Generative World Models: The Dreamer Models

Sanjiban Choudhury

Cornell Bowers CIS
Computer Science

The story so far ...

Robots have to act in the world

Hence, we learned various algorithms for decision making

But we assumed that we can observe the "state"

The story so far ...

But in the real world, no one tells you the "state"

All you see are observations

How do we learn from observations?

The story so far ...

Our focus in this and future lectures will turn to learning representations

Models.

What is a model?

s_t , a_t

Model

What is a model?

$s_t\ ,\ a_t$

$\downarrow$

$\boxed{\text{Model}}$

$\downarrow$

s_{t+1}

What is a model?

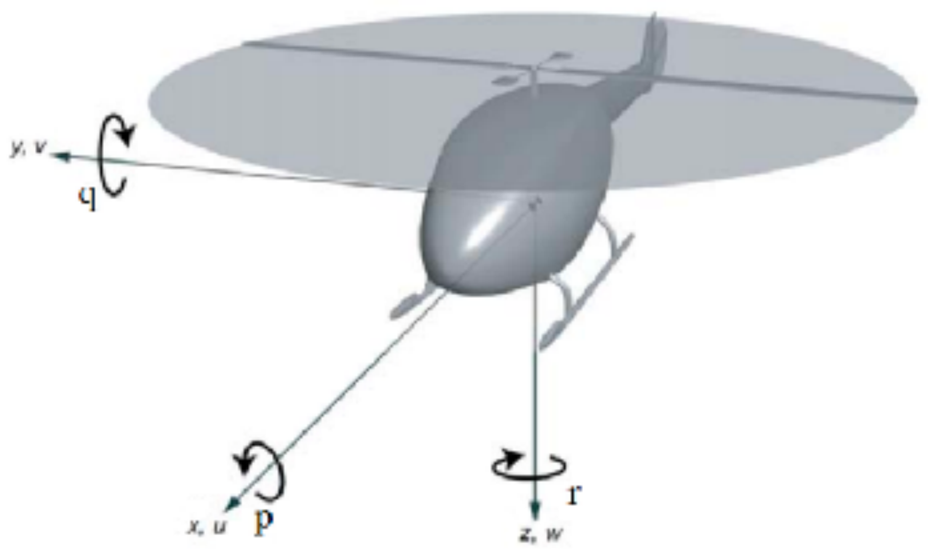
$$P_\theta(s_{t+1} \mid s_t, a_t)$$

Learning Models

Models: From Simple to Complex

Simple Complex

Models: From Simple to Complex



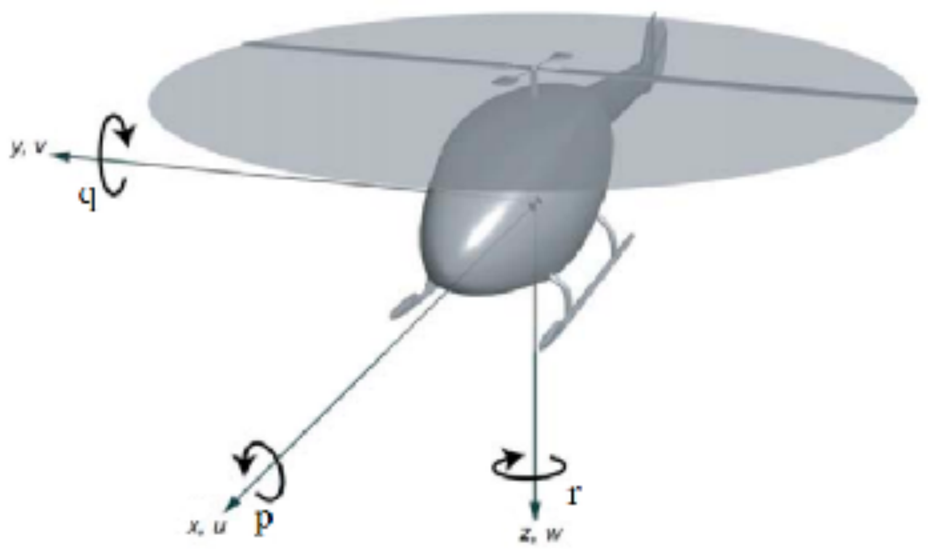
Physics Models

Simple

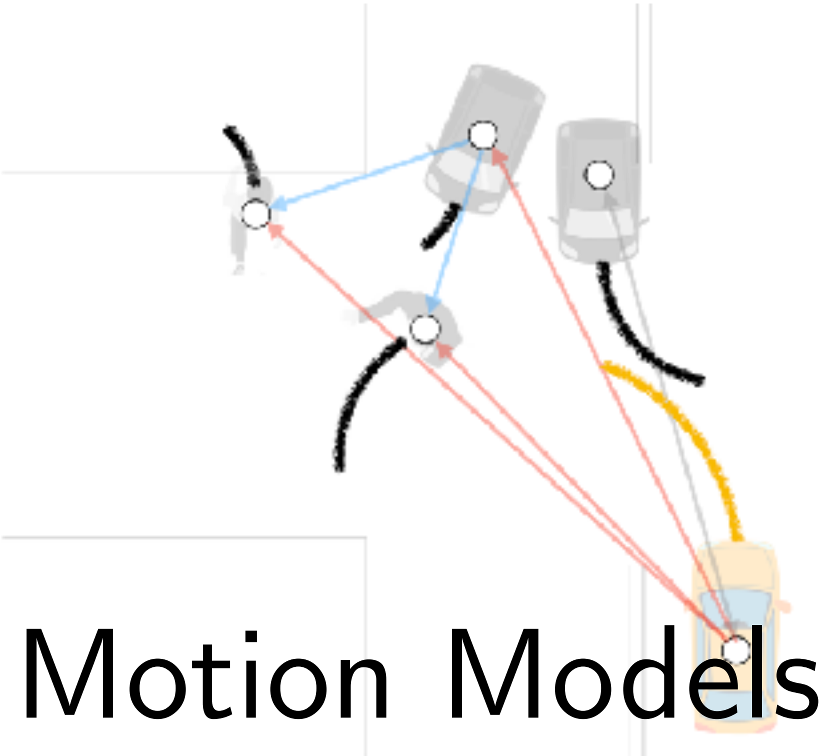
Known state

Strong prior
on dynamics

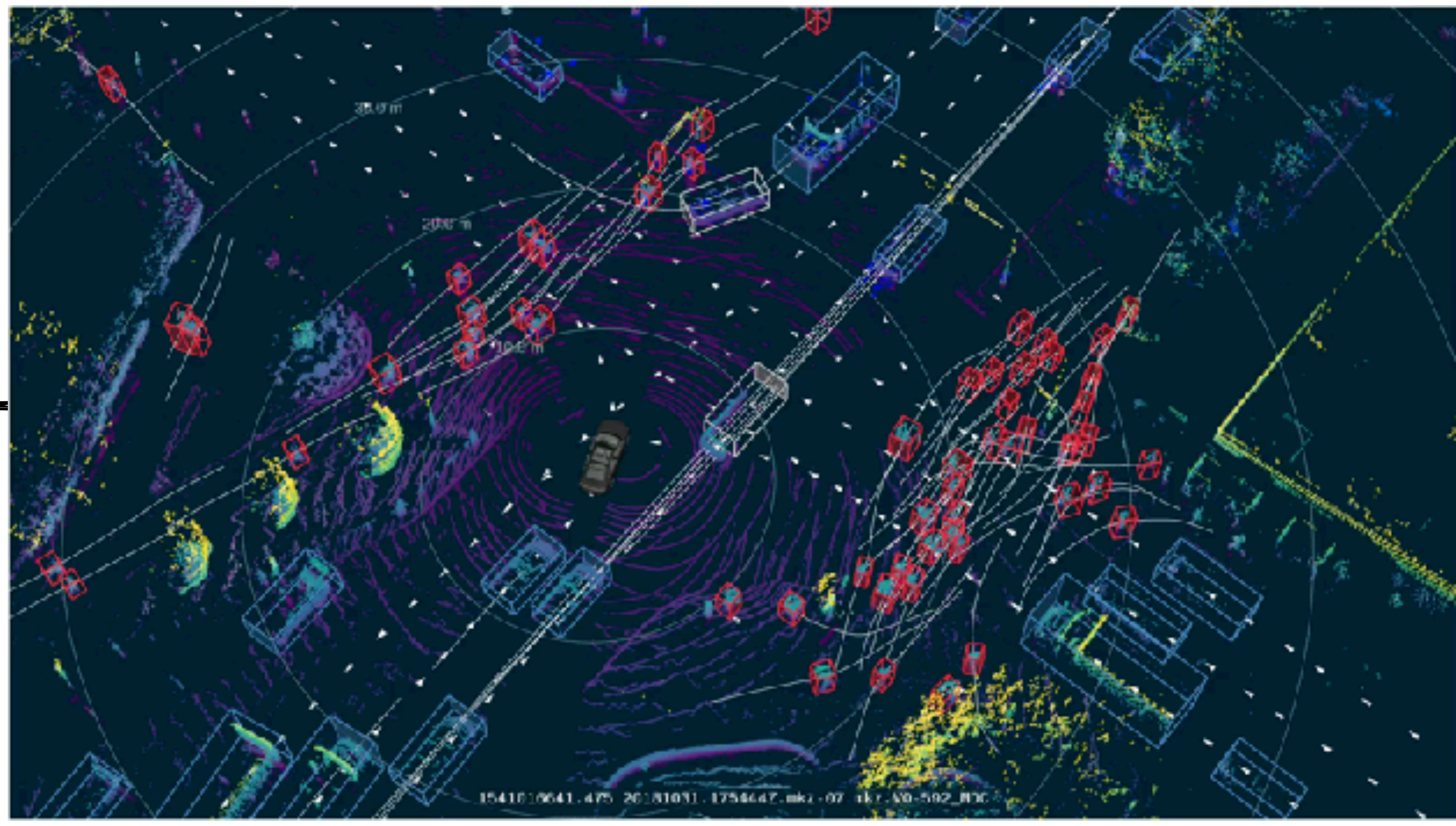
Models: From Simple to Complex



Physics Models



Motion Models



Simple

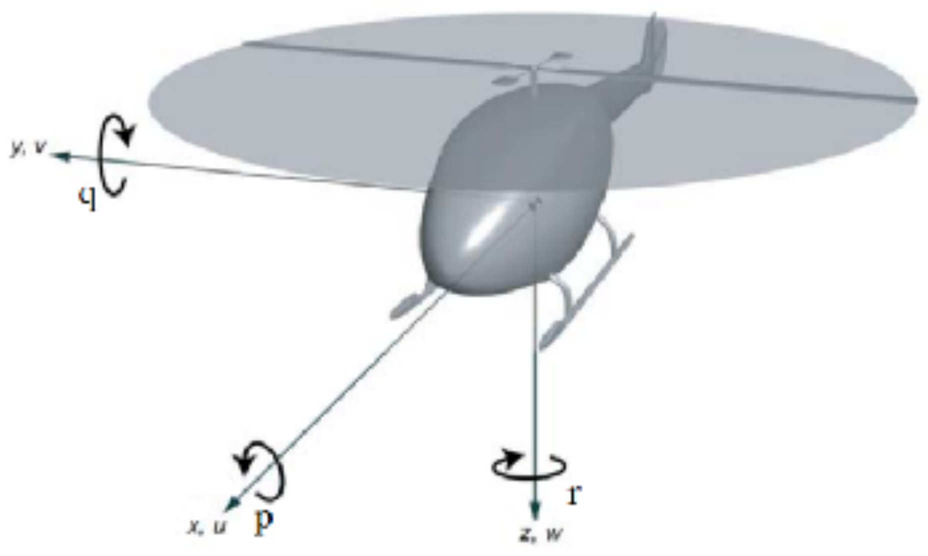
Known state

Strong prior
on dynamics

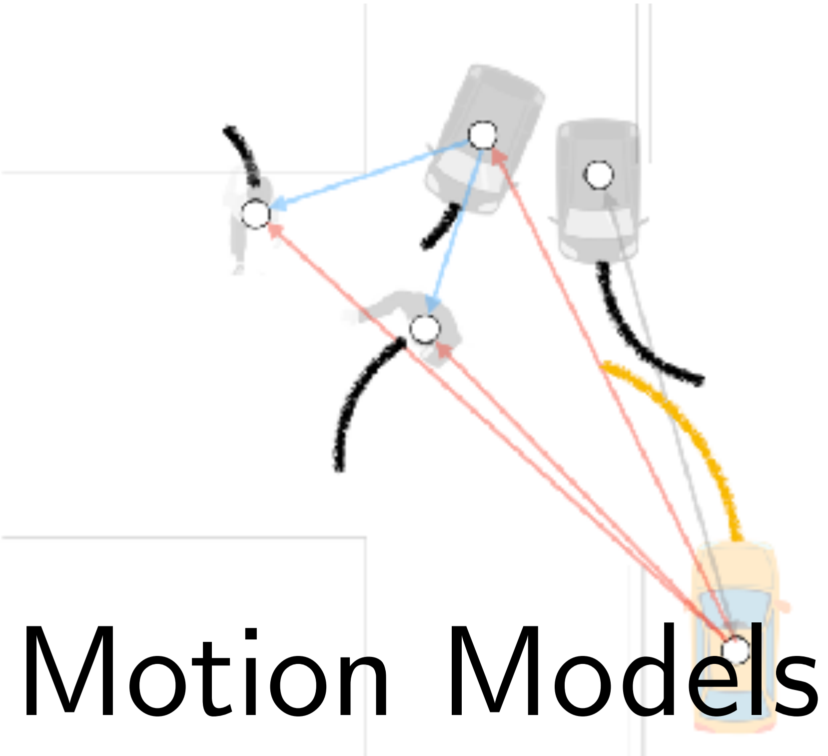
Known state

Unknown
dynamics

Models: From Simple to Complex



Physics Models



Motion Models

Open World Models

Simple → Complex

Physics Models	Motion Models	Open World Models
Known state	Known state	Unknown state
Strong prior on dynamics	Unknown dynamics	Unknown dynamics

Activity!

Modelling Tamago Sushi

Think-Pair-Share!

Think (30 sec): How would you model making tamago sushi?

