Dream to Control
Learning Behaviors by Latent Imagination

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, Mohammad Norouzi

UNIVERSITY OF TORONTO   Google Brain   DeepMind

@danijarh

danijar.com/dreamer

We introduce Dreamer

1. Scalable reinforcement learning from pixels using a world model
2. Learn actor and value in imagination for long-sighted behaviors
3. Efficiently update actor by backprop through imagined sequences

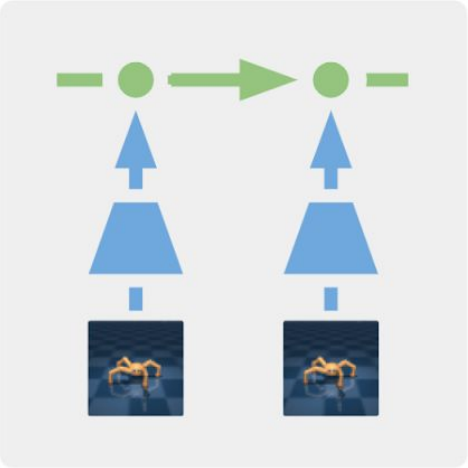
We introduce Dreamer

1. Scalable reinforcement learning from pixels using a world model
2. Learn actor and value in imagination for long-sighted behaviors
3. Efficiently update actor by backprop through imagined sequences

We introduce Dreamer

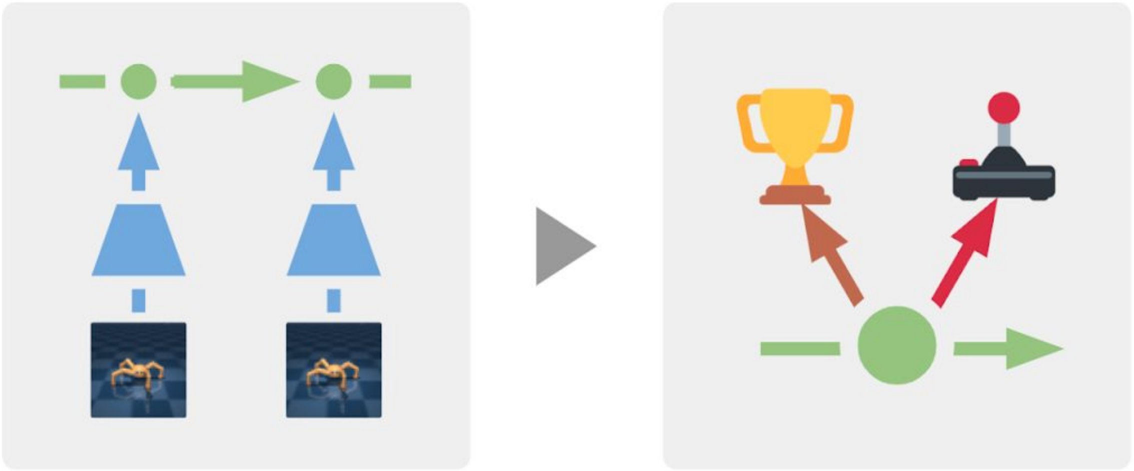
1. Scalable reinforcement learning from pixels using a world model
2. Learn actor and value in imagination for long-sighted behaviors
3. Efficiently update actor by backprop through imagined sequences

