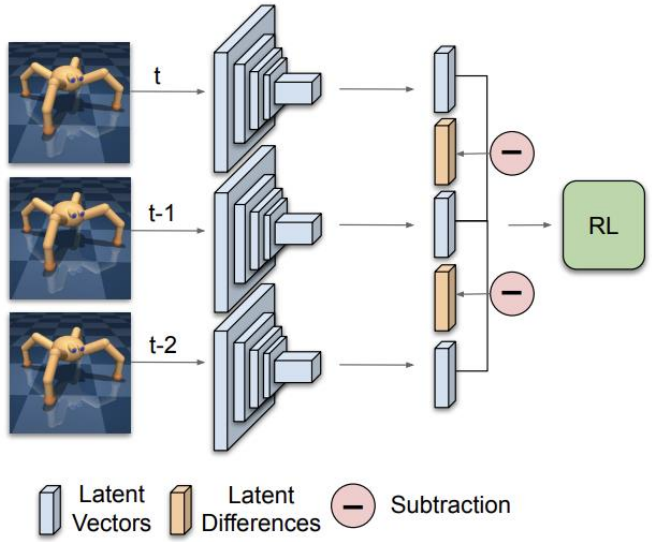
IL improvements

- Collecting better dataset in some smart way （data）
- **Adjust your data input so that your model can make better decision (inputs)**
- Leverage better models to produce useful representations for downstream tasks (model)
- Design better loss function/training methods (training & loss)

IL improvement techniques: Putting histories into your observation.

- Instead of using a single image, you stack multiple images into your observation. (Familiar, right? Same thing as in DQN!)
- The task might be non-markovian, we want the history to help.
- Other ways:
  - Use sequence models: RNN or Transformer.
  - Flare: use concatenated difference between frames

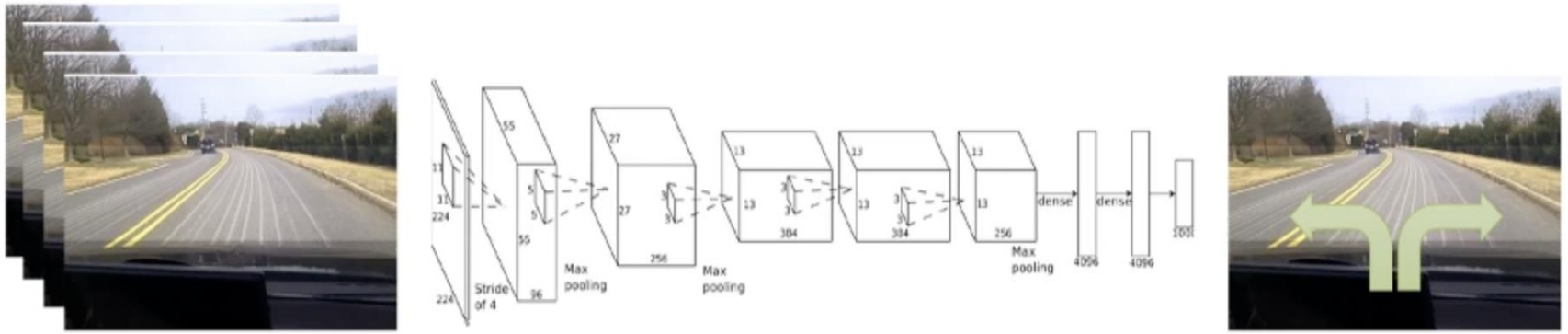
  (works in RL! But can also be applied to IL.)



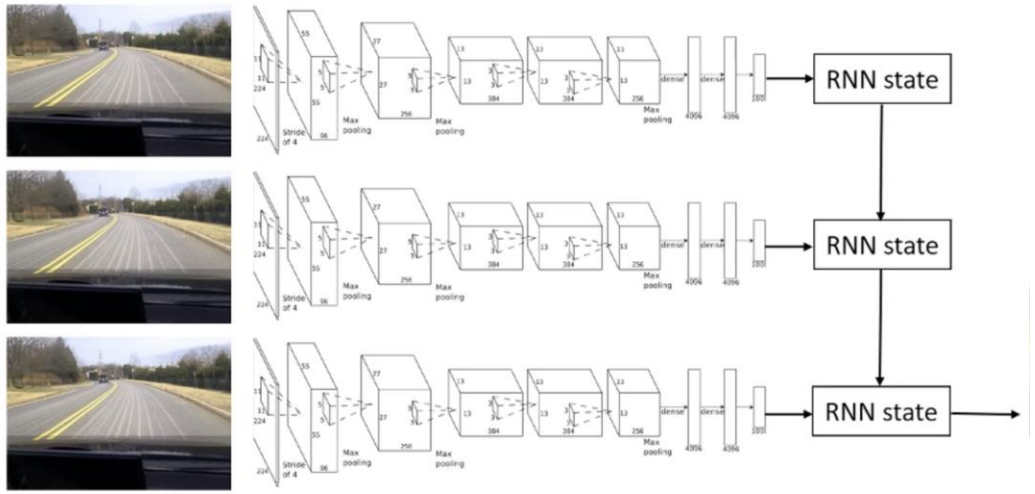
REINFORCEMENT LEARNING WITH LATENT FLOW

Wenling Shang[2,†] **, Xiaofei Wang**[1,†]**,**
Aravind Srinivas[1]**, Aravind Rajeswaran**[3]**, Yang Gao**[1]**,**
Pieter Abbeel[1] **& Michael Laskin**[1]
University of California Berkeley[1], Deepmind[2], University of Washington[3]

Illustration of Stacked Frames and RNN-Based Approach



Stacked Frames

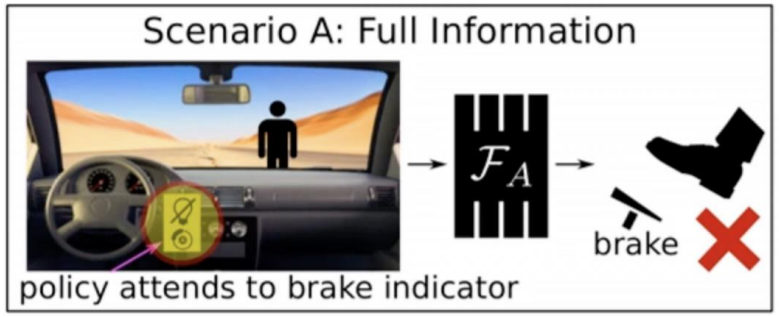


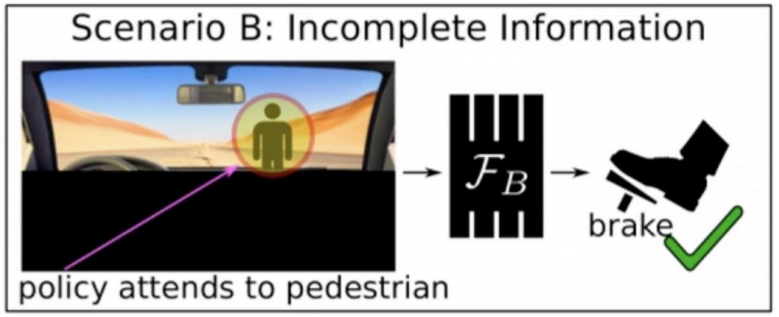
Using RNN to deal with history

Adapted from CS 285 Levine's lecture

Using histories can have causal confusions.

- The driving example
- Scenario A: whenever a driver steps on the brake, the light will be lit up. If use history observations, the policy would just check the light instead of the pedestrian in front of the car.
- Scenario B: The light is broken. The policy would attend to the pedestrian.





"causal confusion"

see: de Haan et al., "Causal Confusion in Imitation Learning"

Papers addressing causal confusion

Fighting Copycat Agents in Behavioral Cloning from Observation Histories

Chuan Wen[*1], **Jierui Lin**[*2], **Trevor Darrell**[2], **Dinesh Jayaraman**[3], **Yang Gao**[†124]
[1]Institute for Interdisciplinary Information Sciences, Tsinghua University
[2]UC Berkeley, [3]University of Pennsylvania, [4]Shanghai Qi Zhi Institute

IL improvement techniques: Multi-view Inputs

- If you are using images as input, it is partially observable.
- Consider adding more cameras or stereo cameras to provide more 3D information.
- This is very common in robotics and driving.

IL improvements

- Collecting better dataset in some smart way （data）
- Adjust your data input so that your model can make better decision (inputs)
- **Leverage better models for downstream tasks (model)**
- Design better loss function/training methods (training & loss)

IL improvement techniques: pre-trained models to extract representations

- This is an active research area and is usually referred as representation learning (for sensorimotor).
- It usually works for both BC and visual RL.