Pair: Find a partner

Share (45 sec): Partners exchange ideas

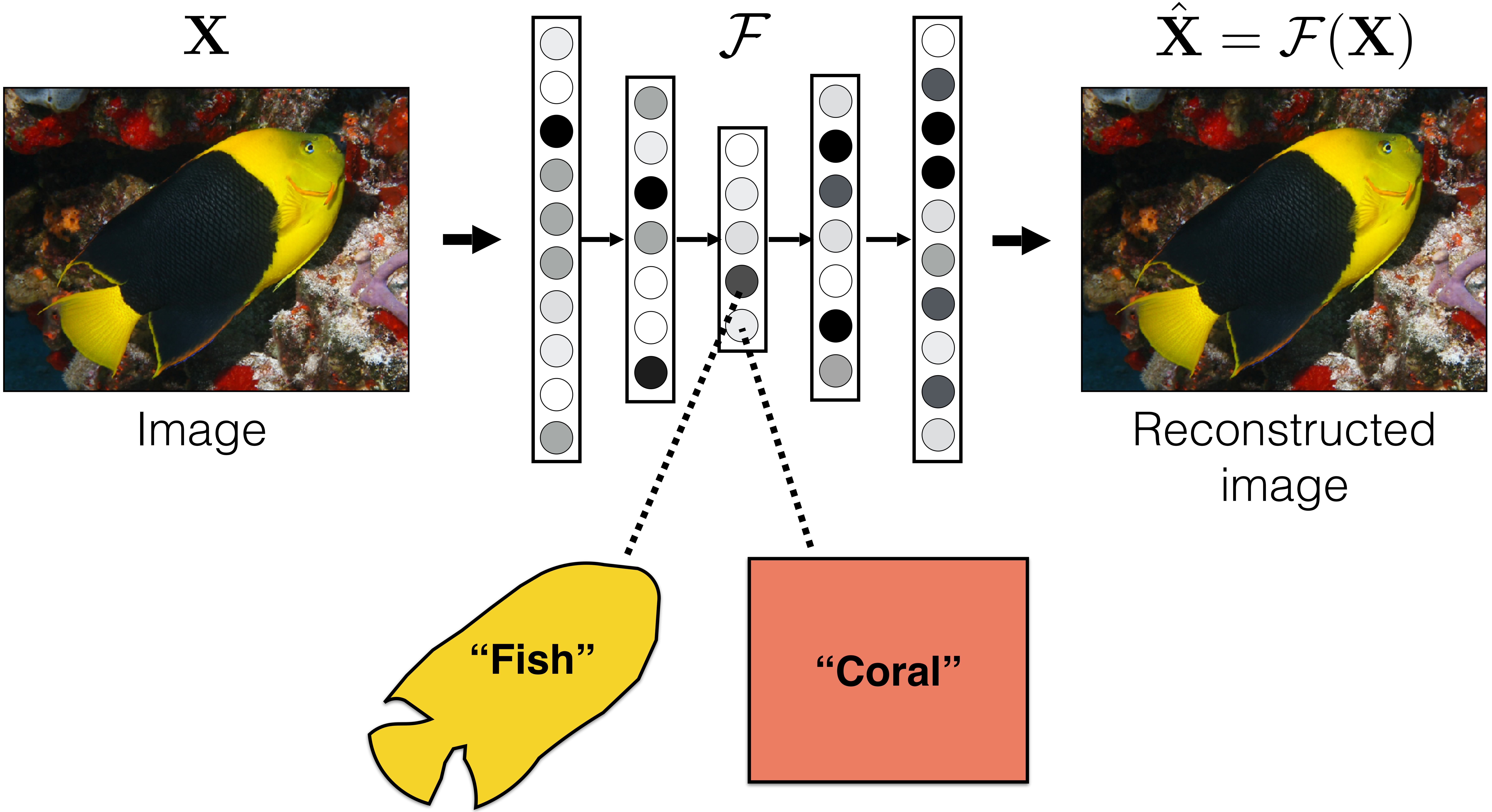
Challenges with learning complex models

Challenge 1: Can't see state, only get high-dimensional observations

Challenge 2: Planning with complex dynamics

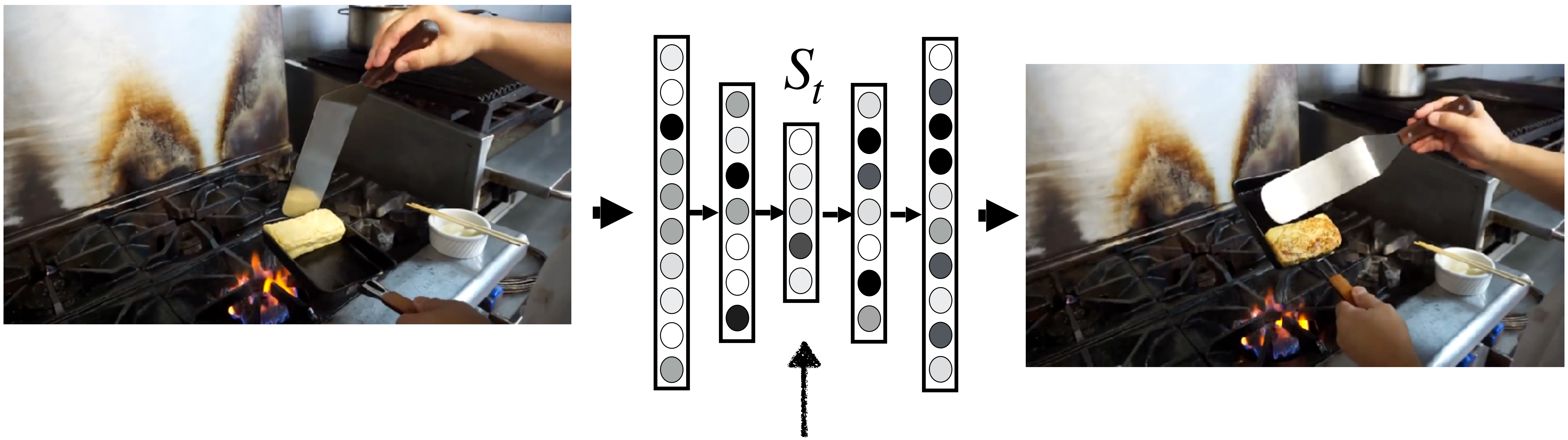
How can we learn latent low-dimensional state from high-dimensional observations?

Idea: Use “auto-encoder” trick from computer vision



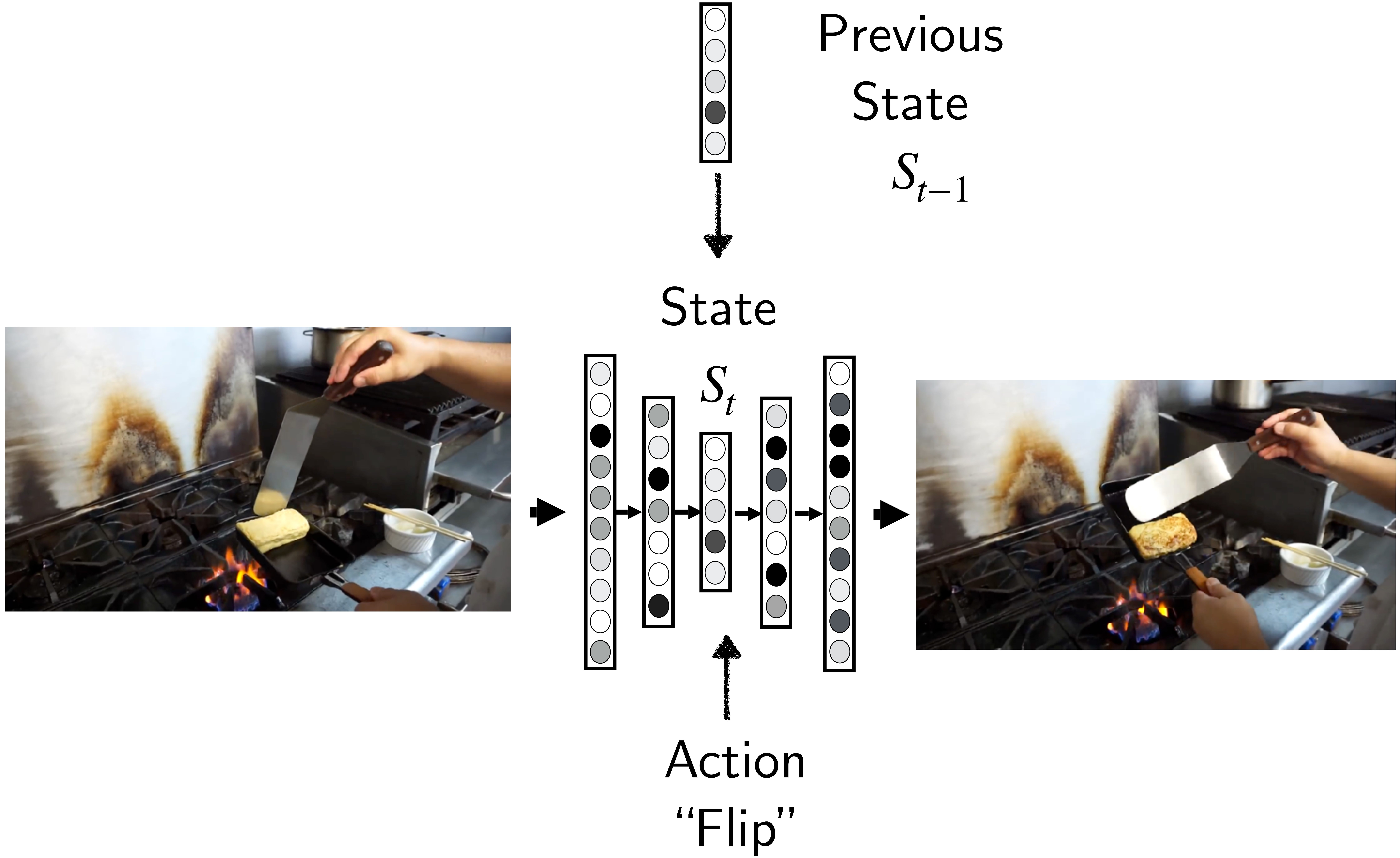
From MIT 6.8300/6.8301: Advances in Computer Vision

State



Action

“Flip”


Previous
State
S_{t-1}
State
S_t
Action
“Flip”

The DREAMER Algorithms

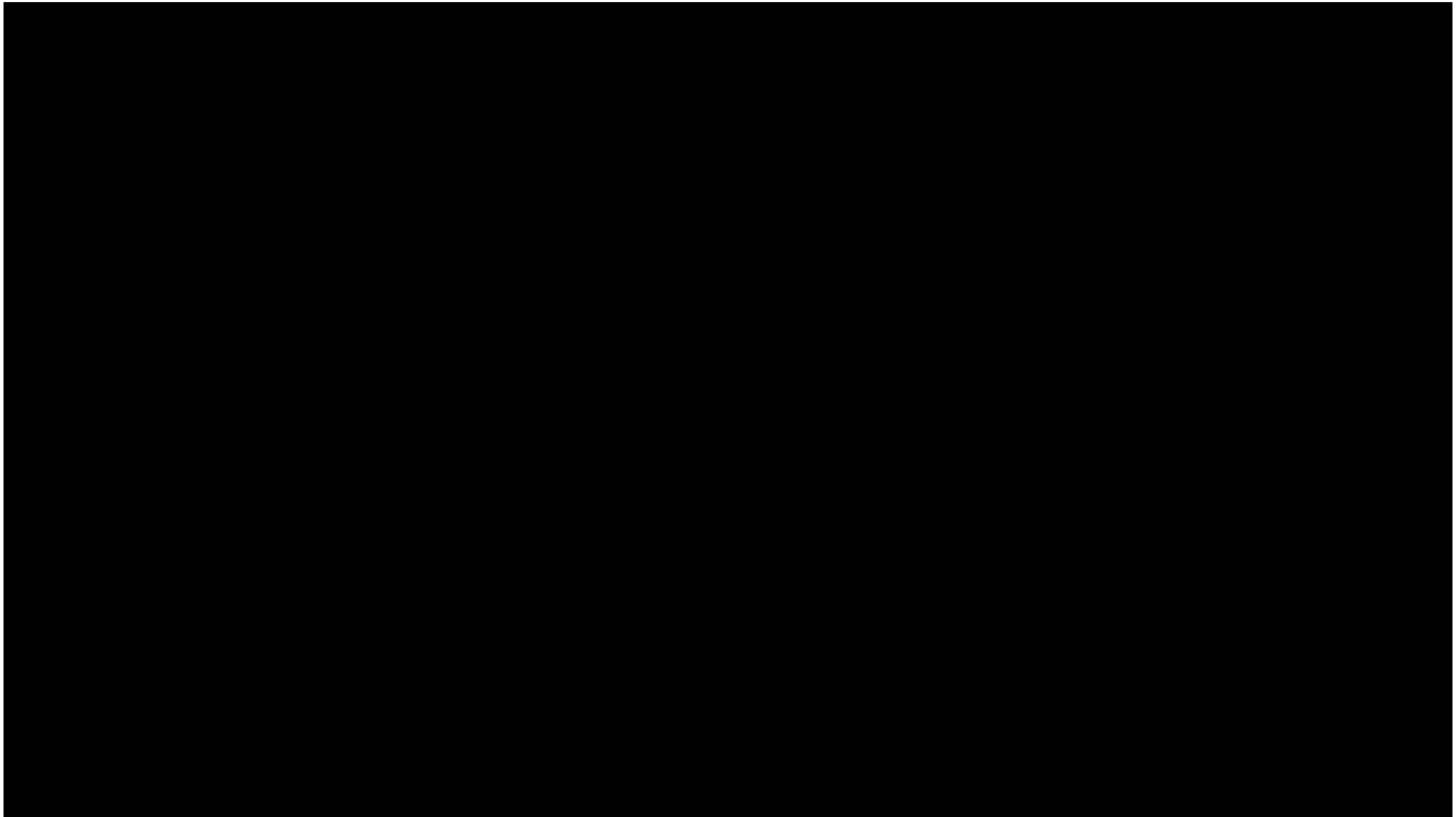
Mastering Diverse Domains through World Models

Danijar Hafner,[1,2] Jurgis Pasukonis,[1] Jimmy Ba,[2] Timothy Lillicrap[1]