Papers:

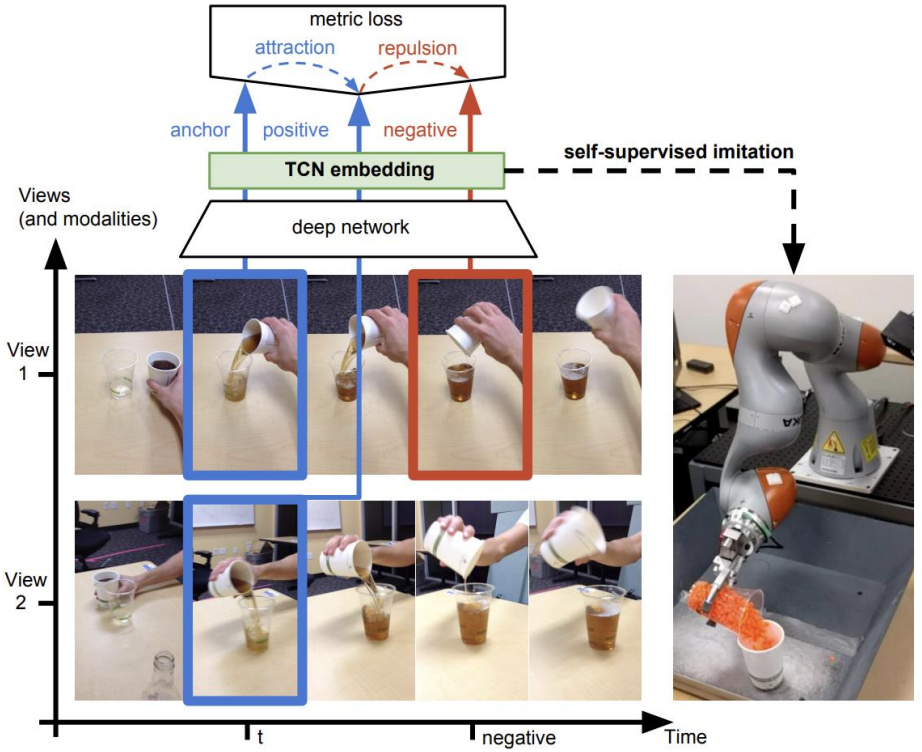
- Temporal Contrastive Network (TCN)
- The (Un)Surprising Effectiveness of Pre-Trained Vision Models for Control (PVR)
- R3M: A Universal Visual Representation for Robot Manipulation (R3M)
- Masked Visual Pre-training for Motor Control (MVP)
- Where are we in the search for an Artificial Visual Cortex for Embodied Intelligence? (VC1)

Time-Contrastive Networks: Self-Supervised Learning from Video (TCN)

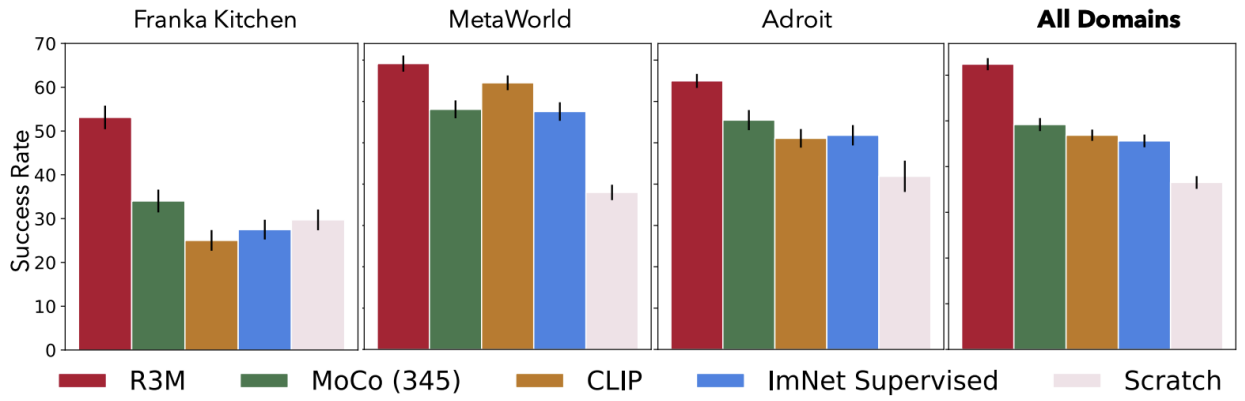
- High-level description:

1. Record synchronized multi-view videos
2. Use networks to extract representations
3. Train the network such that
   1. Temporally close frames'representation are also close to each other.
   2. Temporally distant frames' representation are also far away!

   This technique would provide **better representation for robotic tasks**!

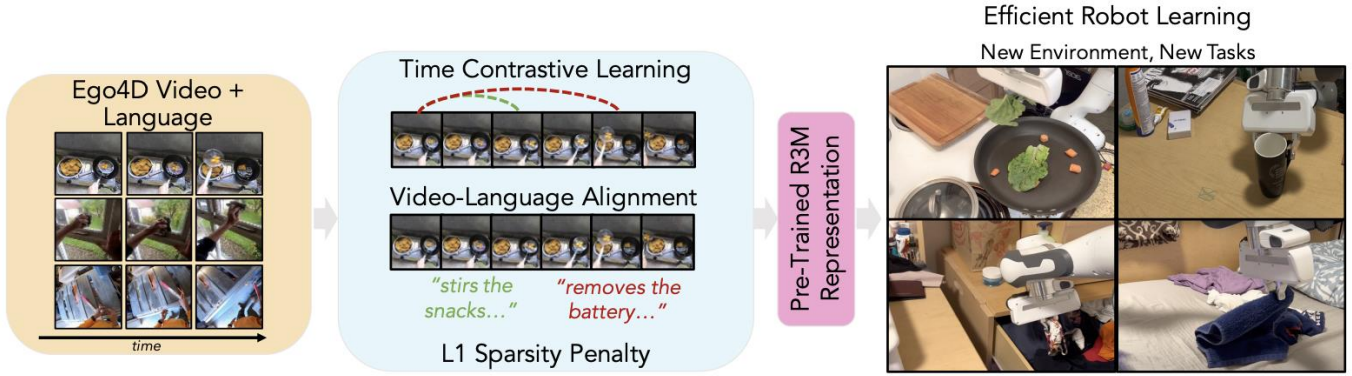


R3M



- High-level description:

1. Prepare diverse human videos such as Ego4D
2. Train with 3 losses:
    1. Video-language joint loss for capturing the semantics
    2. TCN loss for capturing temporal information
    3. L1 sparsity loss
3. When tested on a robot, it works better in performance.



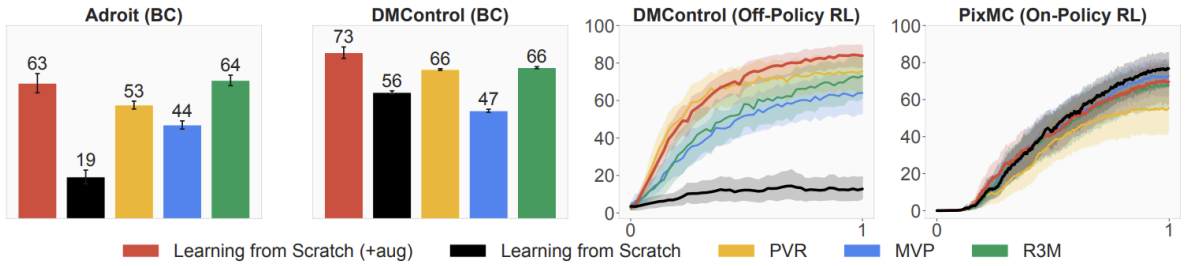
How to better leverage pre-trained model?

- Since these are recent papers, the conclusion hasn't been tested by time.
- A recent ICML paper from UCSD and THU:

On Pre-Training for Visuo-Motor Control: Revisiting a Learning-from-Scratch Baseline

Nicklas Hansen[12*], **Zhecheng Yuan**[3*], **Yanjie Ze**[14*], **Tongzhou Mu**[1*],
Aravind Rajeswaran[2†], **Hao Su**[1†], **Huazhe Xu**[35†], **Xiaolong Wang**[1†]

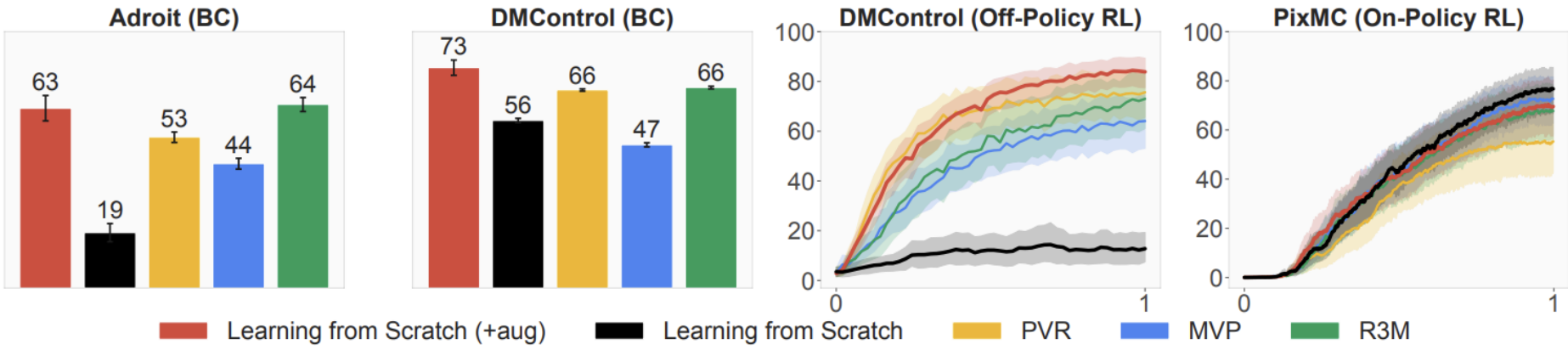
[1]University of California San Diego, [2]Meta AI,
[3]Tsinghua University, [4]Shanghai Jiao Tong University, [5]Shanghai Qi Zhi Institute
*Equal contributions, †Equal advising



On Pre-Training for Visuo-Motor Control: Revisiting a Learning-from-Scratch Baseline

Learning-from-scratch baseline has competitive performance with recent methods that leverage frozen visual representations trained on large-scale vision datasets.

Which one is correct? That can be a good topic for you to figure out!



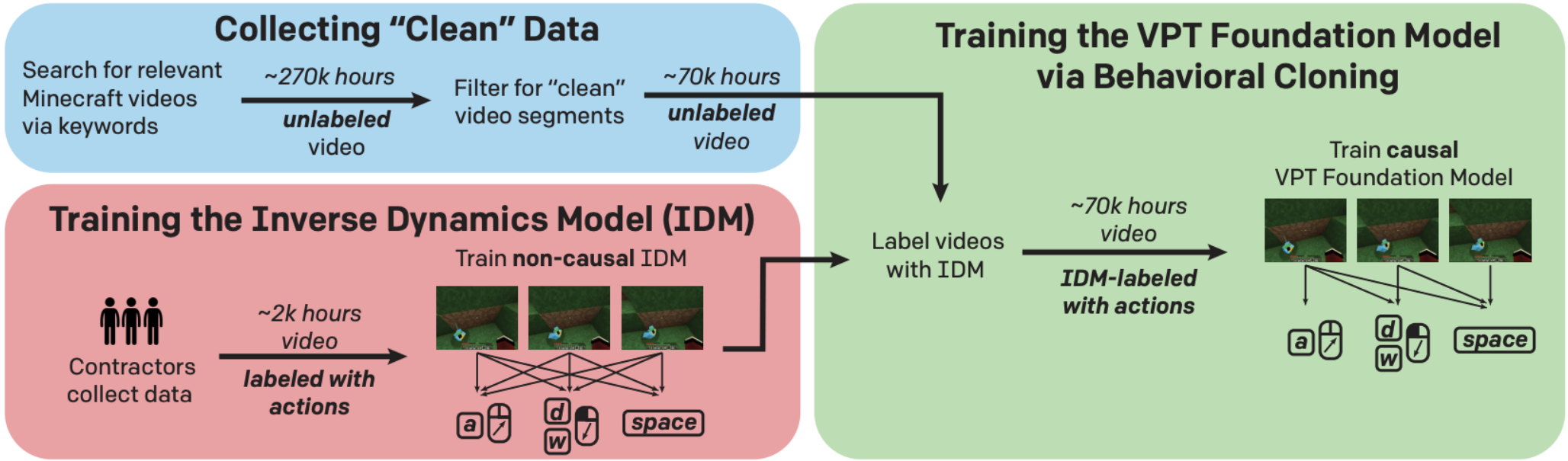
IL improvement techniques: Pre-training Inverse Dynamics Models

- What is an inverse dynamics model?
  - Given state s_t and next state s_{t+1}, predict action $a_t = f(s_t, s_{t+1})$
  - Imagine what action would help you to jump from current state to the next state.
- What can we do if we have an inverse dynamics model?
  - We can use it to label actions if we only have observations, e.g., in video games or youtube videos!
- VPT:
  - You have a small set of demonstration data in Minecraft game.
  - You have near infinite videos of Minecraft on youtube/bilibili.
  - You train an inverse model on small set and use this model to label large sets.
  - Do IL and the execution of your learned policy can help you to enrich the small dataset.
- I put it into the pre-training schemes. But you can also regard it as a way to collect better data.

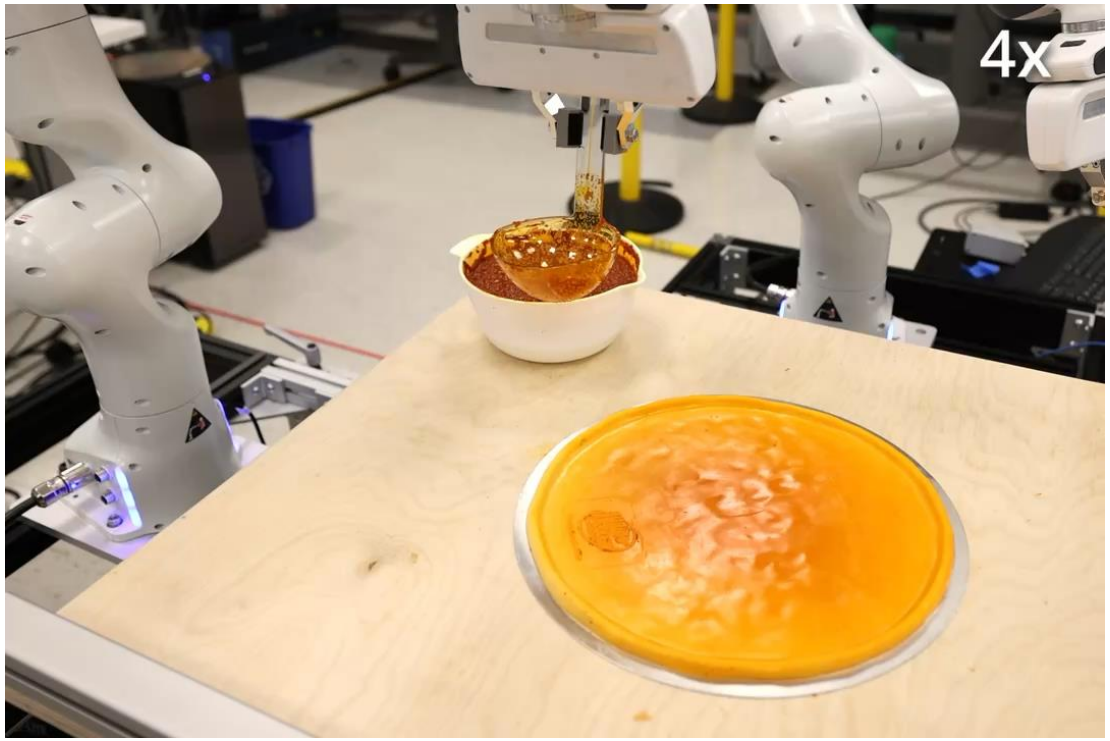
Video PreTraining (VPT)

Video PreTraining (VPT): Learning to Act by Watching Unlabeled Online Videos

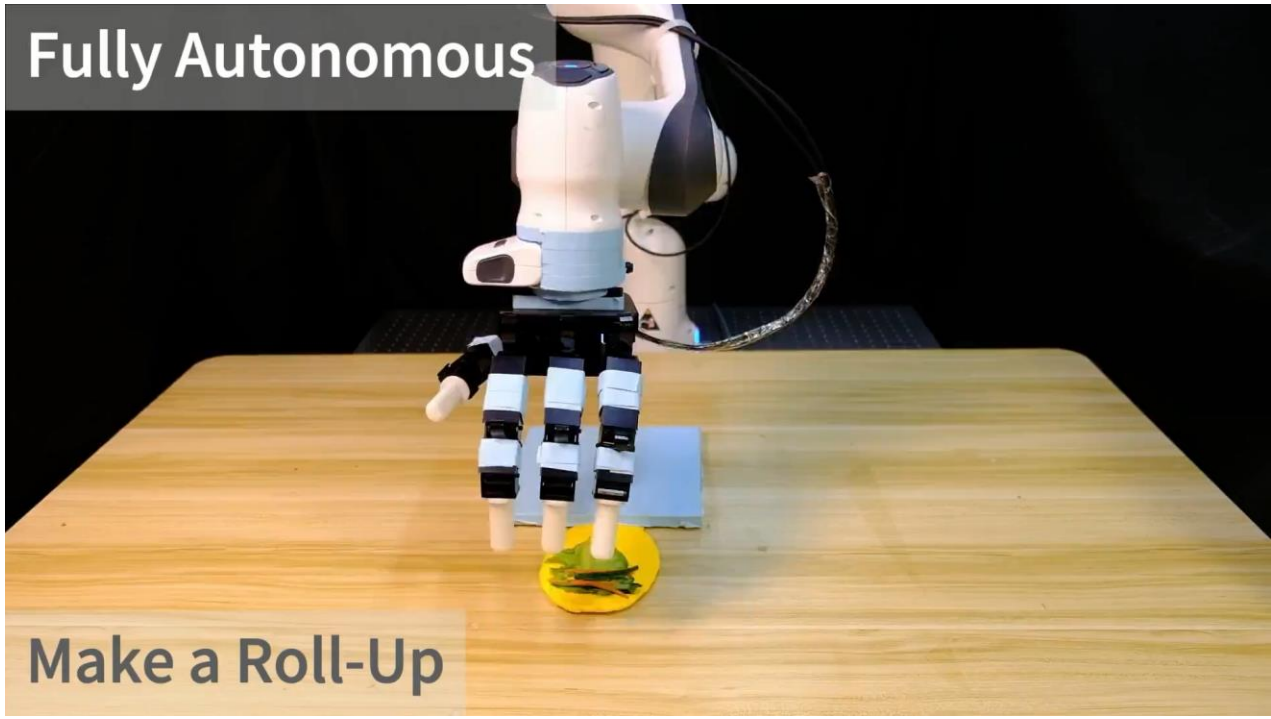
Bowen Baker[*†] bowen@openai.com
Ilge Akkaya[*†] ilge@openai.com
Peter Zhokhov[*†] peterz@openai.com
Joost Huizinga[*†] joost@openai.com
Jie Tang[*†] jietang@openai.com
Adrien Ecoffet[*†] adrien@openai.com
Brandon Houghton[*†] brandon@openai.com
Raul Sampedro[*†] raulsamg@gmail.com
Jeff Clune[*†‡] jclune@gmail.com



Diffusion Policy



DP from Columbia U



DP3 from THU TEA lab

IL improvements

- Collecting better dataset in some smart way （data）
- Adjust your data input so that your model can make better decision (inputs)
- Leverage better models to produce useful representations for downstream tasks (model)
- **Design better loss function/training methods (training & loss)**

IL improvement techniques: Multi-modality-Aware Training

- While the dataset can be good (i.e., from expert and accomplish tasks near optimally), the innate nature of human behaviors are multimodal.
- An example:

Xu et al, End-to-end Learning of Driving Models from Large-scale Video Datasets

(c) multiple possible actions: turn left or go straight

- Shall we use classification loss like cross-entropy or regression loss like MSE?
  - Cross-entropy! MSE would make your agent going directly toward the obstacle.