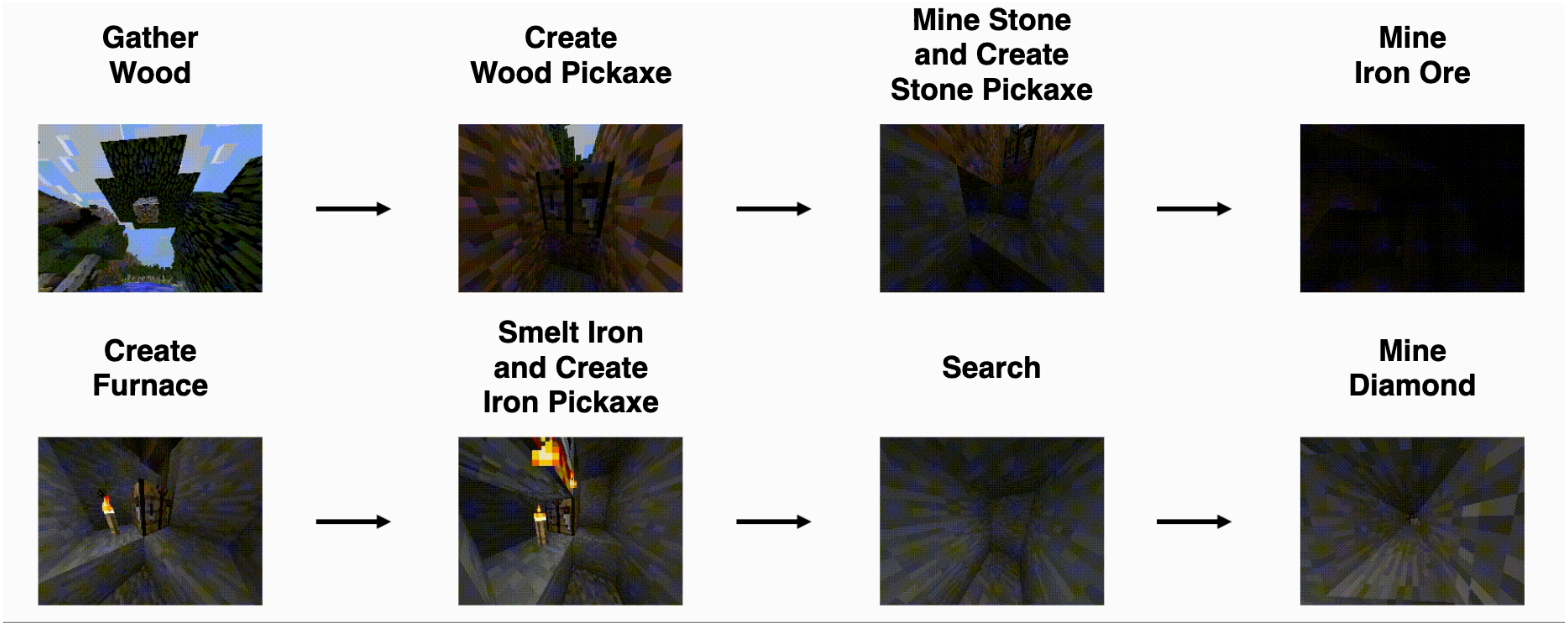
[1]DeepMind [2]University of Toronto

2023

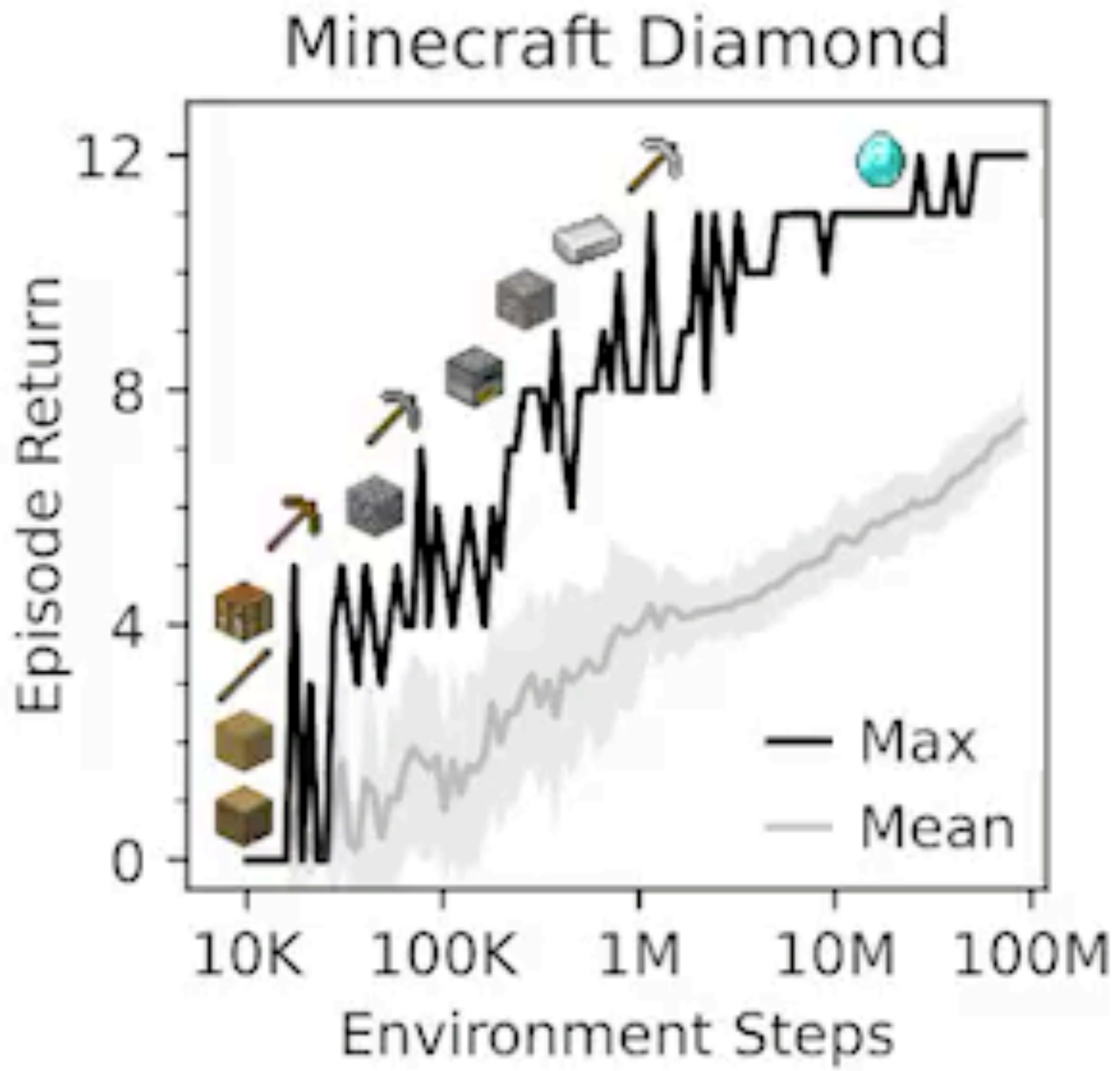
MineRL Diamond Challenge

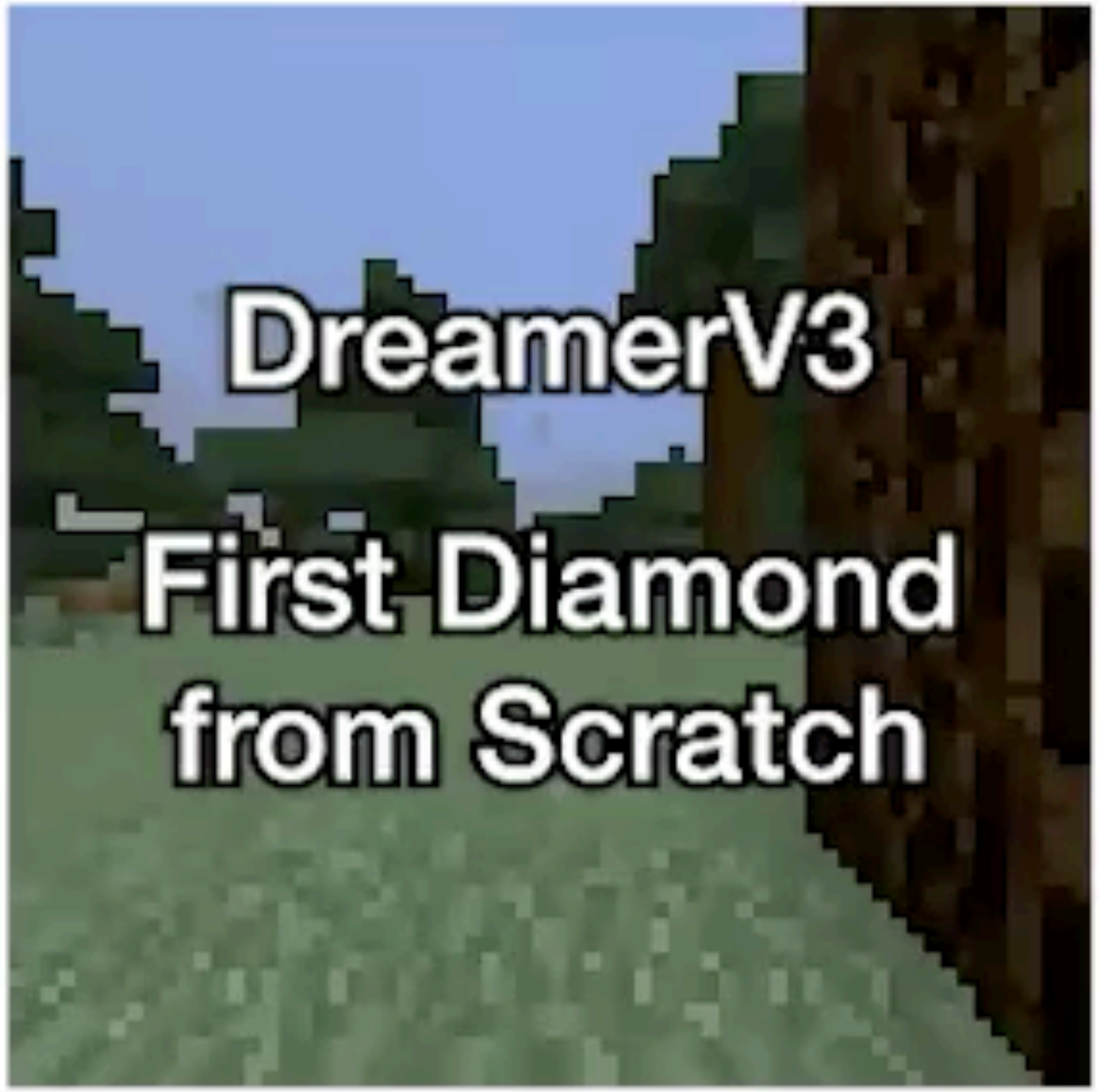


MineRL Diamond Challenge



DreamerV3 solved this task!





The DREAMER Algorithm

DREAM TO CONTROL: LEARNING BEHAVIORS BY LATENT IMAGINATION

Danijar Hafner *
University of Toronto
Google Brain

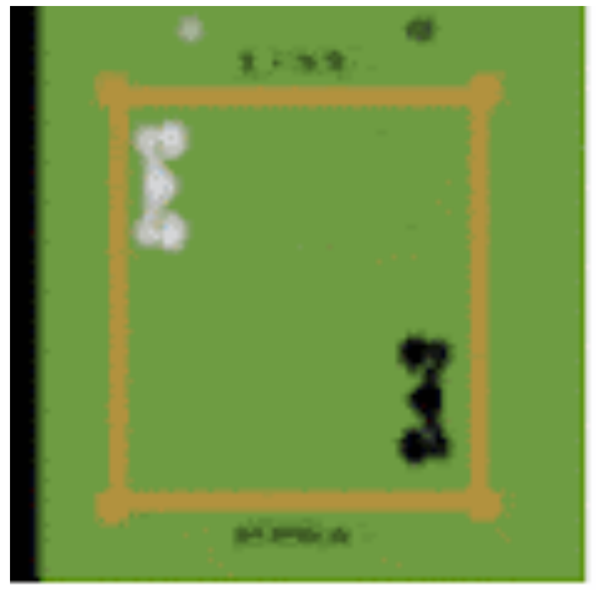
Timothy Lillicrap
DeepMind

Jimmy Ba
University of Toronto

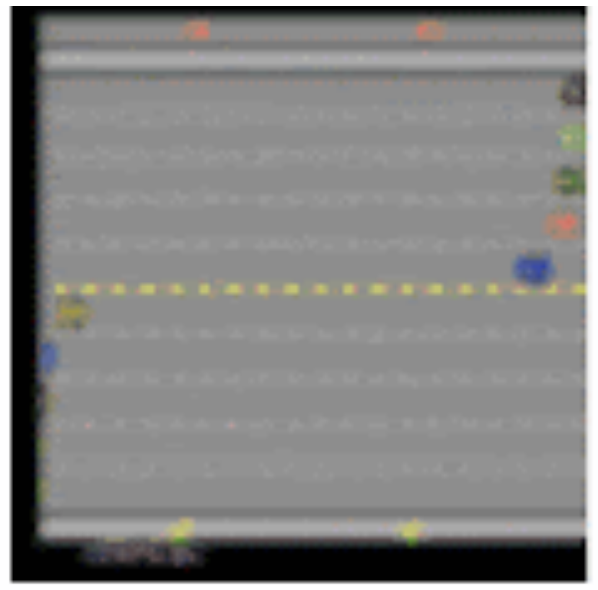
Mohammad Norouzi
Google Brain

2020

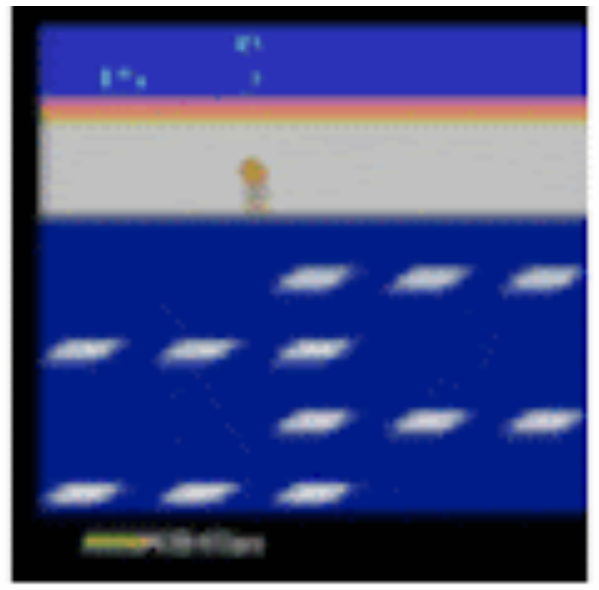
Look at the videos below



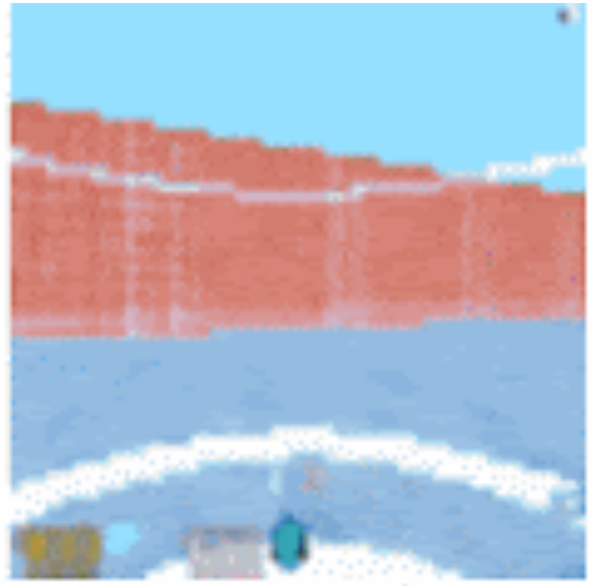
Boxing



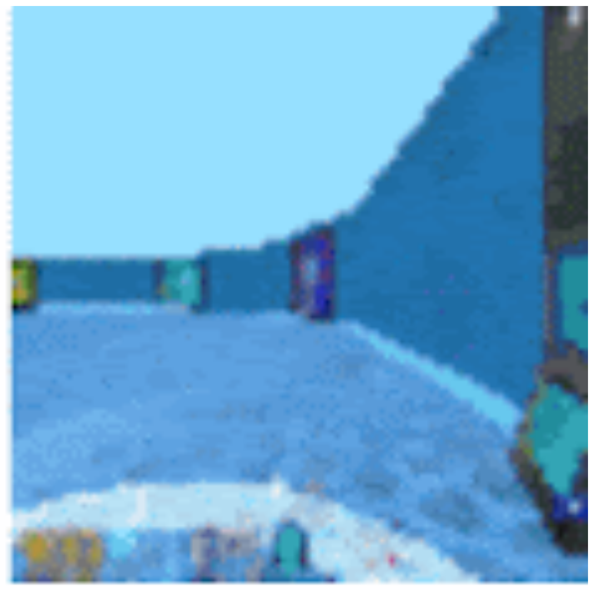
Freeway



Frostbite



Collect Objects



Watermaze

Sparse Cartpole

Acrobot Swingup

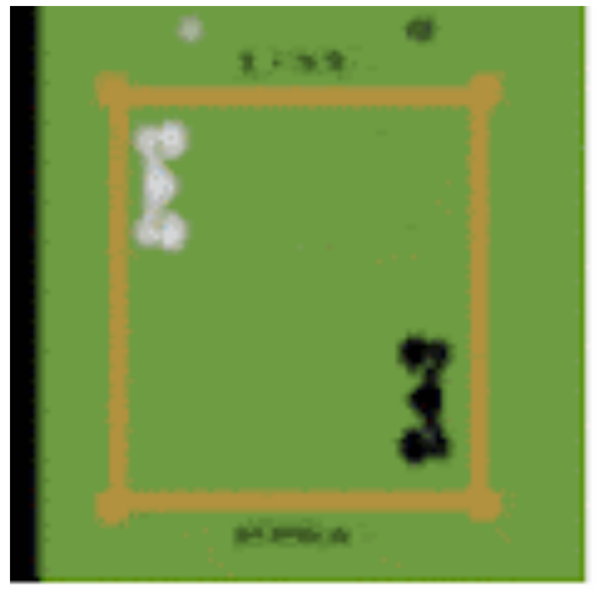
Hopper Hop

Walker Run

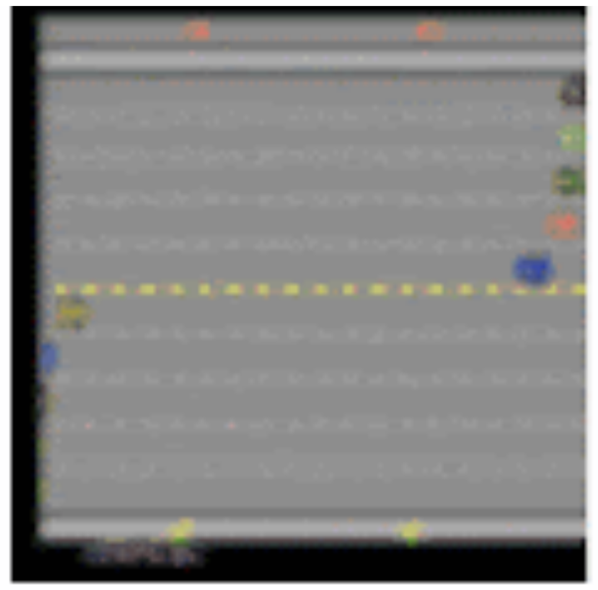
Quadruped Run

Is this from the actual simulator or predictions made by a model?