IL improvement techniques: Multi-modality-Aware Training

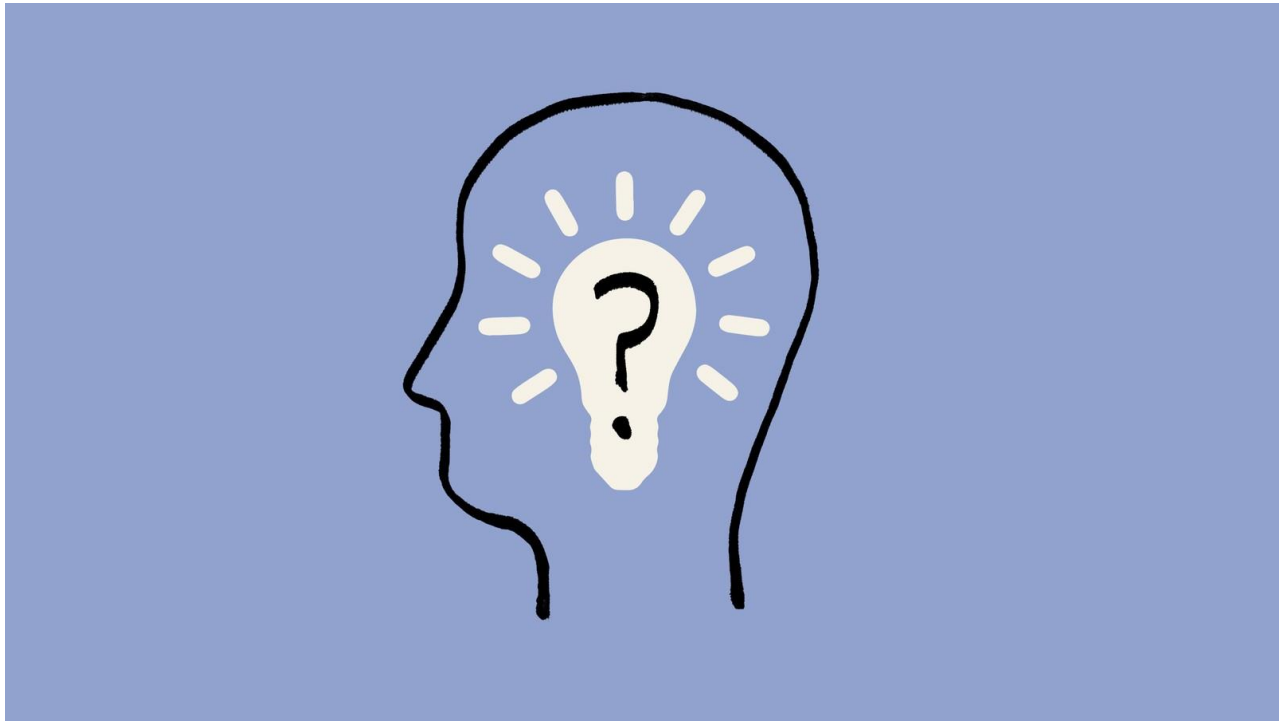
- What other methods can you use to capture multi-modal behavior?
  - Use Mixture-of-Gaussian instead of gaussian distribution.
  - Learn a policy that is conditioned on a latent variable.
    - An intuitive explanation: the model takes in both an image & some noise, the noise will determine the modality.
    - Not going into details: Variation Autoencoder, Normalizing Flow, Stein Variational Gradient Descent
  - Autoregressive discretization
    - If we have discrete actions, everything is solved!
    - But discretize the actions can be intractable, e.g., your action have high dimensionality.
    - Then you discretize the actions dimension one by one: given an image we output a discretized action dim 1, then input action dim 1 to another neural network to output dim 2.

In Lec10

1. Imitation Learning
2. Improving Imitation Learning
3. Special settings in Learning from Demonstration

Previously, most of the settings are offline.

- What if online interactions are allowed? What can you think of?



Reinforcement Learning from Demonstrations

- There are multiple ways to use demonstrations.
- Today we introduce Deep Q-learning from Demonstrations (DQfD).

Deep Q-learning from Demonstrations

Todd Hester
Google DeepMind
toddhester@google.com

Matej Vecerik
Google DeepMind
matejvecerik@google.com

Olivier Pietquin
Google DeepMind
pietquin@google.com

Marc Lanctot
Google DeepMind
lanctot@google.com

Tom Schaul
Google DeepMind
schaul@google.com

Bilal Piot
Google DeepMind
piot@google.com

Dan Horgan
Google DeepMind
horgan@google.com

John Quan
Google DeepMind
johnquan@google.com

Andrew Sendonaris
Google DeepMind
sendos@yahoo.com

Ian Osband
Google DeepMind
iosband@google.com

Gabriel Dulac-Arnold
Google DeepMind
gabe@squirrelsoup.net

John Agapiou
Google DeepMind
jagapiou@google.com

Joel Z. Leibo
Google DeepMind
jzl@google.com

Audrunas Gruslys
Google DeepMind
audrunas@google.com

DQfD

- DQfD uses not only the collected experience but also expert demonstrations, i.e., expert demos in the replay buffer.
- At the heart of this approach is this loss:

$$J(Q) = J_{DQ}(Q) + \lambda_1 J_n(Q) + \lambda_2 J_E(Q) + \lambda_3 J_{L2}(Q)$$

- J_{DQ} is the bellman loss, J_n is the n-step loss, and J_{L2} is L2 regularization.
- The major change is J_E. In this loss, with any state, the Q-value of some action a is at least a margin smaller than Q-value of an expert action. a_E is the expert action. l is a hinge loss between a_E and a.

$$J_E(Q) = \max_{a \in A}[Q(s,a) + l(a_E, a)] - Q(s, a_E)$$

Exemplar papers with demonstrations

VRL3: A Data-Driven Framework for Visual Deep Reinforcement Learning

Che Wang[1,2*] **Xufang Luo**[3] **Keith Ross**[1] **Dongsheng Li**[3]

[1] New York University Shanghai
[2] New York University
[3] Microsoft Research Asia, Shanghai, China

Reinforcement learning with Demonstrations from Mismatched Task under Sparse Reward

Yanjiang Guo[1], **Jingyue Gao**[1], **Zheng Wu**[2], **Chengming Shi**[1], **Jianyu Chen**[1,3,*]
[1]Institute for Interdisciplinary Information Sciences, Tsinghua University
[2]University of California, Berkeley
[3]Shanghai Qizhi Institute
{guoyj22, gaojy19, shicm19}@mails.tsinghua.edu.cn, zheng_wu@berkeley.edu
* Correspondence to: Jianyu Chen (jianyuchen@tsinghua.edu.cn)

Learning Complex Dexterous Manipulation with Deep Reinforcement Learning and Demonstrations

Aravind Rajeswaran[1*], Vikash Kumar[1,2*], Abhishek Gupta[3], Giulia Vezzani[4],
John Schulman[2], Emanuel Todorov[1], Sergey Levine[3]

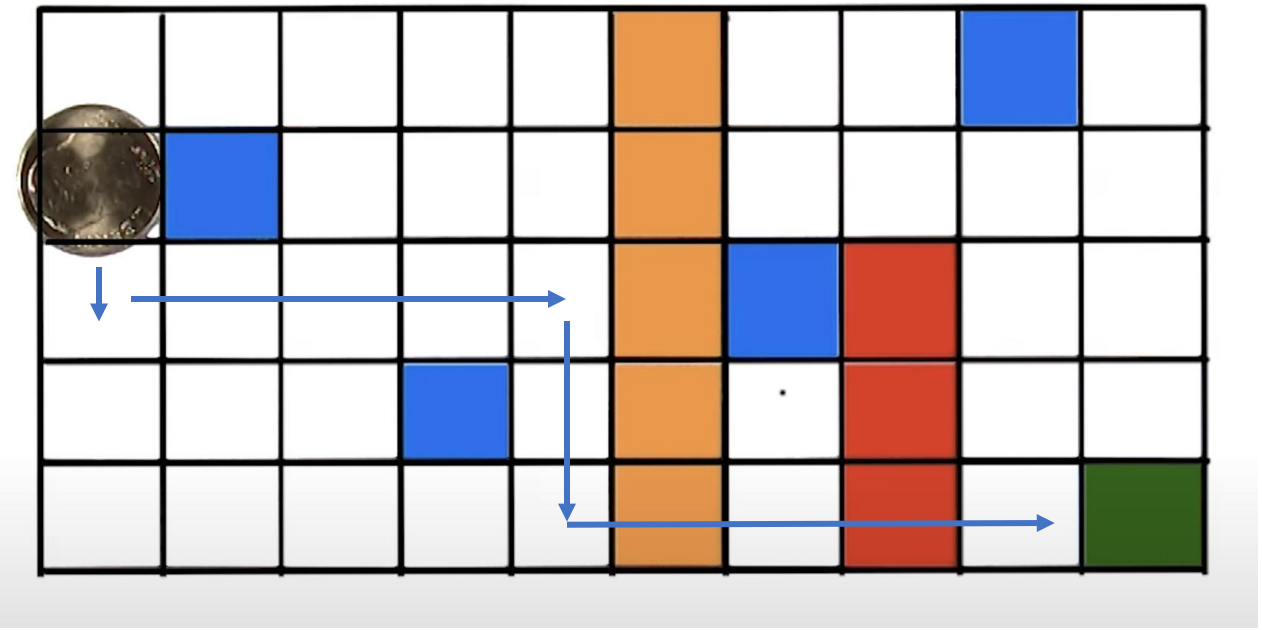
Other Settings in Imitation Learning

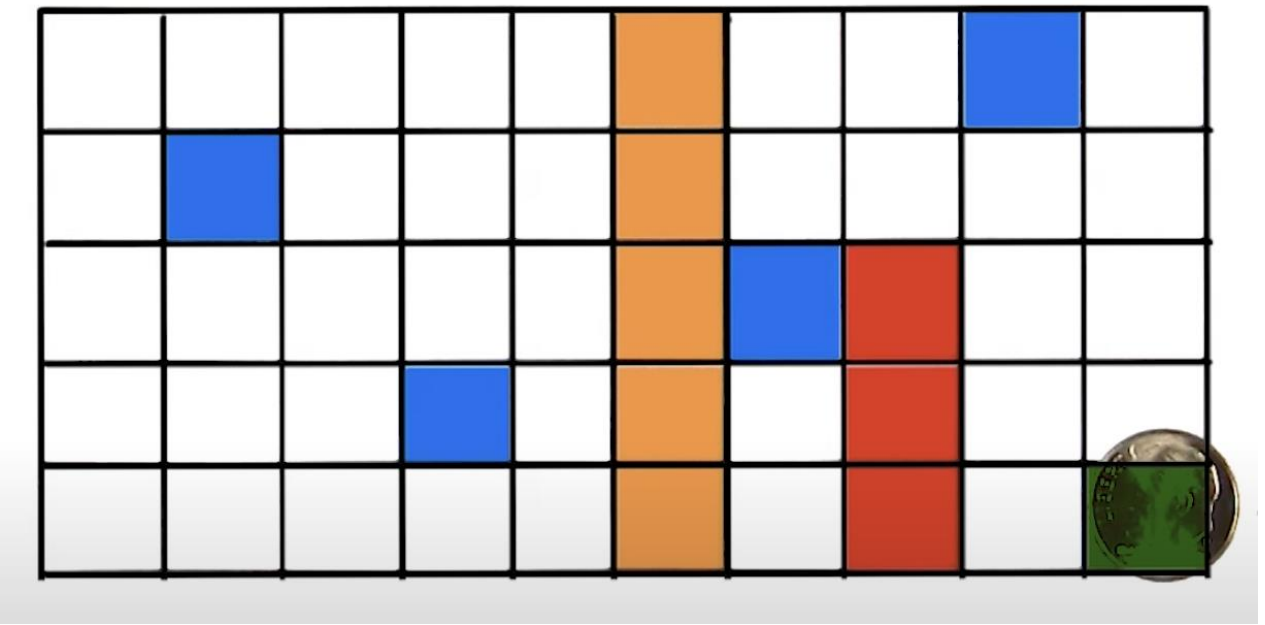
- Learning from observations where actions are missing
- Model-based imitation learning where dynamics model is provided or learned
- Third-person imitation learning
  - Just like how humans learn from their parents/teachers, we don't always learn things from a first person view.
  - Moreover, your teachers appearance and shape might be very different from you. But you can still learn from him/her.
- Learning from suboptimal demonstrations
- Learning from instructions rather than demonstrations

Another way to learn from demonstrations: Inverse RL

- Instead of learning policies from demonstrations
- Can we learn about the reward function?
- Can we then use these reward function to train an RL agent?
- We will learn these in future lectures if time permits ☺

Understand IRL through an Example





- What can you infer?

Source: Udacity

Why do we need inverse RL?

- For science!
  - Model animal and human behaviors

Dynamic Inverse Reinforcement Learning for Characterizing Animal Behavior

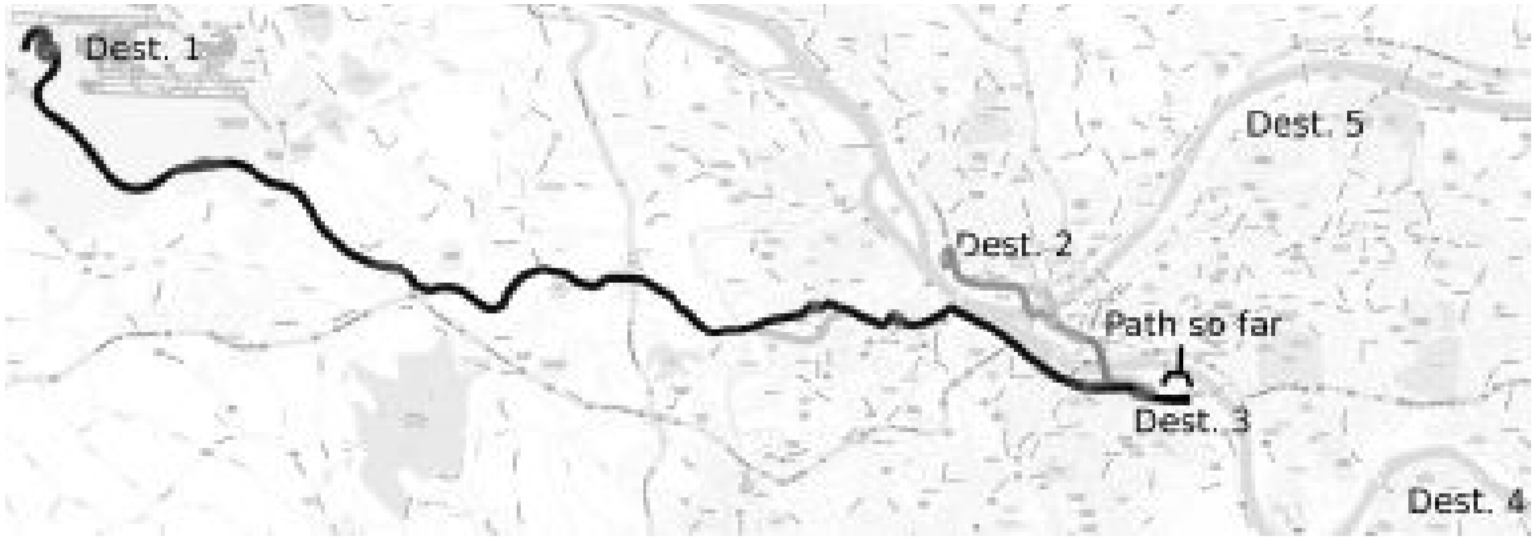
Zoe C. Ashwood[1,2,*] **Aditi Jha**[1,3,*] **Jonathan W. Pillow**[1]

[1]Princeton Neuroscience Institute, Princeton University
[2]Dept. of Computer Science, Princeton University
[3]Dept. of Electrical and Computer Engineering, Princeton University
{zashwood, aditijha, pillow}@princeton.edu

- Imitation
  - Presupposition: reward function provides the most succinct and transferable definition of the task
- Modeling of other agents, both adversarial and cooperative

IRL in Urban Navigation

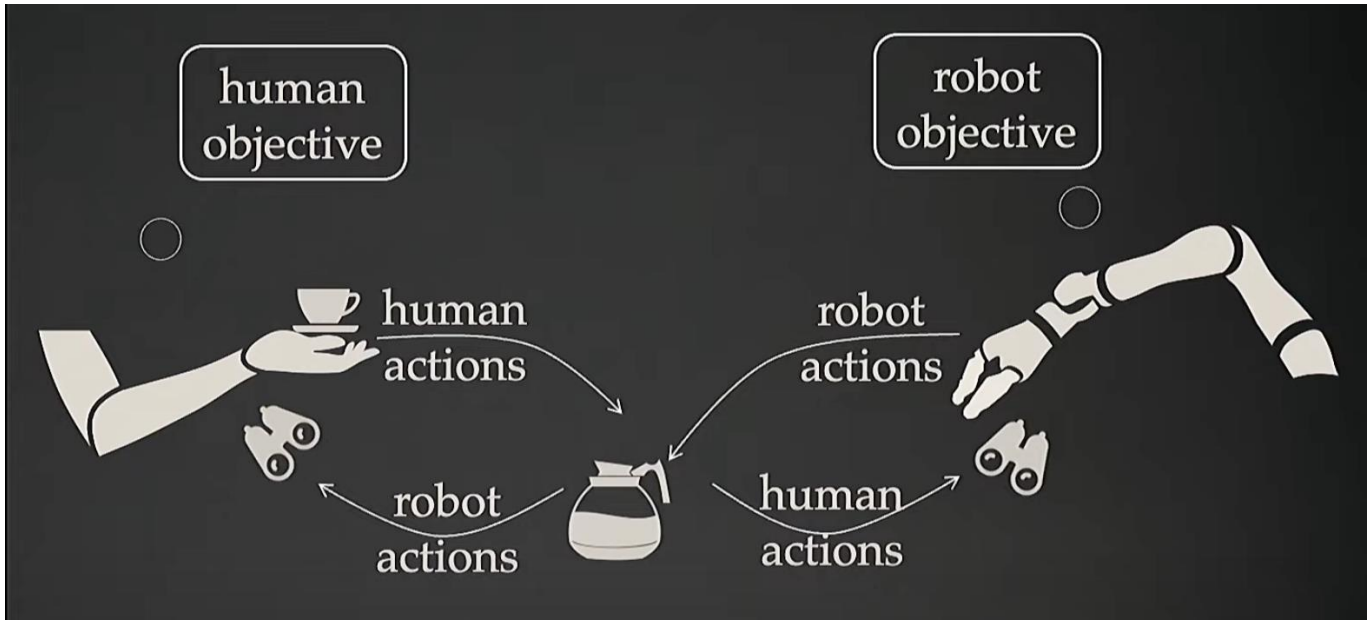
- Infer destination based of partial navigation trajectory