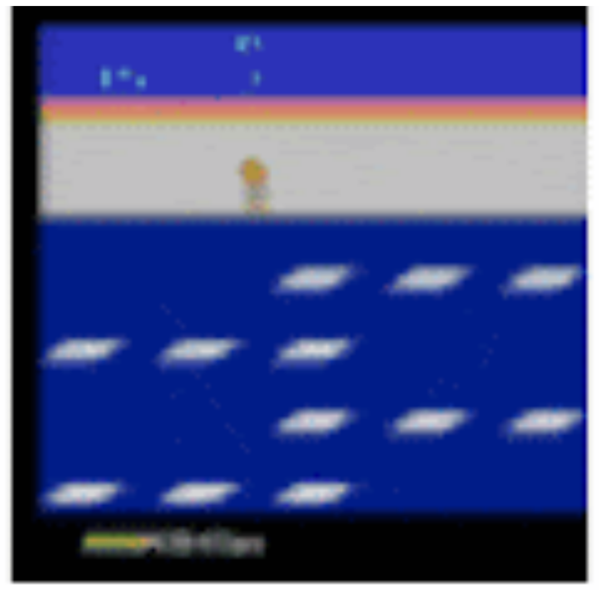
Look at the videos below



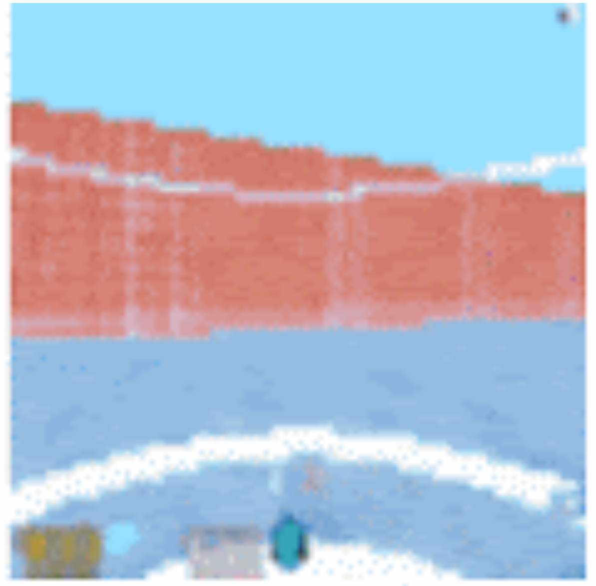
Boxing



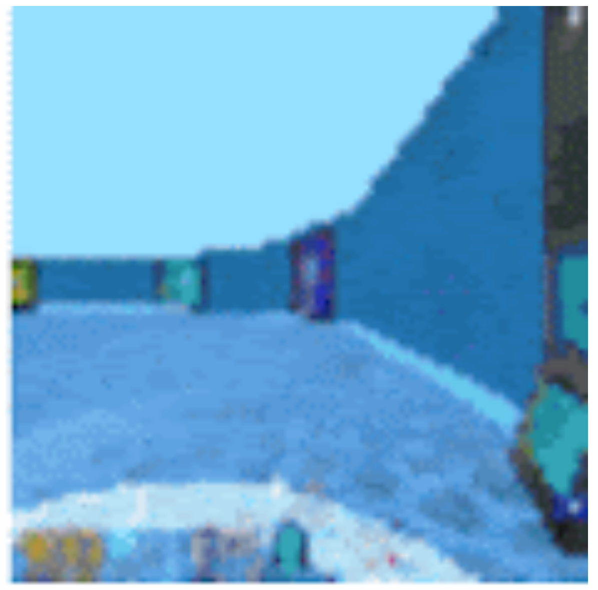
Freeway



Frostbite



Collect Objects



Watermaze

Sparse Cartpole

Acrobot Swingup

Hopper Hop

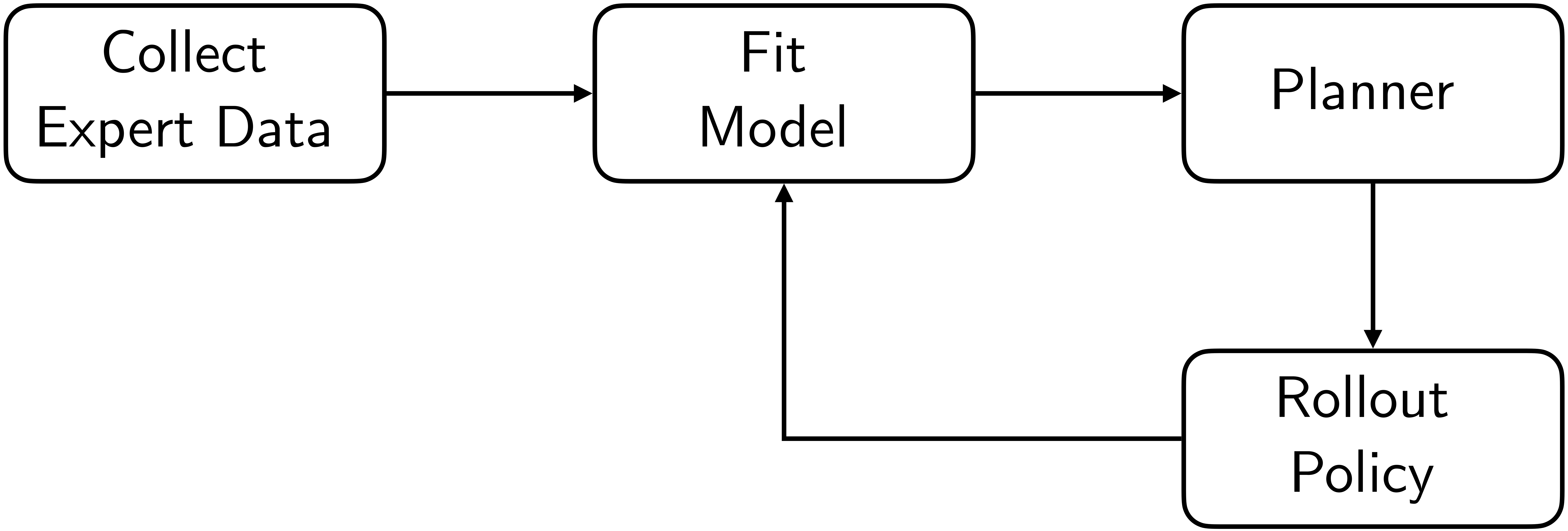
Walker Run

Quadruped Run

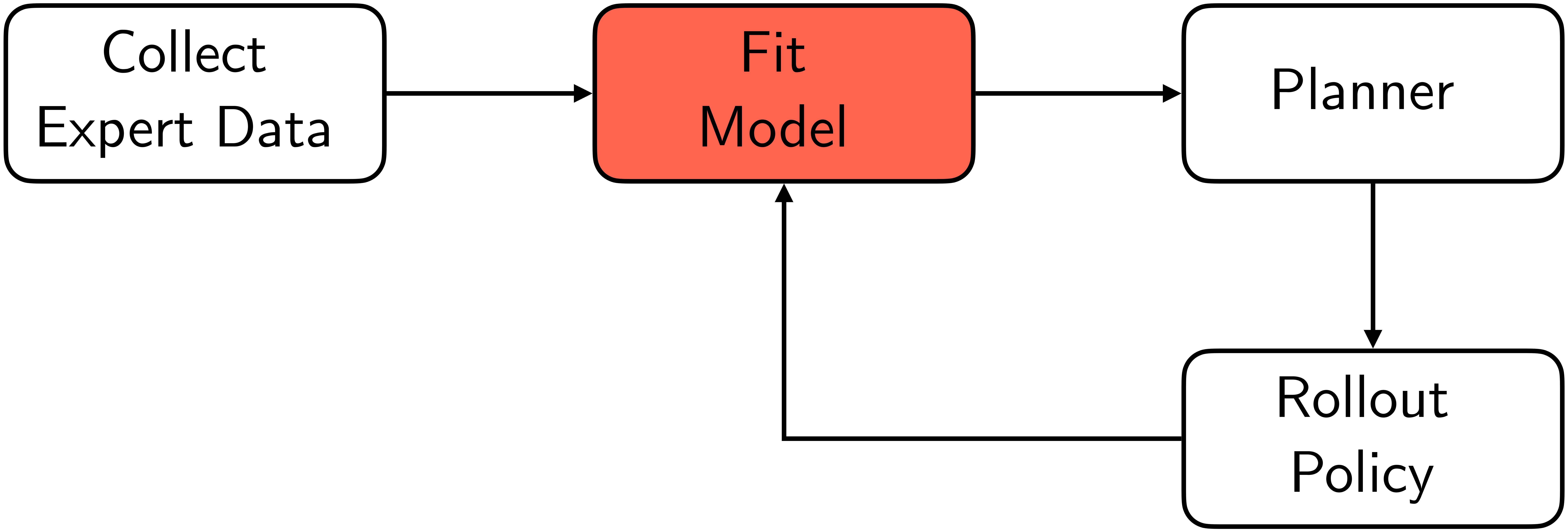
Predictions by a model!

Recap: Model-based RL

(Ross & Bagnell, 2012)



How does DREAMER fit a model?