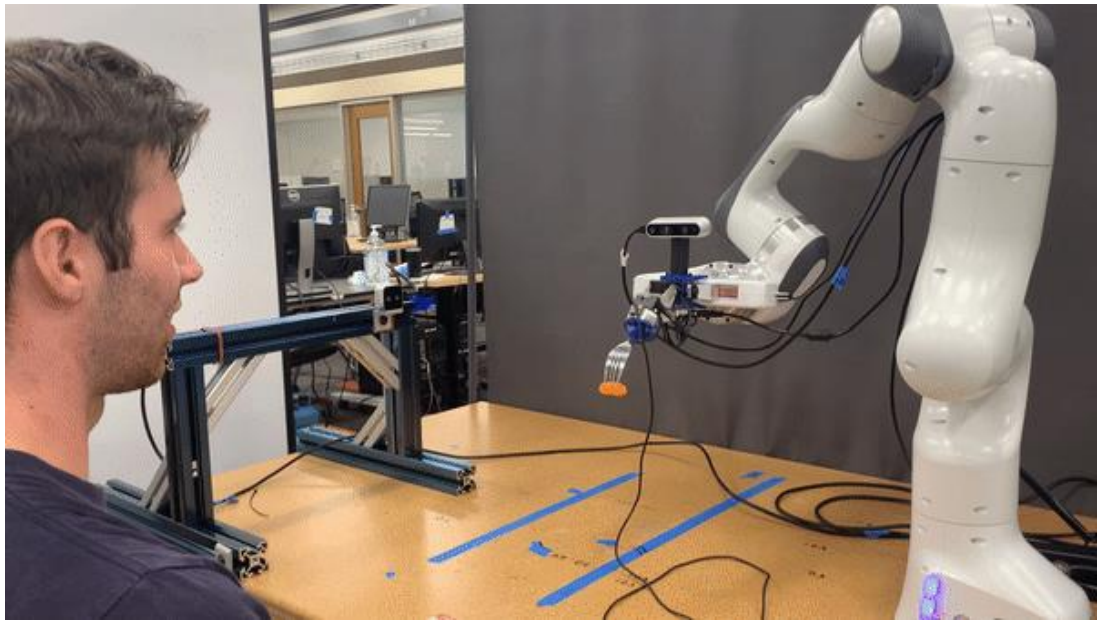


Maximum Entropy Inverse Reinforcement Learning

Brian D. Ziebart, Andrew Maas, J.Andrew Bagnell, and **Anind K. Dey**
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
bziebart@cs.cmu.edu, amaas@andrew.cmu.edu, dbagnell@ri.cmu.edu, anind@cs.cmu.edu

IRL in Human-Robot Interaction (HRI)





Source: Talk from Anca Dragan

IRL v.s. BC

Which has the most succinct description: π^* v.s. R^*?

- It depends.
  - But for RL and planning, reward can be better.
  - From a human video, it is hard to tell what is the optimal policy but easier to tell about the reward.

Basic Principle

- Find a reward function R^* which explains the expert behavior.
- Find R^* such that

$$\mathbf{E}\left[\sum_{t=0}^{\infty}\gamma^t R^*(s_t) \mid \pi^*\right] \geq \mathbf{E}\left[\sum_{t=0}^{\infty}\gamma^t R^*(s_t) \mid \pi\right] \quad \forall \pi$$

- Challenges:
  - R=0 is always a solution. Reward function ambiguity!
  - We observe trajectories rather than policies. How to compute the left-hand side?
  - Expert has to be optimal.
  - It is not possible to enumerate policies.

Feature-Based Reward Function

$$\textbf{Let } R(s) = w^\top \phi(s), \textbf{ where } w \in \Re^n, \textbf{ and } \phi : S \to \Re^n.$$

$$\begin{aligned} \mathrm{E}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t) \mid \pi\right] &= \mathrm{E}\left[\sum_{t=0}^{\infty} \gamma^t w^\top \phi(s_t) \mid \pi\right] \\ &= w^\top \mathrm{E}\left[\sum_{t=0}^{\infty} \gamma^t \phi(s_t) \mid \pi\right] \\ &= w^\top \mu(\pi) \end{aligned}$$

- μ is the feature expectation.
- Recall that $\mathrm{E}\left[\sum_{t=0}^{\infty} \gamma^t R^*(s_t) \mid \pi^*\right] \geq \mathrm{E}\left[\sum_{t=0}^{\infty} \gamma^t R^*(s_t) \mid \pi\right] \quad \forall \pi$

$$\textbf{Find } w^* \textbf{ such that } w^{*\top} \mu(\pi^*) \geq w^{*\top} \mu(\pi) \quad \forall \pi$$

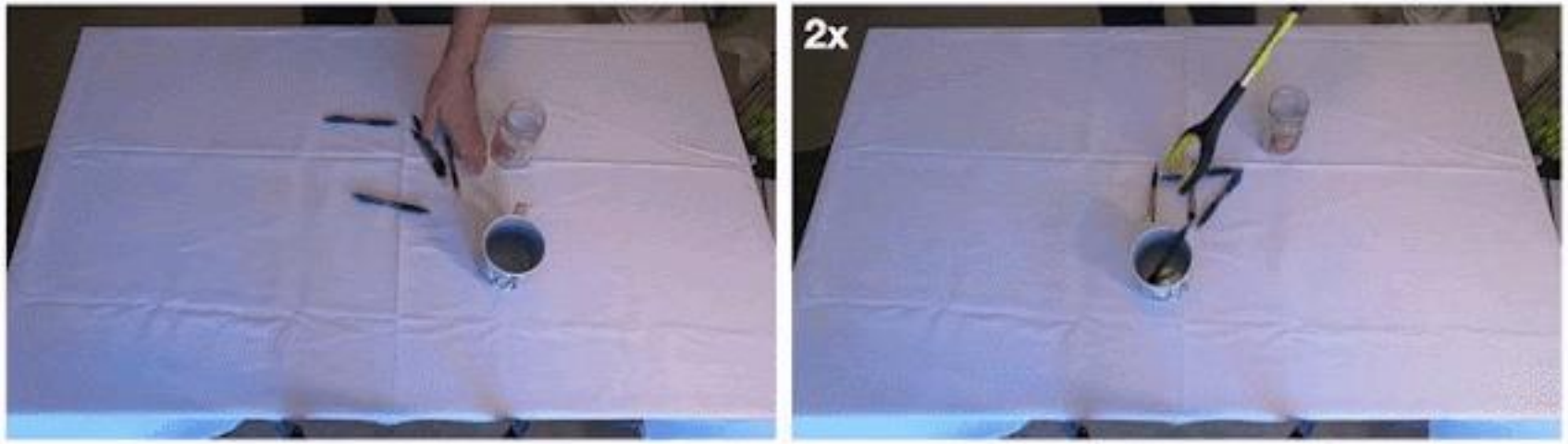
What can you do if demos are videos?

- Follow the idea of IRL
  - Reward learning from videos!
- How can you achieve that?

Source: google ai blog

What are the challenges?

- Different domain!

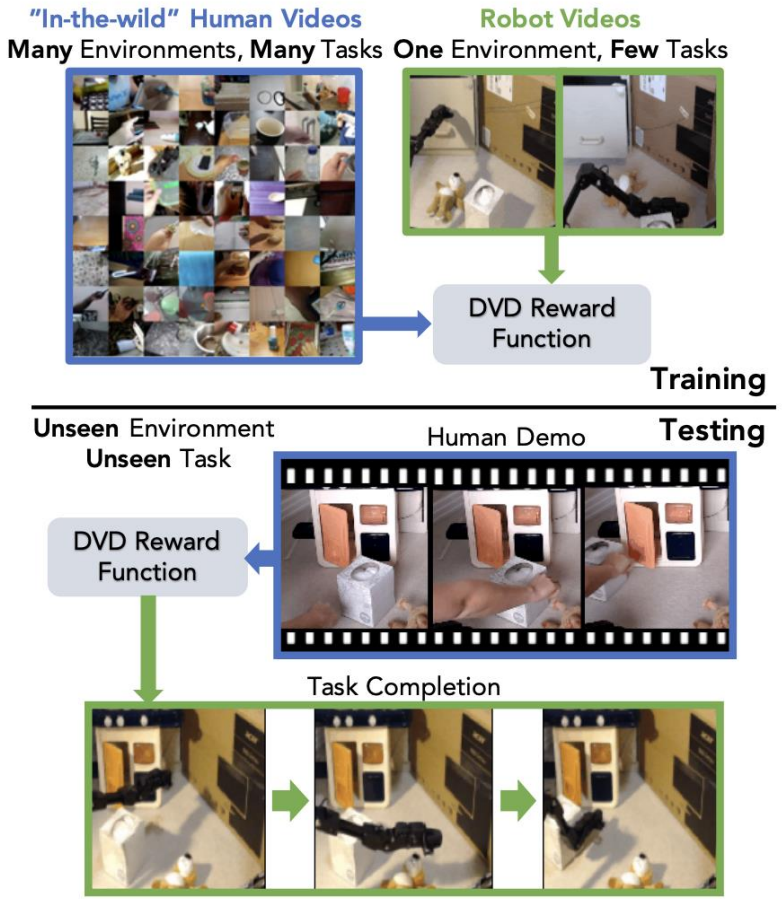


Learning Generalizable Robotic Reward Functions from "In-The-Wild" Human Videos

Annie S. Chen, Suraj Nair, Chelsea Finn
Stanford University

- Domain-agnostic Discriminator
  - A neural network that learns similarity Among tasks.
    + Positive samples, same task
    - Negative samples, different task