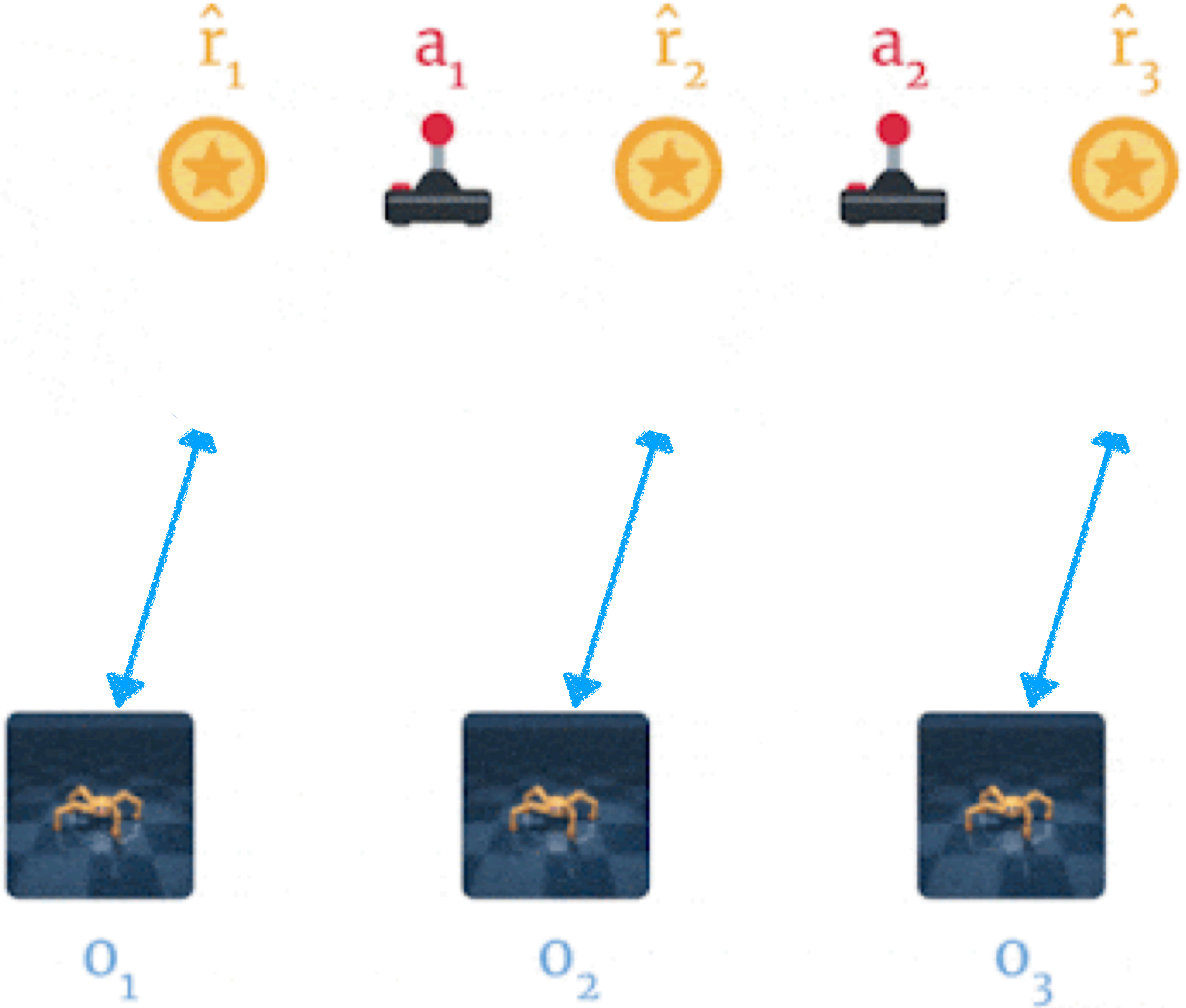
Goal: Fit a Model given data

Given Data:
Observations, rewards,
actions

Goal: Fit a Model given data

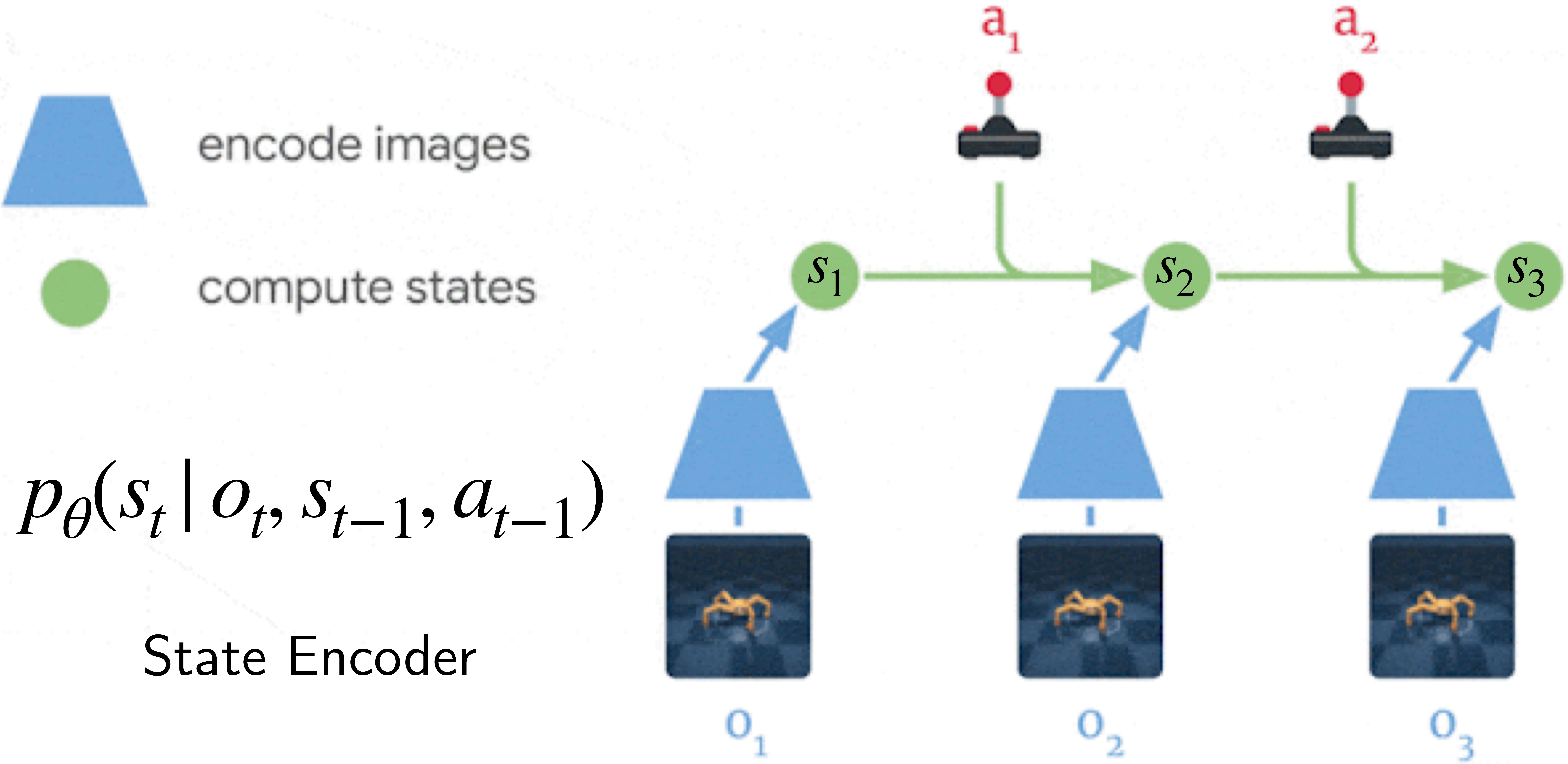
Given:
Observations, rewards, actions

Predict:
States,
Dynamics Function,
Reward Function



Actions

Observations



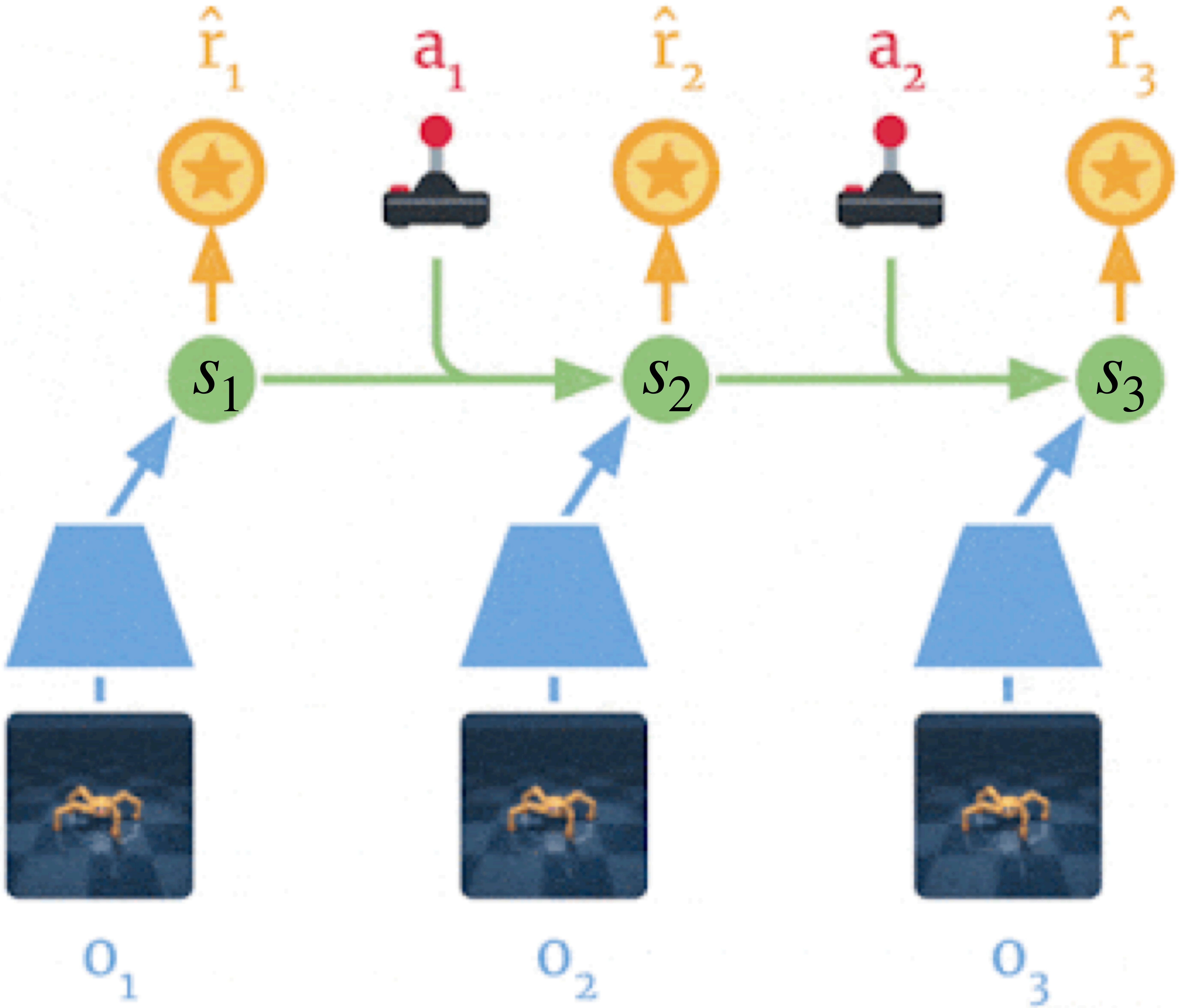
$$p_\theta(s_t \mid o_t, s_{t-1}, a_{t-1})$$

State Encoder

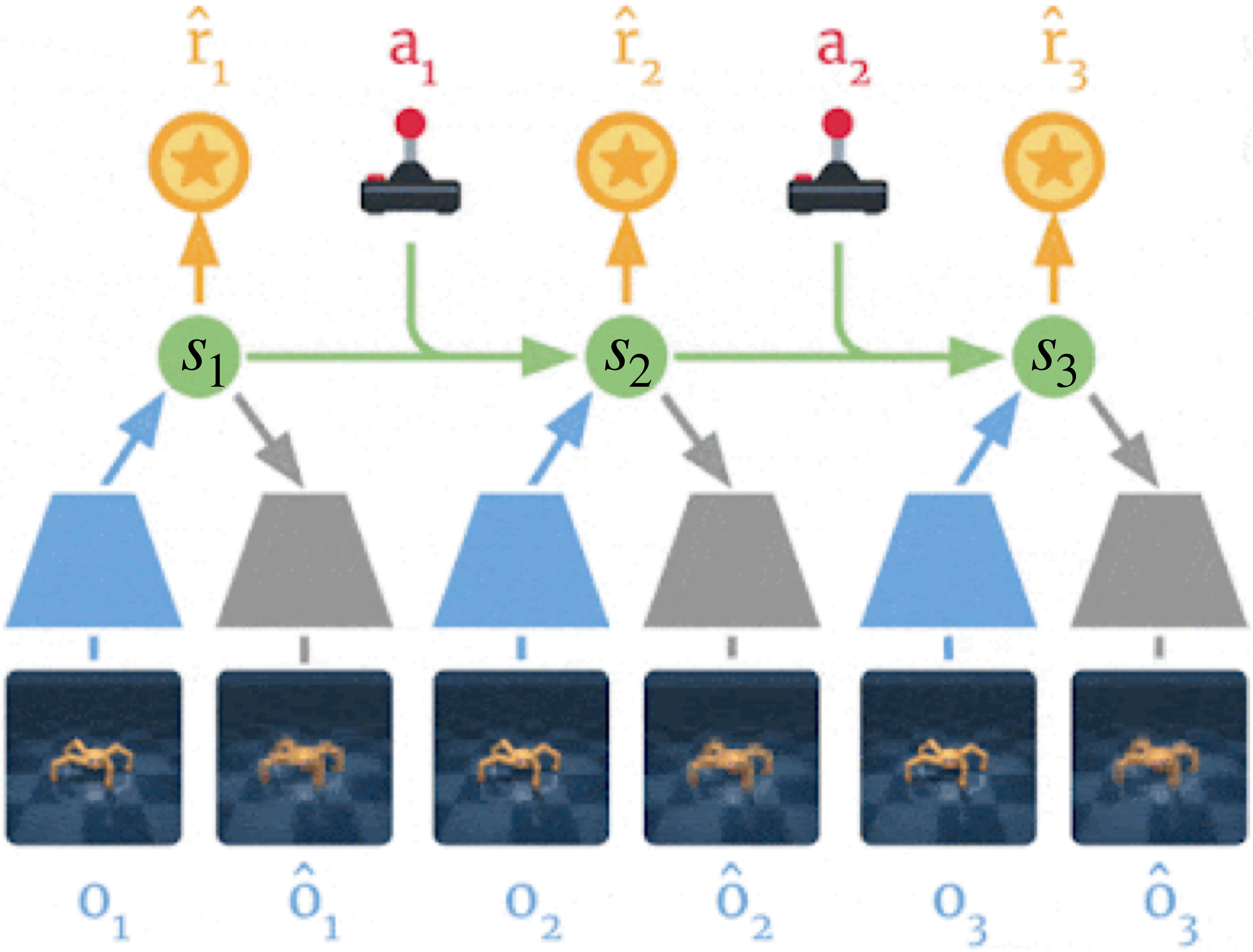
$$\ell = (r_t - \hat{r}_t)^2$$

$$q_\theta(r_t \mid s_t)$$

Reward Decoder

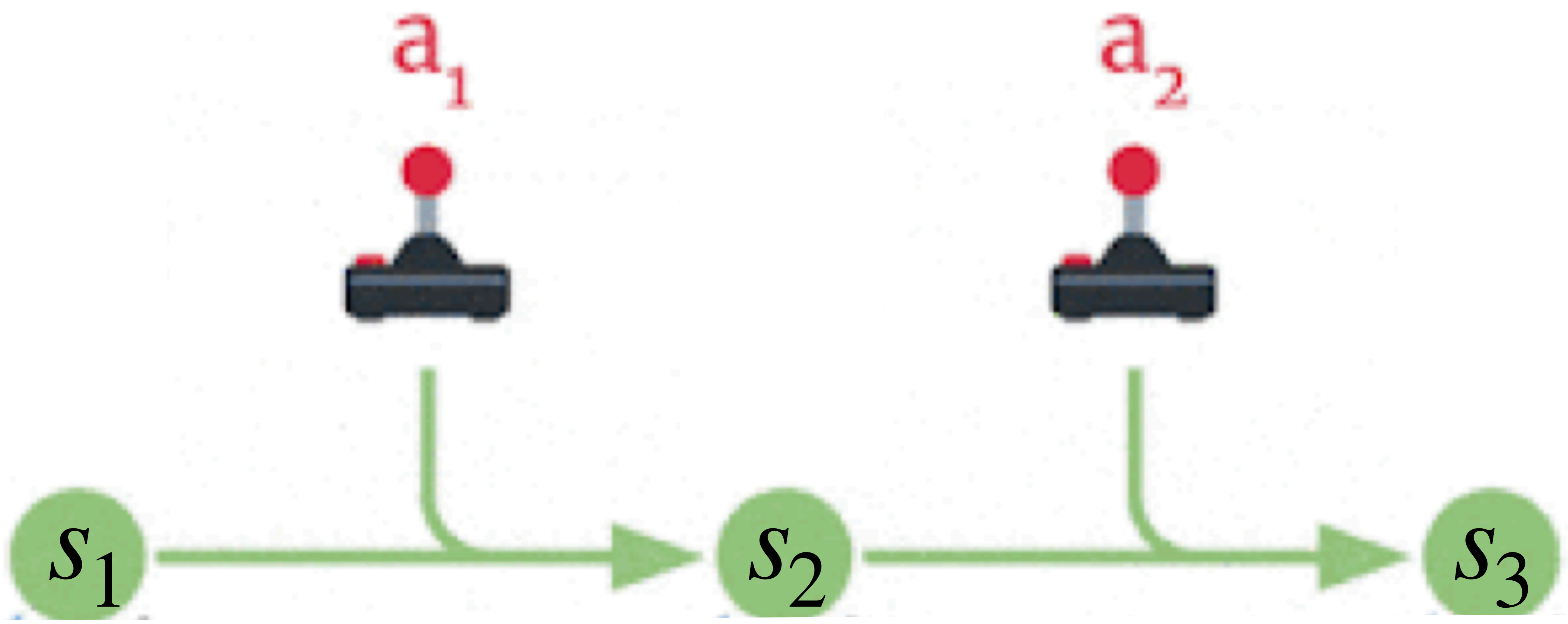


$$\ell = (o_t - \hat{o}_t)^2$$



$$q_\theta(o_t \mid s_t)$$

Observation Decoder



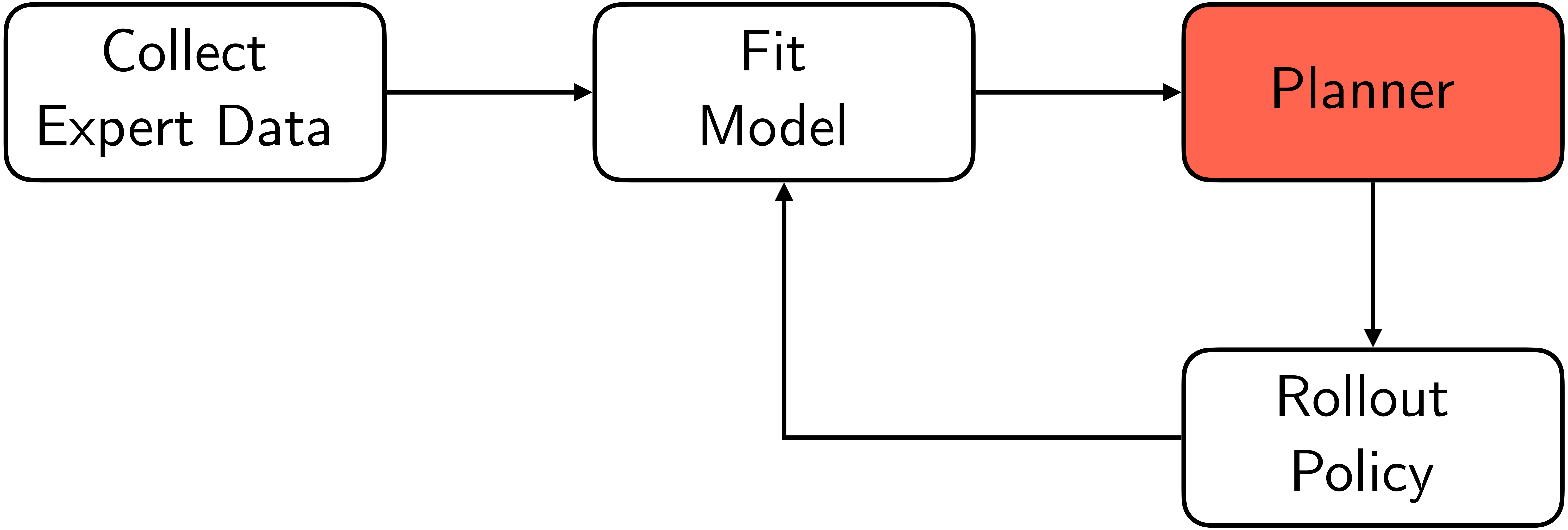
$$q_\theta(s_t \mid s_{t-1}, a_{t-1})$$

Dynamics
Function

Results: Learning World Model

Results: Learning World Model

How does DREAMER do planning?