    Tasks are from human or robot.
- Video Predictor
  - Serves as a dyanmics model



Learning Generalizable Robotic Reward Functions from "In-The-Wild" Human Videos

Annie S. Chen, Suraj Nair, Chelsea Finn
Stanford University

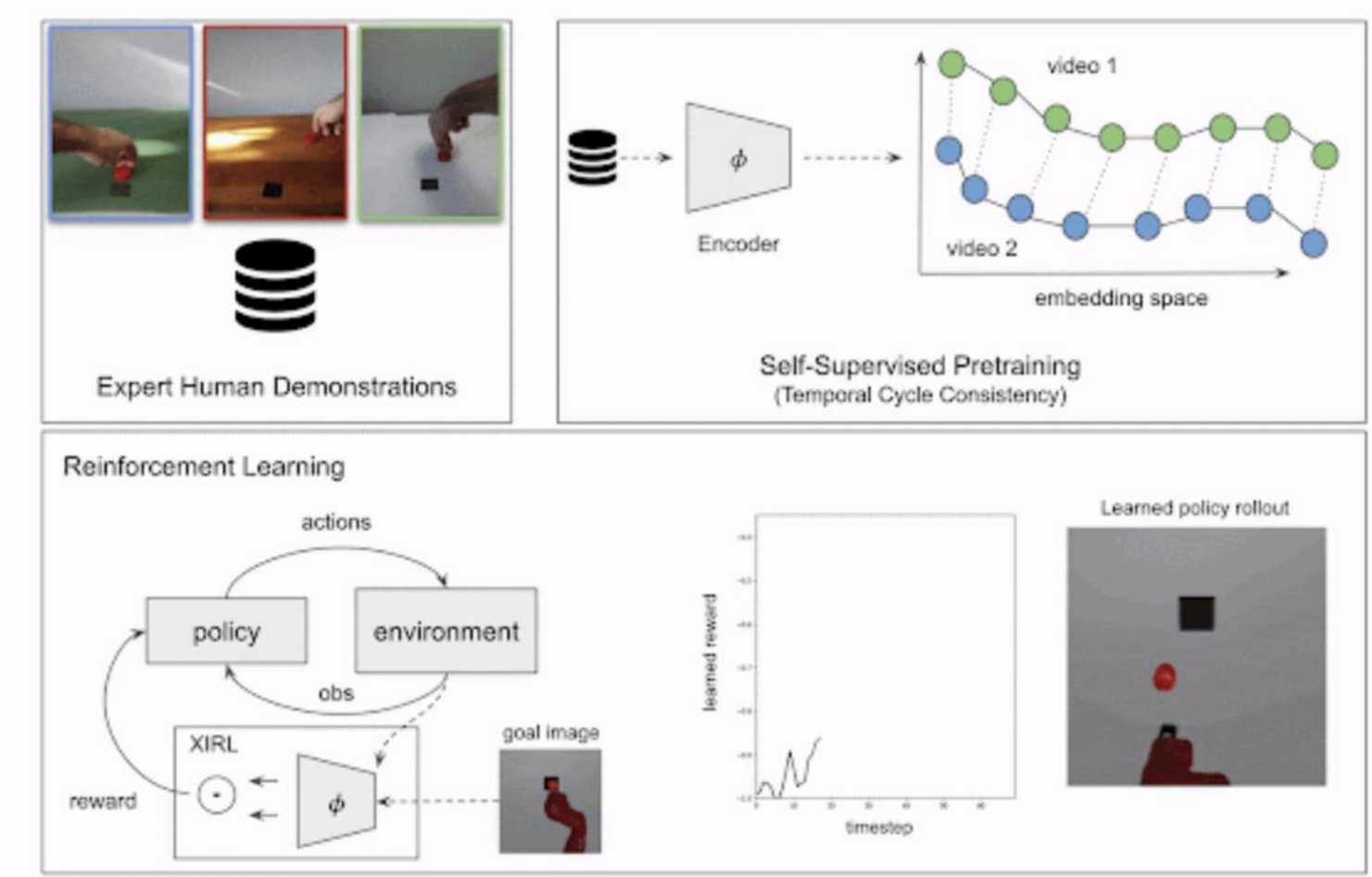
- Algorithm

Algorithm 1 DOMAIN-AGNOSTIC VIDEO DISCRIMINATOR (DVD)

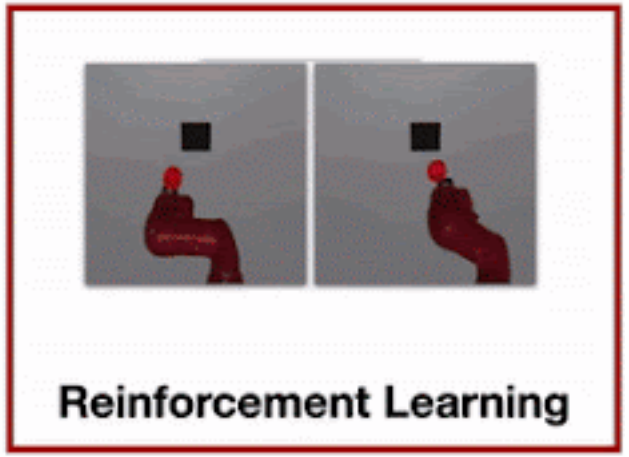
// Training DVD
Require: $\mathcal{D}^h$ human demonstration data for N tasks $\{\mathcal{T}_n\}$
Require: $\mathcal{D}^r$ robot demonstration data for M tasks $\{\mathcal{T}_m\} \subseteq \{\mathcal{T}_n\}$
Require: Pre-trained video encoder f_{enc}
Randomly initialize θ
while training **do**
    Sample anchor video $d_i \in \mathcal{D}^h \cup \mathcal{D}^r$
    Sample positive video $d_i' \in \{\mathcal{D}^h_{\mathcal{T}_i}\} \cup \{\mathcal{D}^r_{\mathcal{T}_i}\} \setminus d_i$
    Sample negative video $d_j \in \{\mathcal{D}^h_{\mathcal{T}_j}\} \cup \{\mathcal{D}^r_{\mathcal{T}_j}\} \forall j \neq i$
    Update $\mathcal{R}_\theta$ with d_i, d_i', d_j according to Eq. 1
// Planning Conditioned on Video Demo
Require: Trained reward function $\mathcal{R}_\theta$ & video prediction model p_ϕ
Require: Human video demo d_i for task $\mathcal{T}_i$
for trials $1, ..., n$ **do**
    Sample $\{a^{1:G}_{1:H}\}$ & get predictions $\{\tilde{s}^g_{1:H}\} \sim \{p_\phi(s_0, a^g_{1:H})\}$
    Step $a^*_{1:H}$ which maximizes $\mathcal{R}_\theta(\tilde{s}^g_{1:H}, d_i)$

XIRL:
Cross-embodiment Inverse Reinforcement Learning

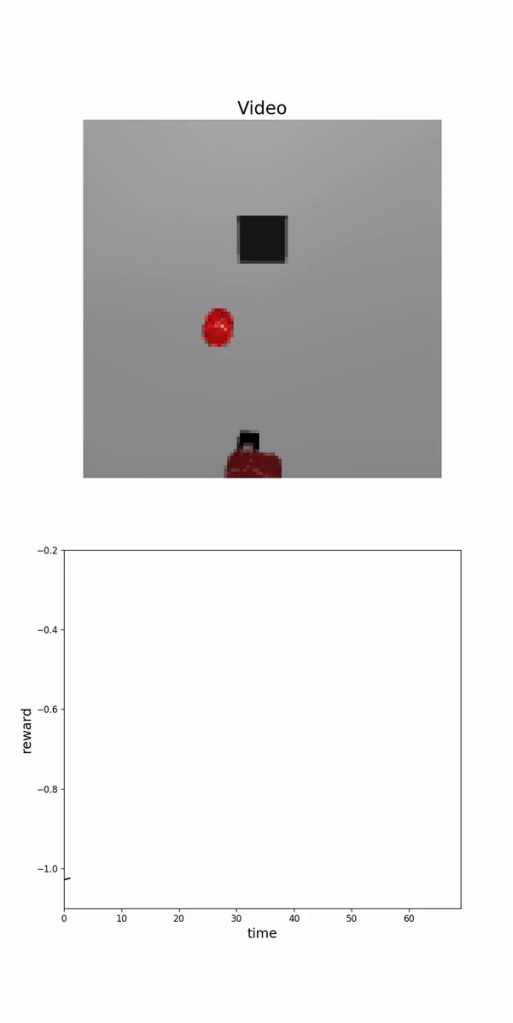
Kevin Zakka[1,3*], **Andy Zeng**[2], **Pete Florence**[2], **Jonathan Tompson**[2],
Jeannette Bohg[1], **and Debidatta Dwibedi**[2]
[1]Stanford University, [2]Robotics at Google, [3]UC Berkeley

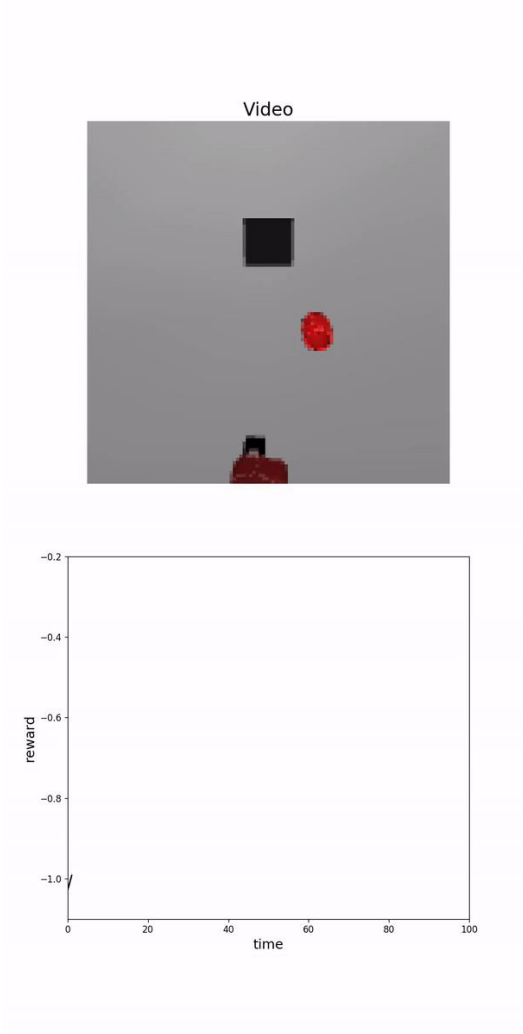


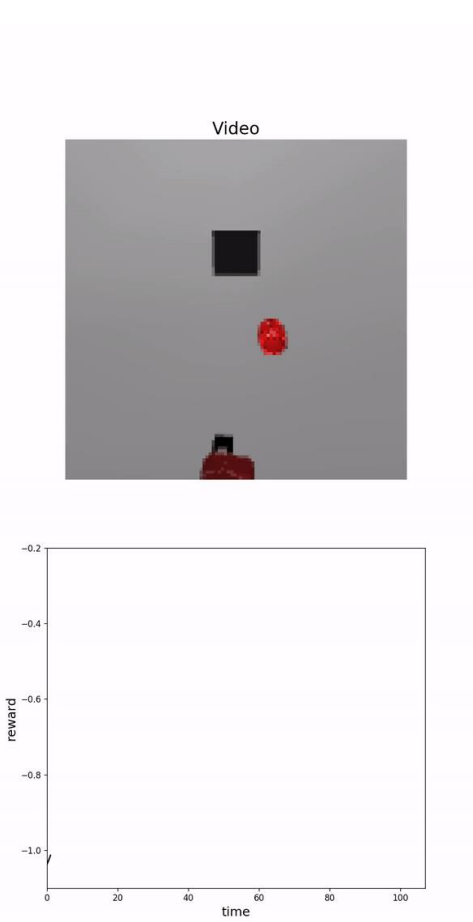
From IRL to RL



The XIRL-learned reward

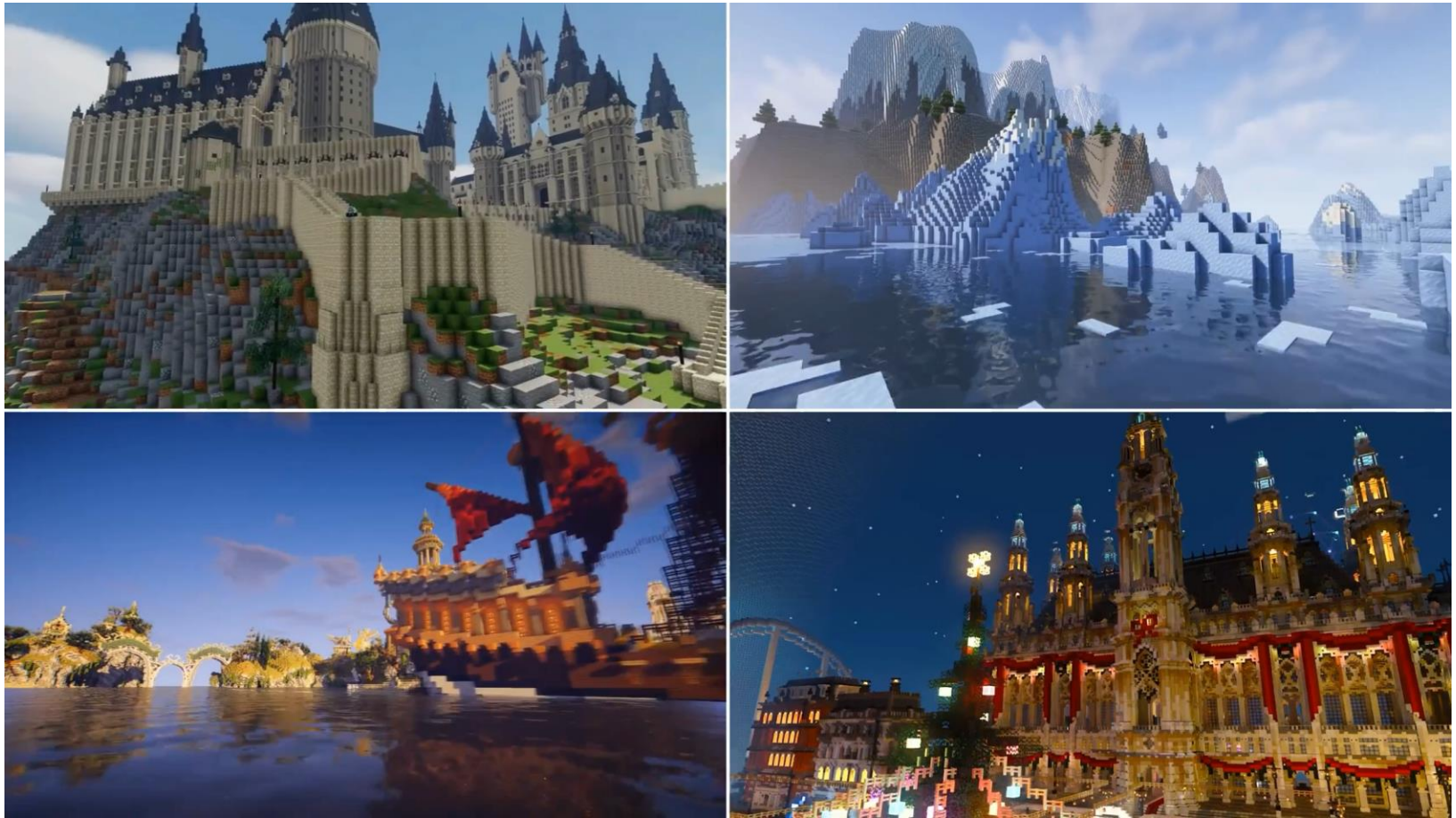




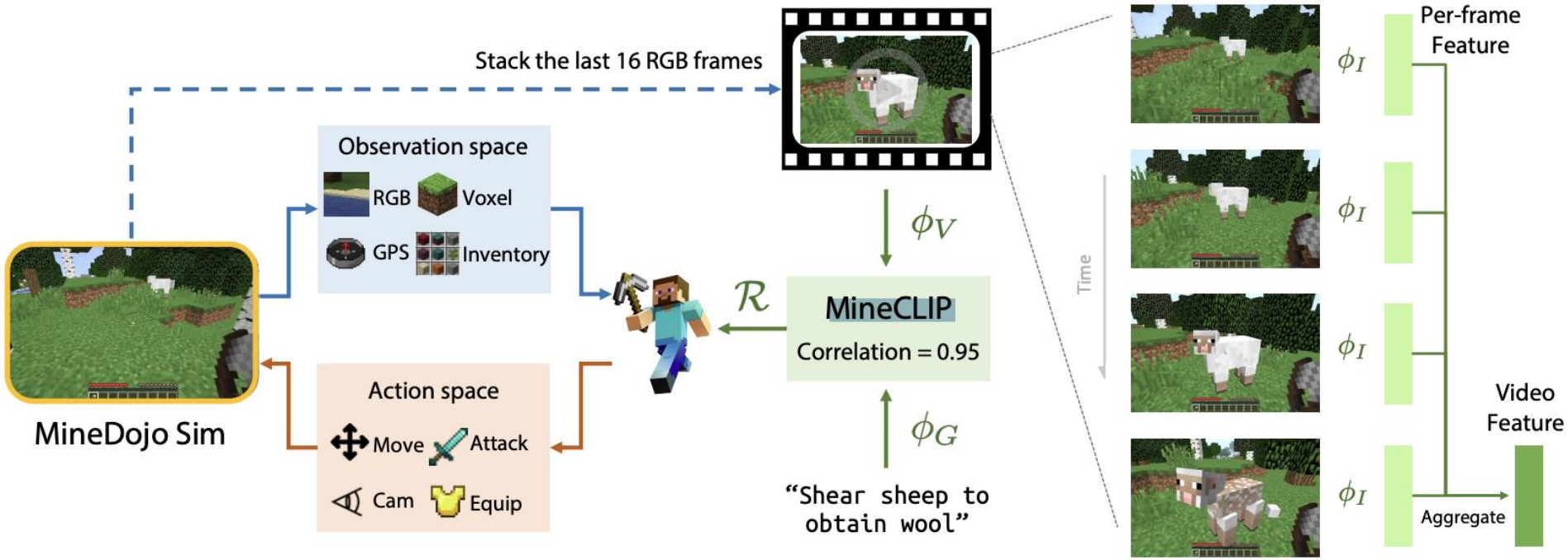


MINEDOJO: Building Open-Ended Embodied Agents with Internet-Scale Knowledge

Linxi Fan[1], **Guanzhi Wang**[2*], **Yunfan Jiang**[3*], **Ajay Mandlekar**[1], **Yuncong Yang**[4], **Haoyi Zhu**[5], **Andrew Tang**[4], **De-An Huang**[1], **Yuke Zhu**[1 6†], **Anima Anandkumar**[1 2†]
[1]NVIDIA, [2]Caltech, [3]Stanford, [4]Columbia, [5]SJTU, [6]UT Austin
[*]Equal contribution [†]Equal advising
https://minedojo.org



Learning Rewards from Language



Thank you!