Goal: Learn a Policy using Actor-Critic

$\pi_\phi(a_t \mid s_t)$

Actor

$V_\psi(s_t)$

Critic

From rollouts in the model

$q_\theta(s_t \mid s_{t-1}, a_{t-1})$

Recall: Actor-Critic

Start with an arbitrary initial policy $\pi_\theta(a \mid s)$

while *not converged* **do**

Roll-out $\pi_\phi(a \mid s)$ in the model $q_\theta(s' \mid s, a)$ to collect trajectories $D = \{s^i, a^i, r^i, s_+^i\}_{i=1}^N$

Fit value function $V_\psi(s^i)$ using TD, i.e. minimize $(r^i + \gamma V_\psi(s_+^i) - V_\psi(s^i))^2$

Compute advantage $\hat{A}(s^i, a^i) = r(s^i, a^i) + \gamma V_\psi(s_+^i) - V_\psi(s^i)$

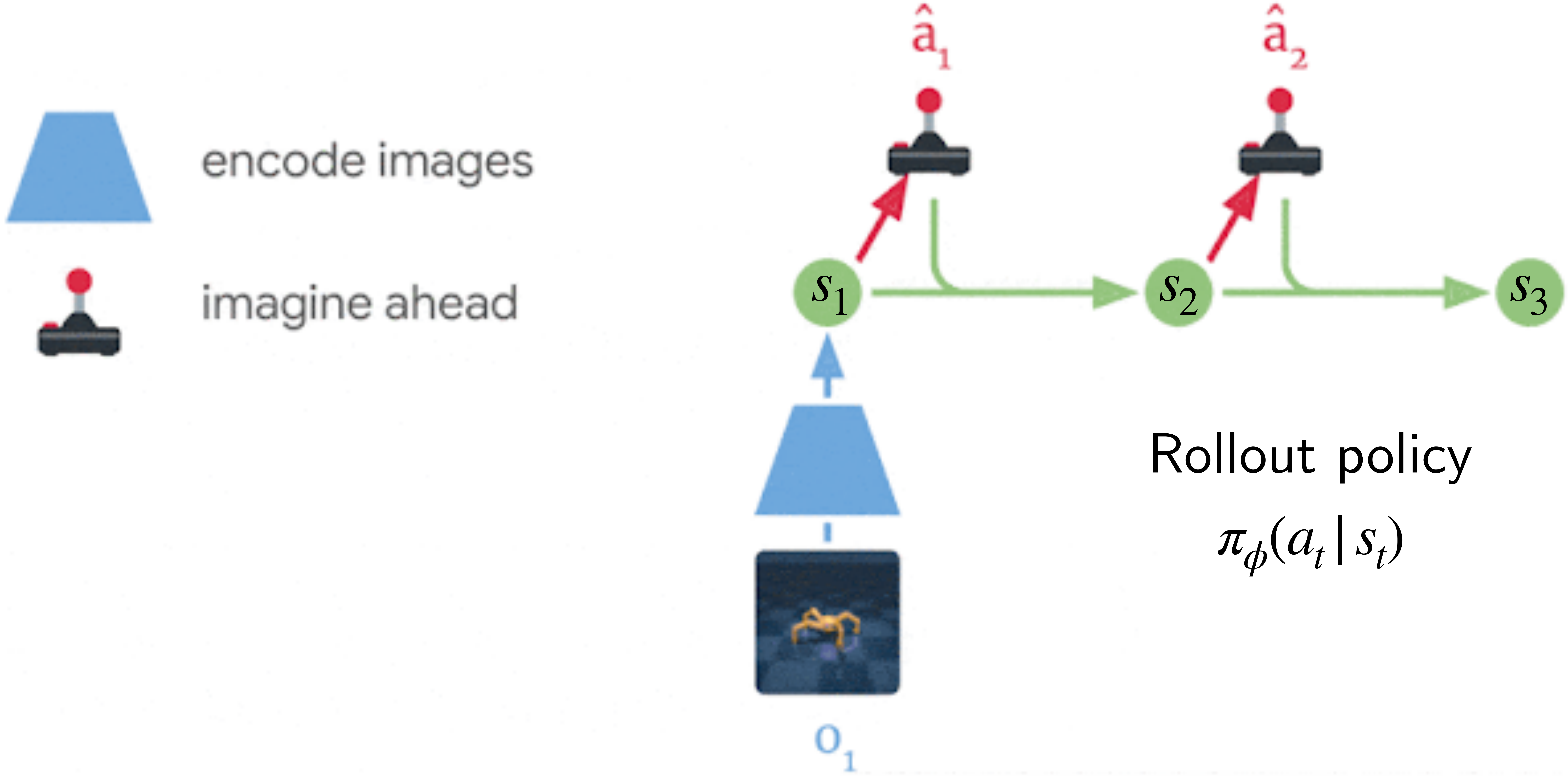
Compute gradient

$$\nabla_\phi J(\phi) = \frac{1}{N}\left[\sum_{t=0}^{T-1} \nabla_\theta \log \pi_\phi(a_t^i \mid s_t^i)\, \hat{A}(s^i, a^i)\right]$$

Update parameters $\phi \leftarrow \phi + \alpha \nabla_\phi J(\phi)$

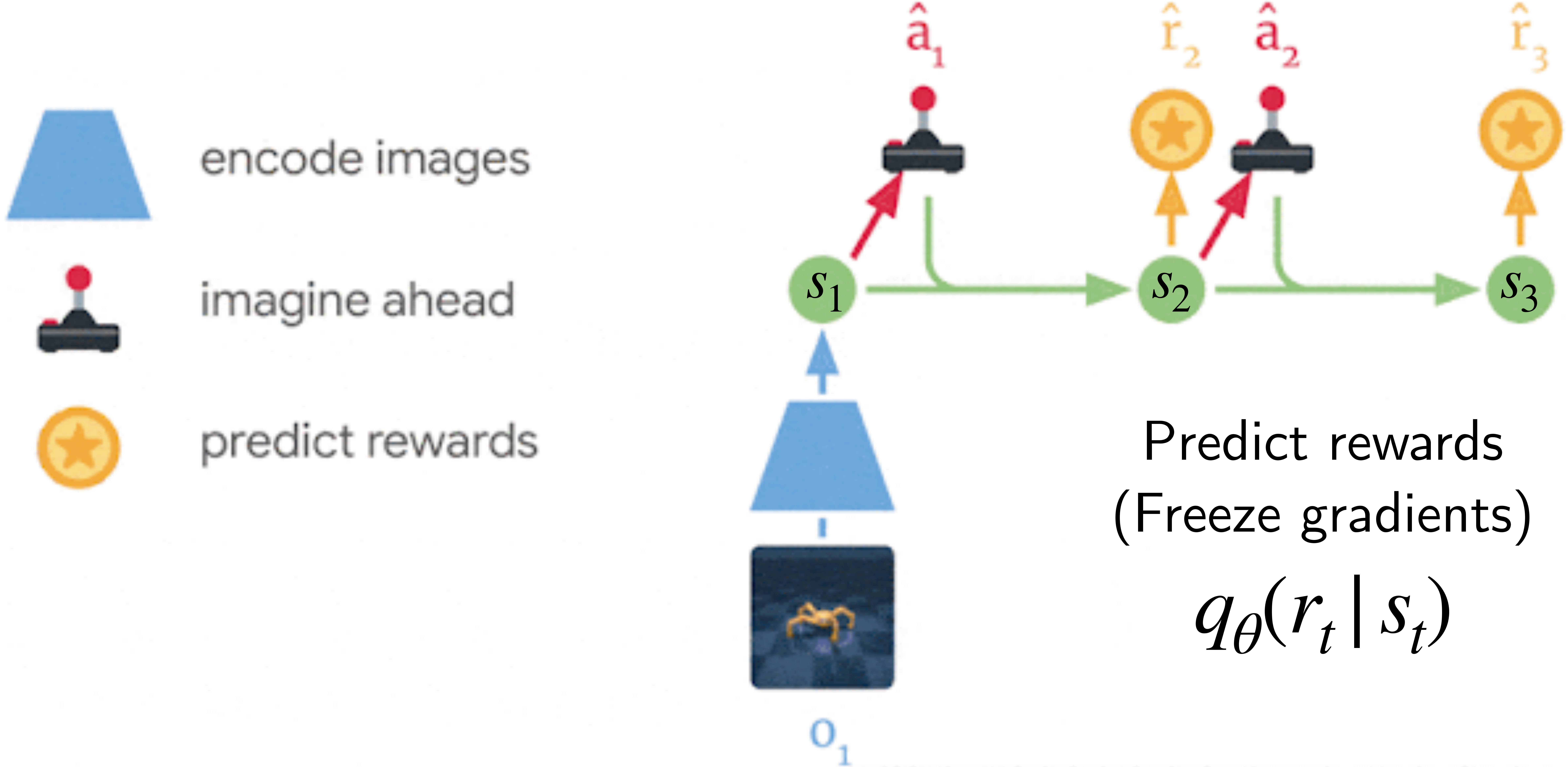
o_1

encode images



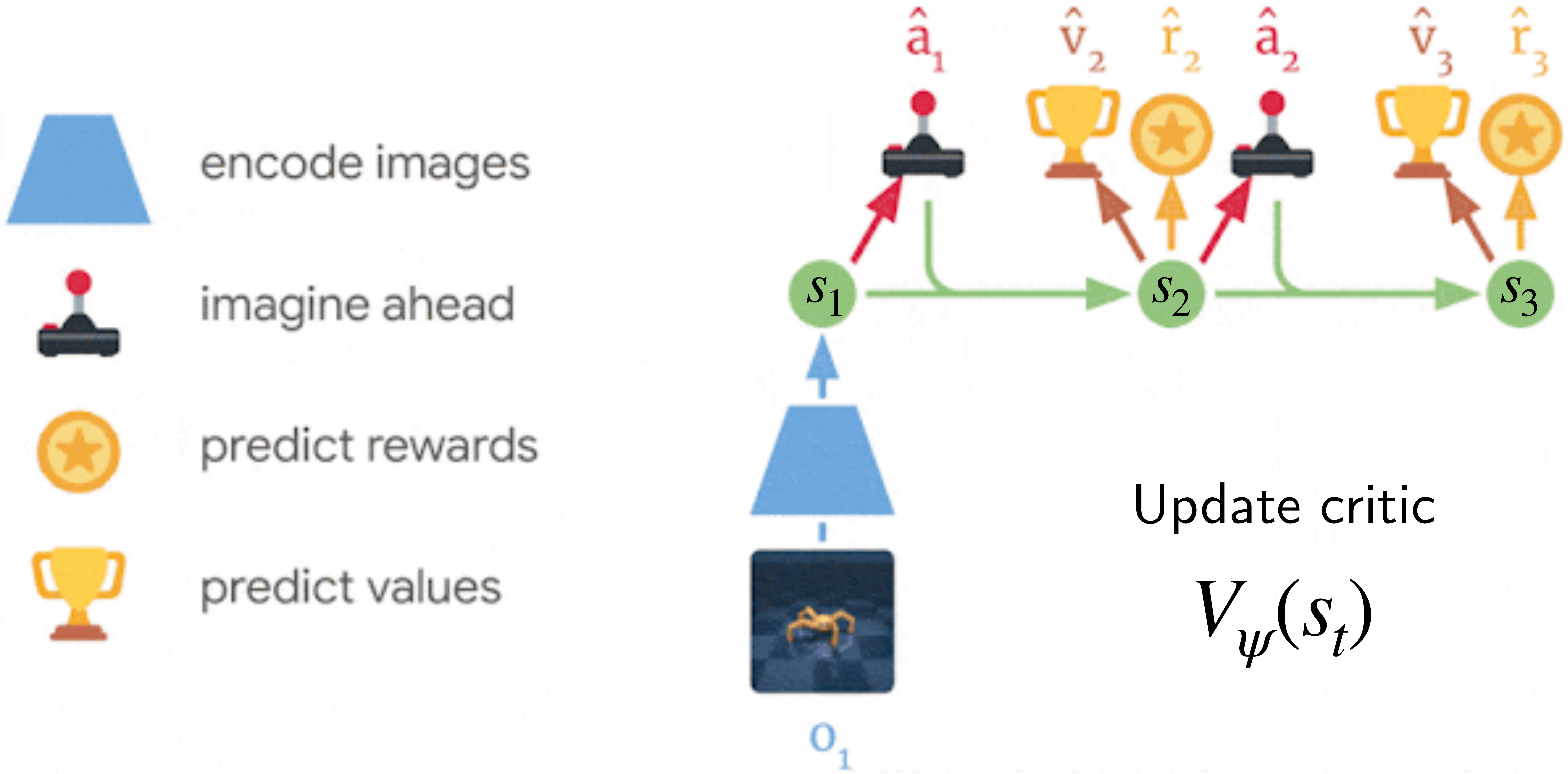
Rollout policy

$$\pi_\phi(a_t | s_t)$$



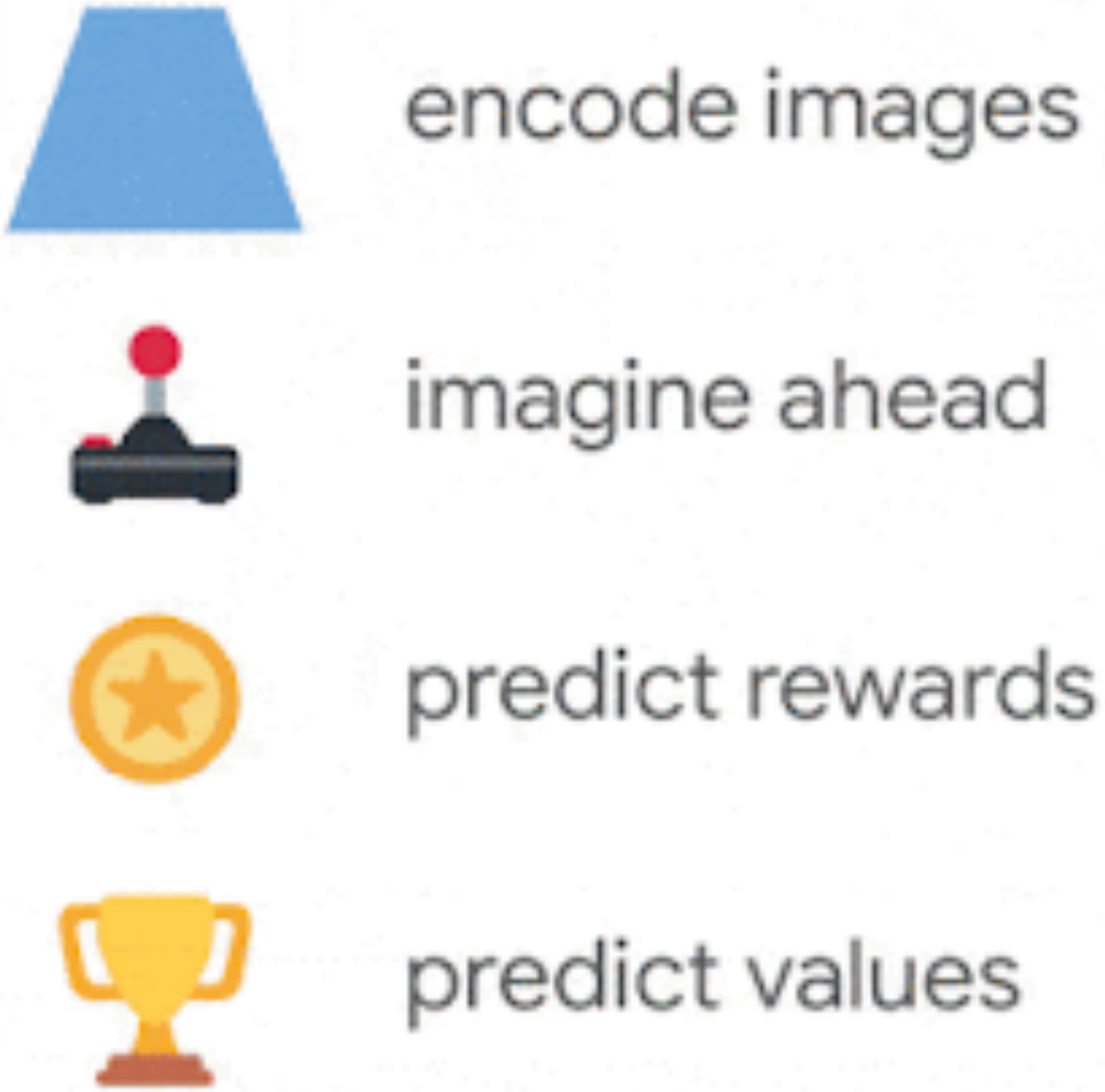
Predict rewards
(Freeze gradients)

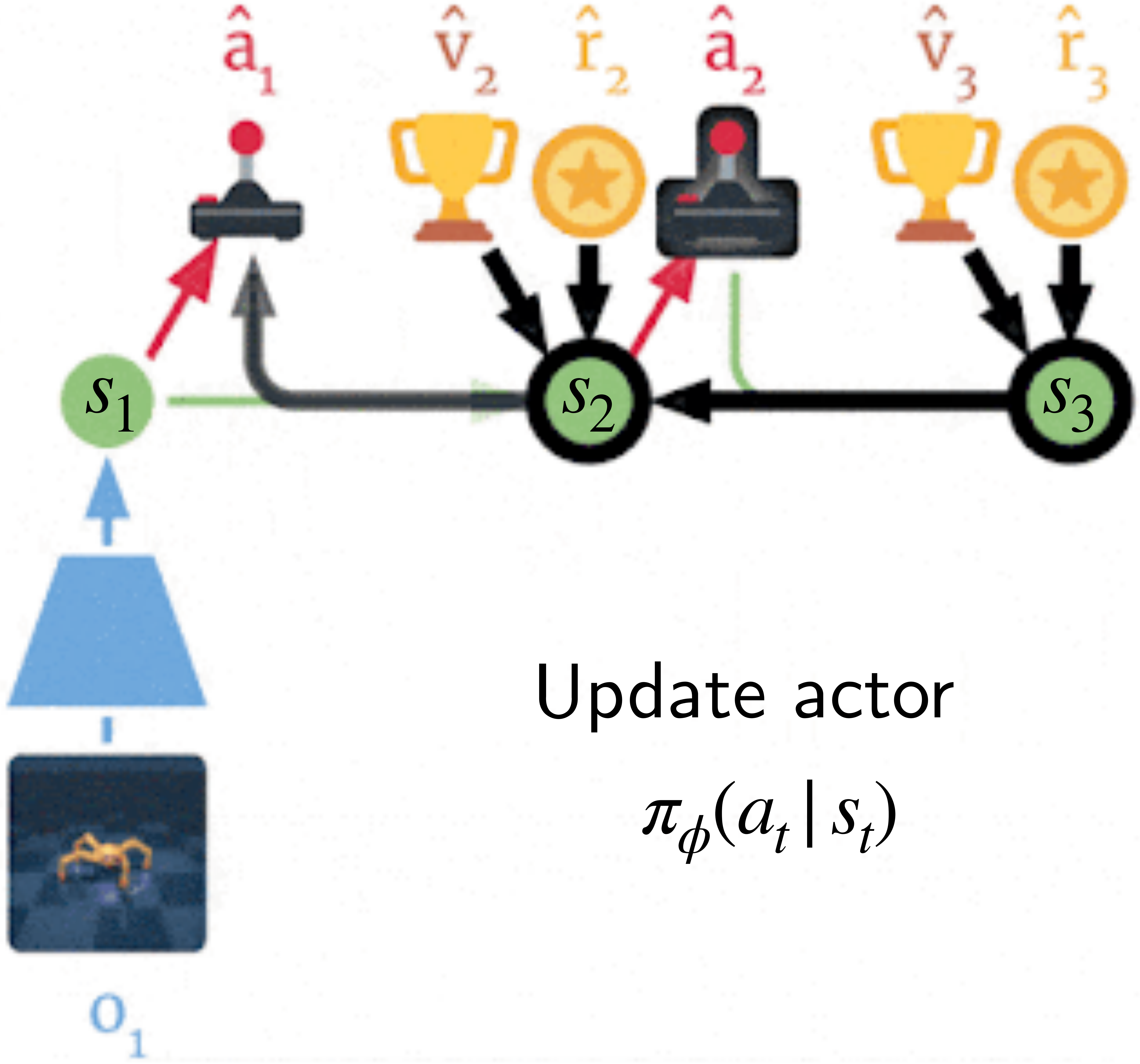
$$q_\theta(r_t \mid s_t)$$



Update critic

$$V_\psi(s_t)$$

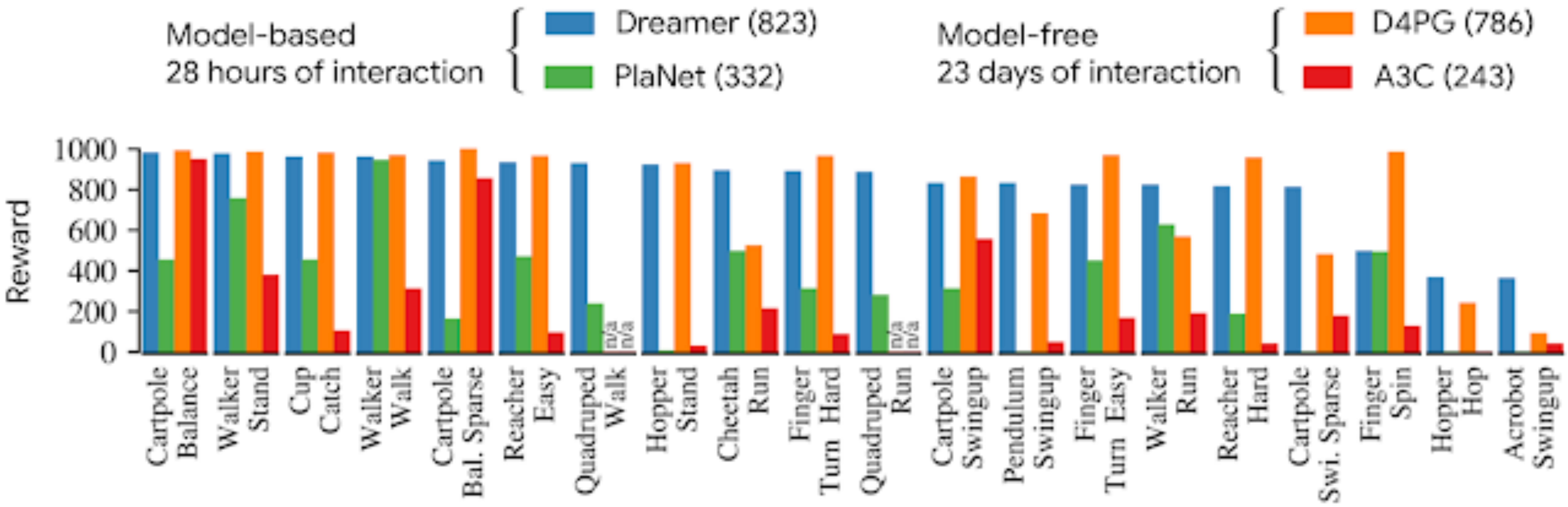




Update actor

$$\pi_\phi(a_t \mid s_t)$$

DREAMER: Results



DREAMER is a template for Model-based RL

But there are many challenges as we scale to harder real-world applications

DREAMER V2:

Tackling the world of Atari Games

Mastering Atari with Discrete World Models

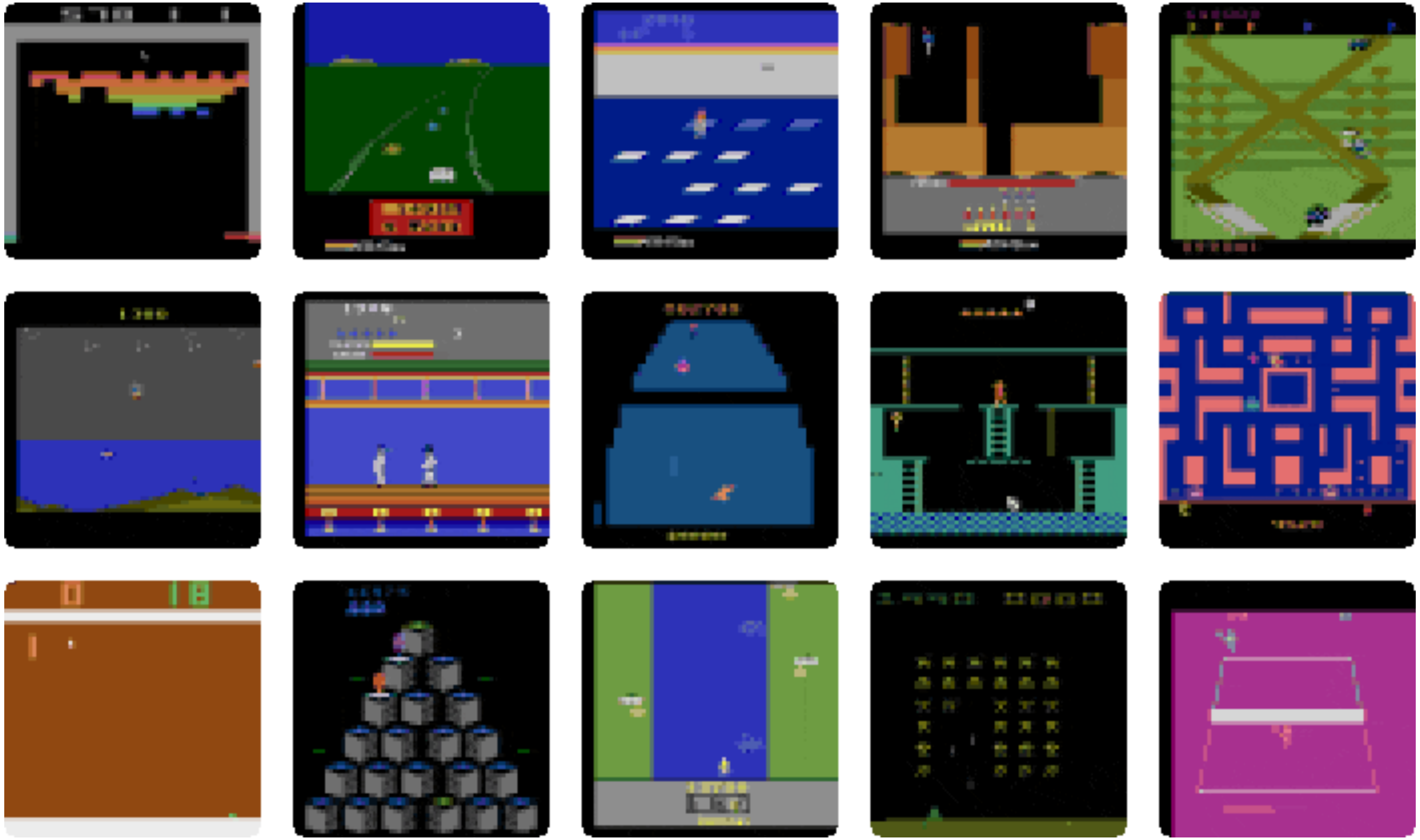
2021

Danijar Hafner *
Google Research

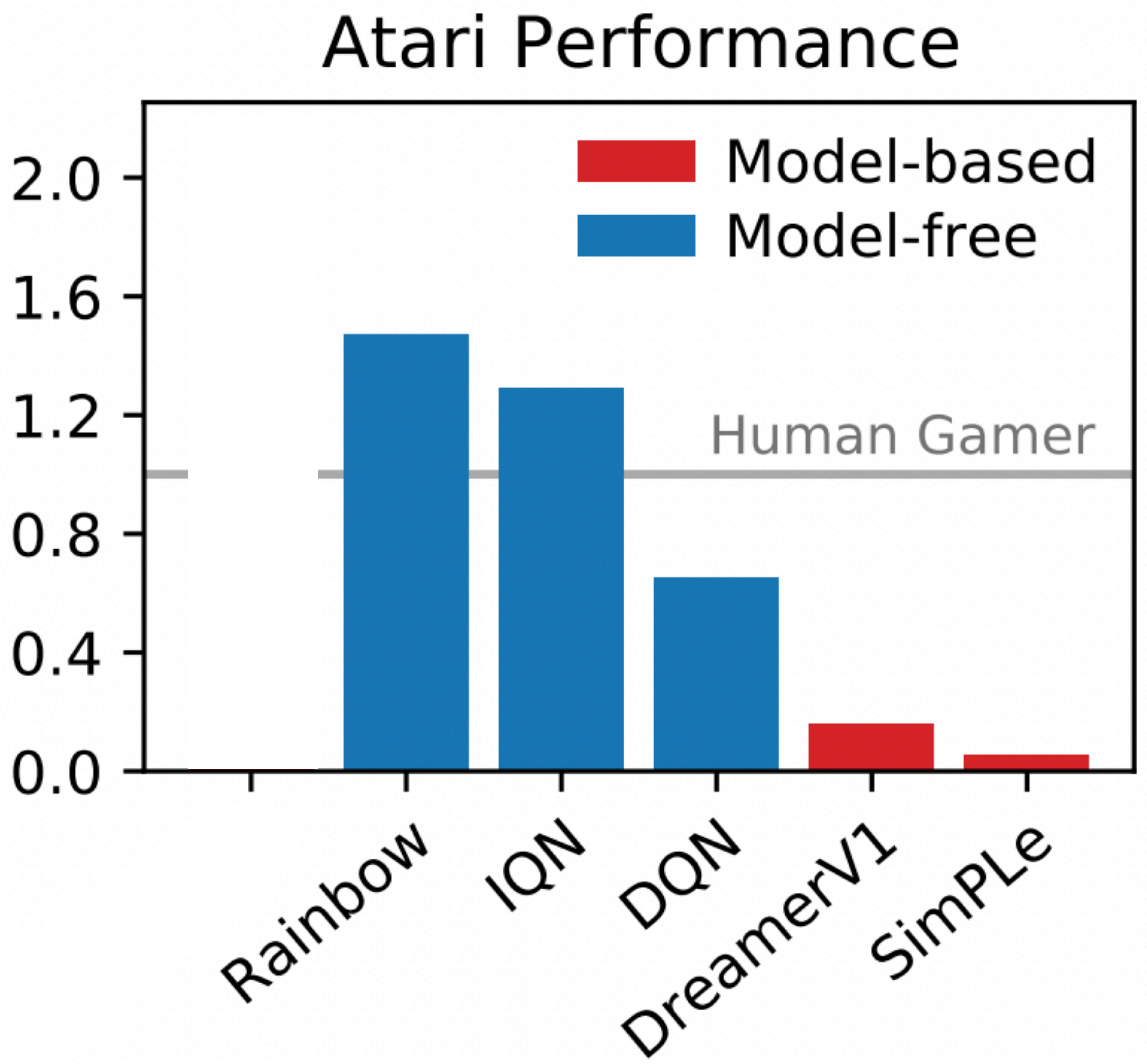
Timothy Lillicrap
DeepMind

Mohammad Norouzi
Google Research

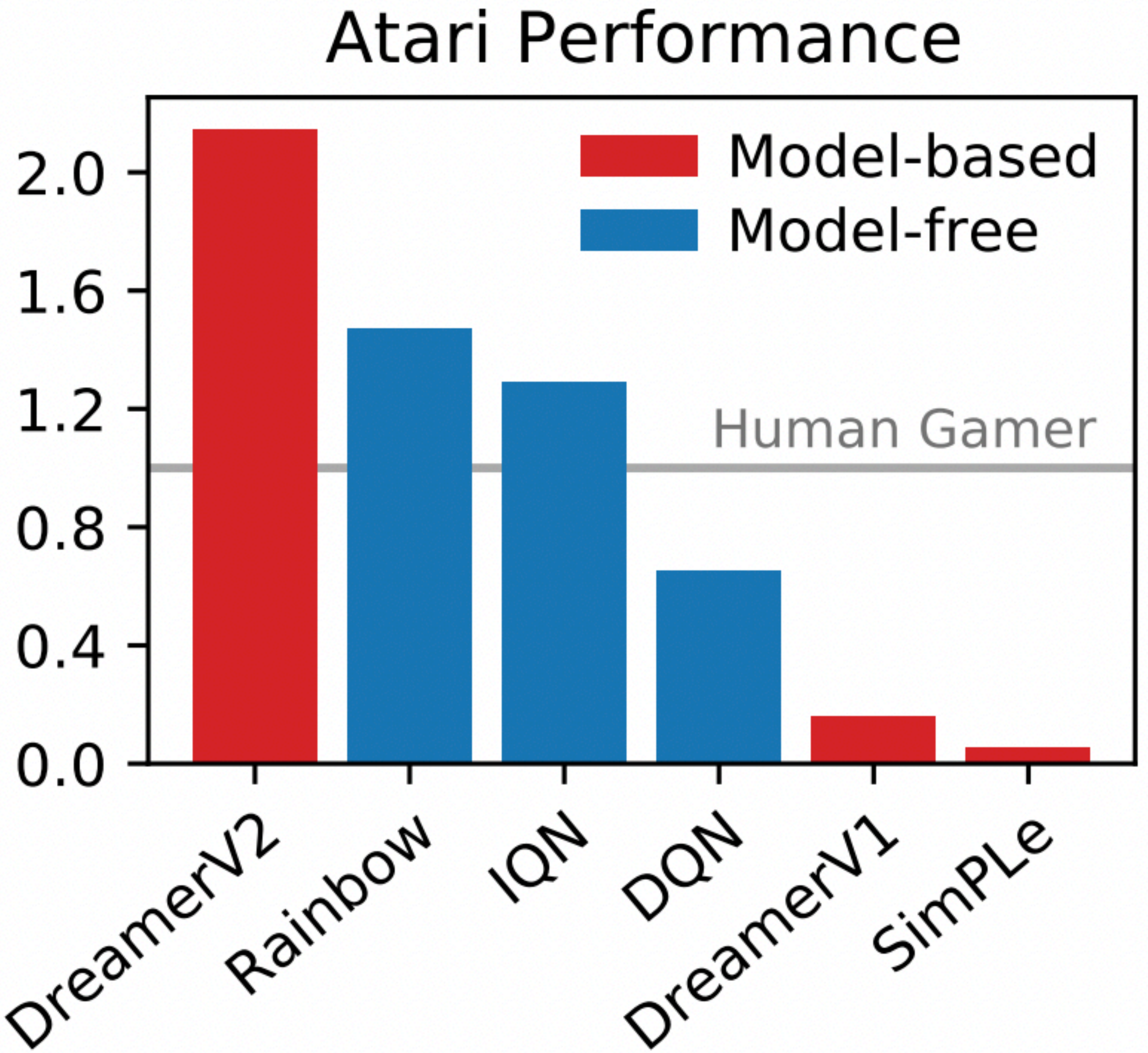
Jimmy Ba
University of Toronto

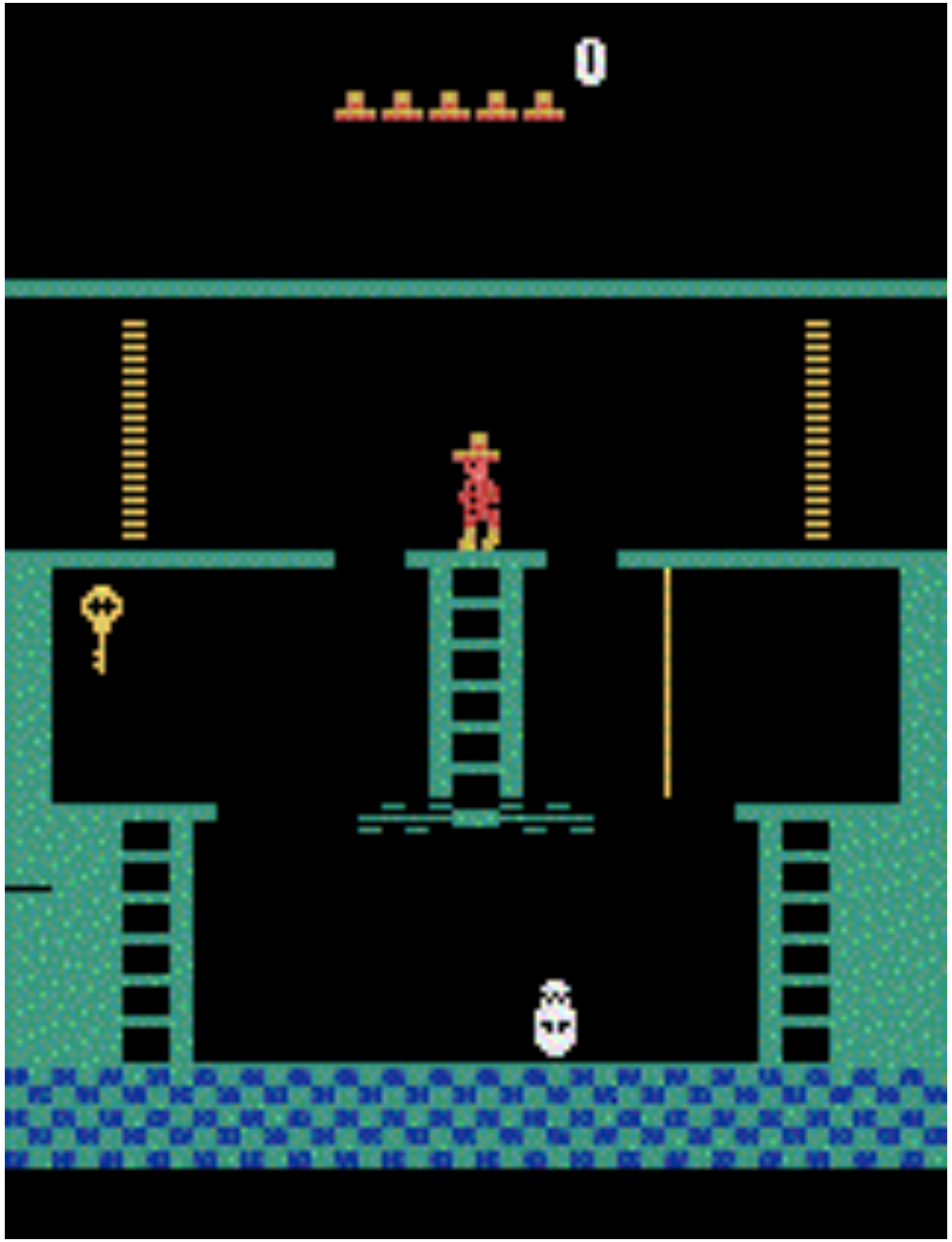


Atari was hard for Model Based RL



DreamerV2 beats all model free!



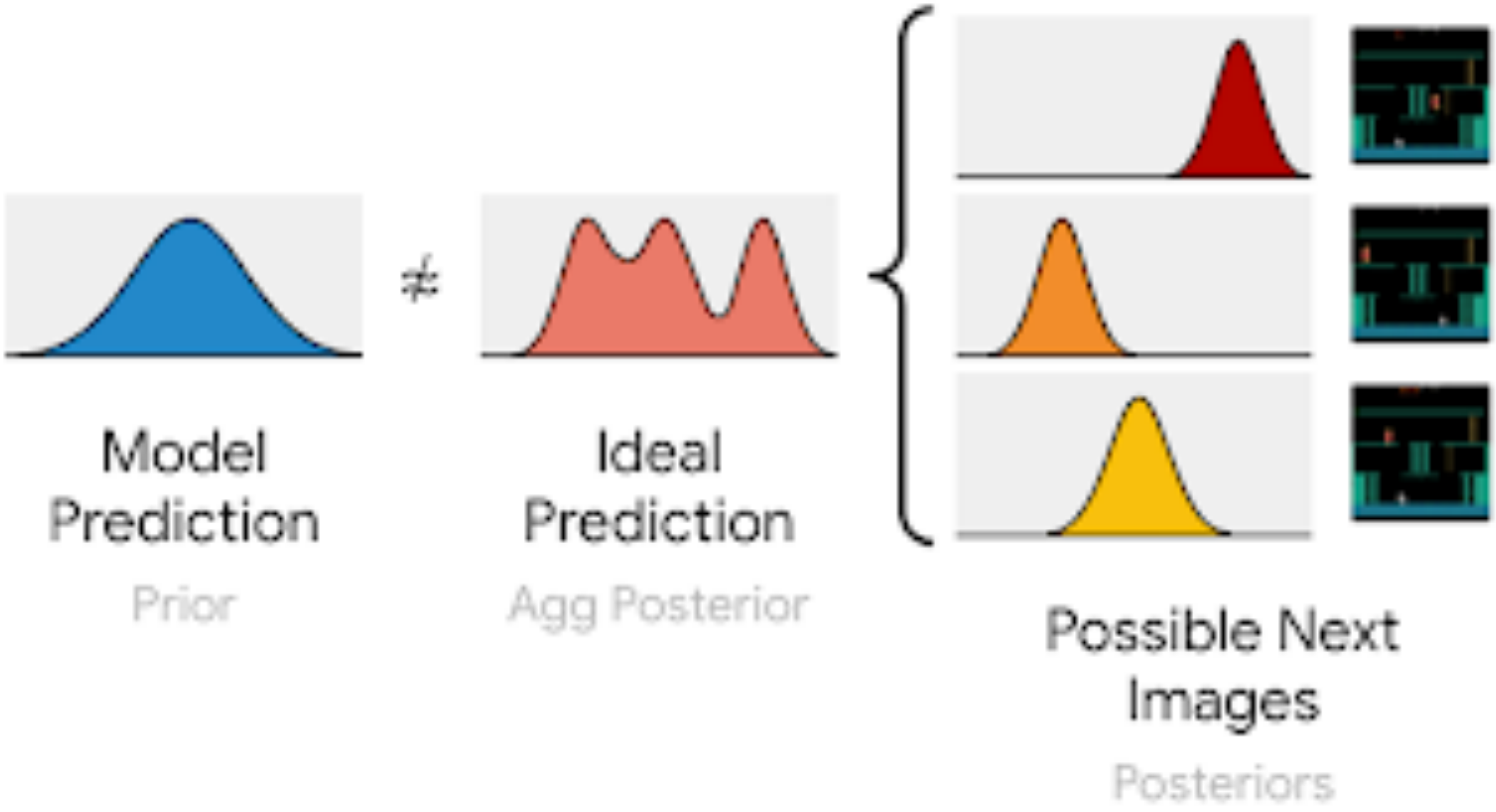


Montezuma's Revenge:

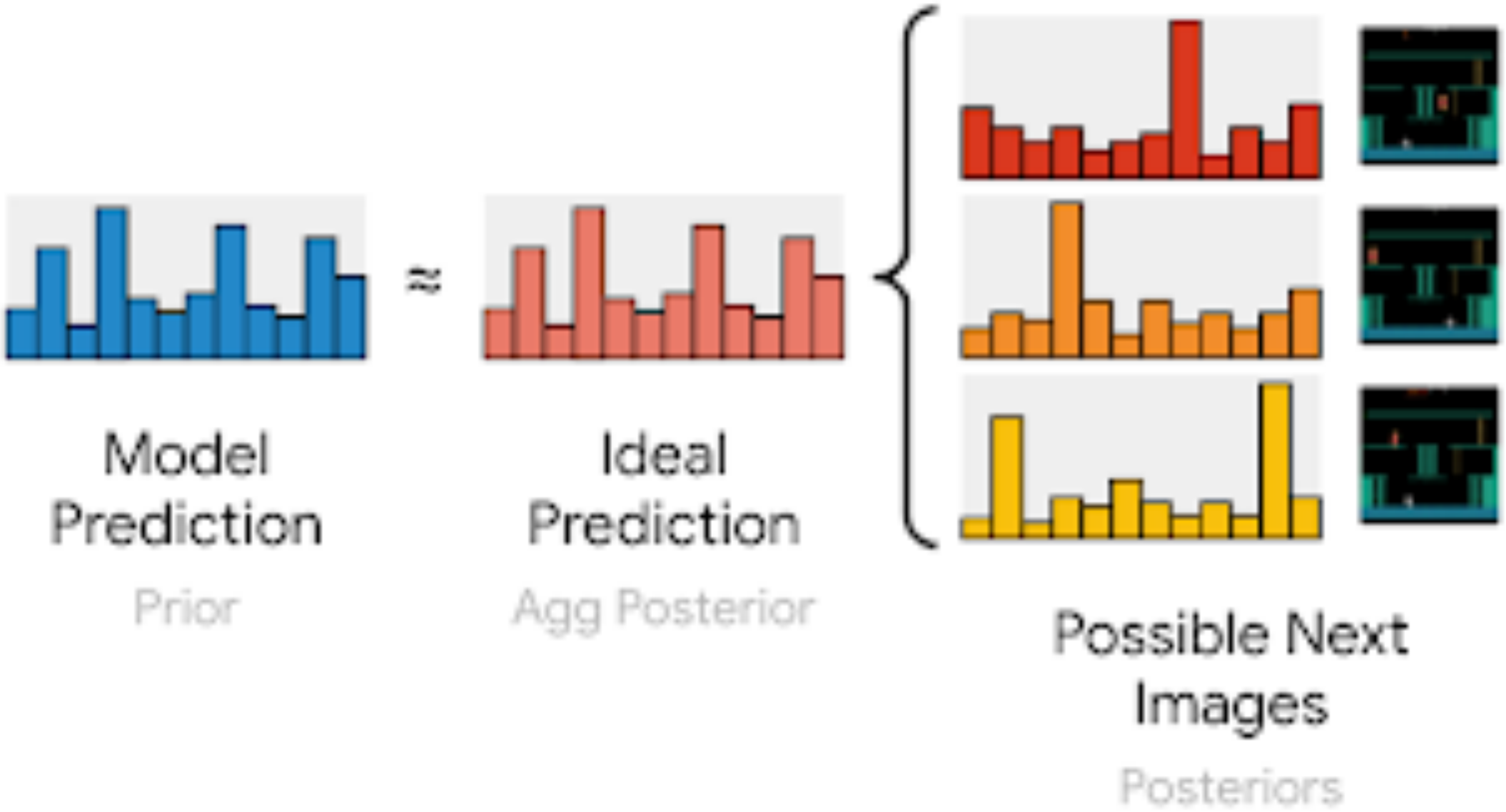
A really challenging
Atari Game!

Challenge: Dreamer V1 predicts a single mode of dynamics

Dreamer V1 predicts single mode dynamics

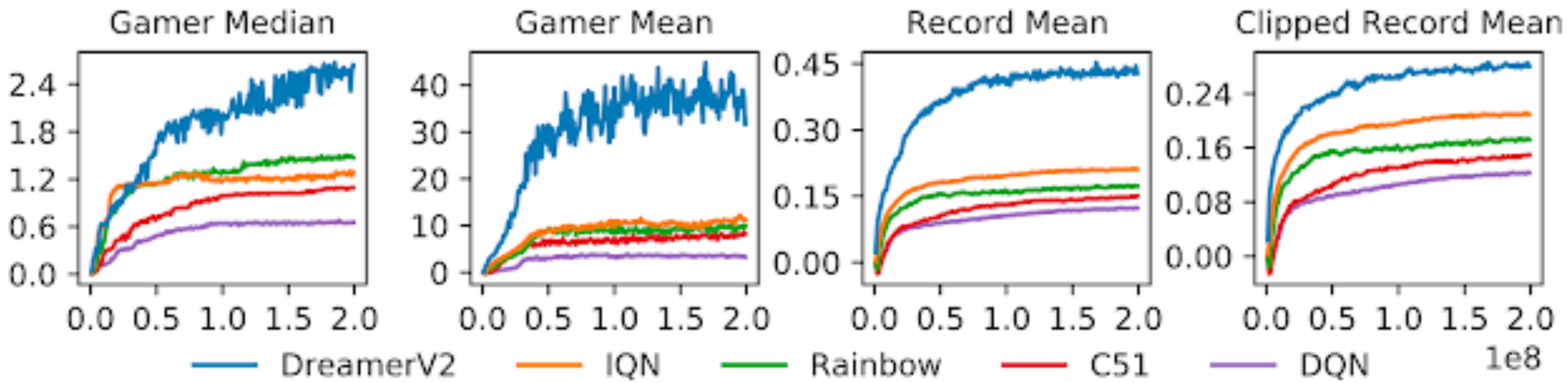


Idea: Predict multiple discrete modes!




$\hat{x}_1$
$\hat{x}_2$
$\hat{x}_3$
a_1
a_2
Sparse Representation
h_1
h_2
h_3
$\hat{z}_1$
z_1
$\hat{z}_2$
z_2
$\hat{z}_3$
z_3
min KL
min KL
min KL
32 categoricals
32 classes each
x_1
x_2
x_3


True
Model


Gamer Median
Gamer Mean
Record Mean
Clipped Record Mean
2.4
1.8
1.2
0.6
0.0
40
30
20
10
0
0.45
0.30
0.15
0.00
0.24
0.16
0.08
0.00
0.0 0.5 1.0 1.5 2.0
1e8
DreamerV2
IQN
Rainbow
C51
DQN