Dreamer Agent Overview



World Model
Learning

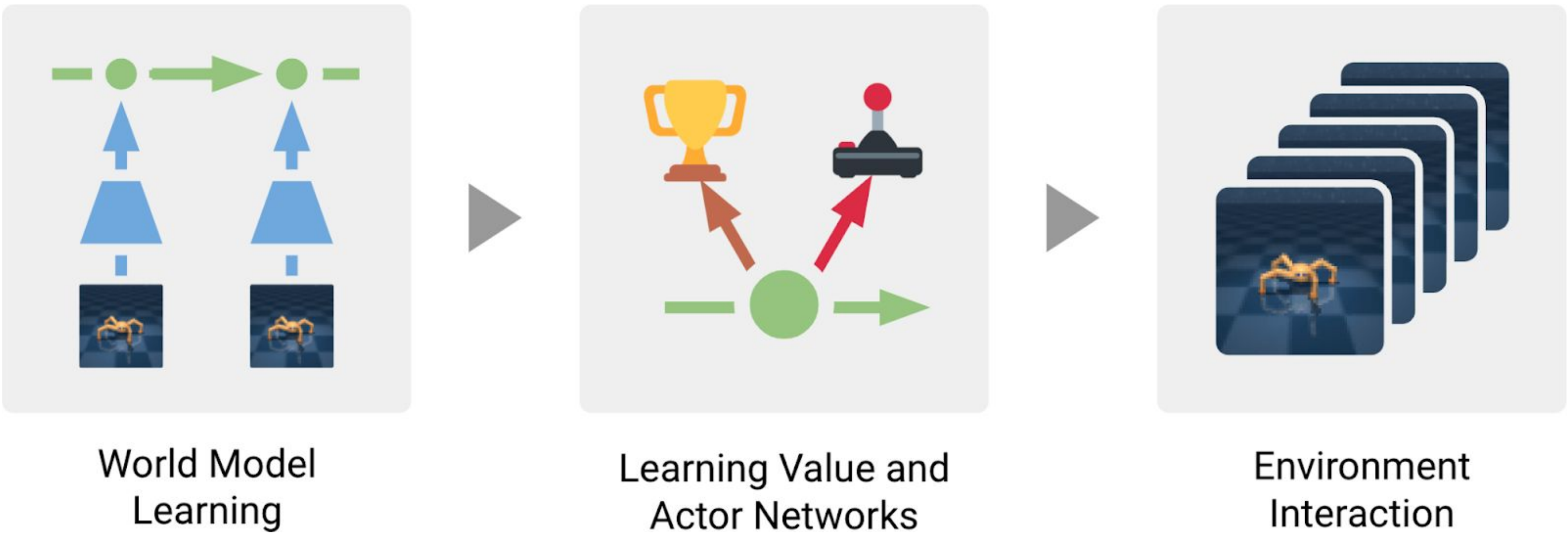
Dreamer Agent Overview



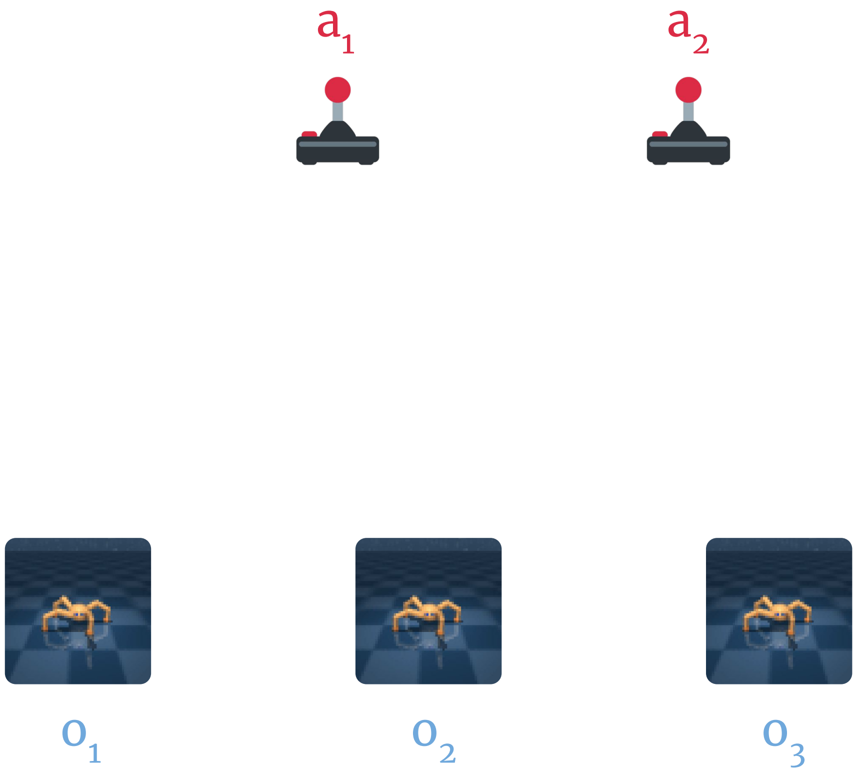
World Model Learning

Learning Value and Actor Networks

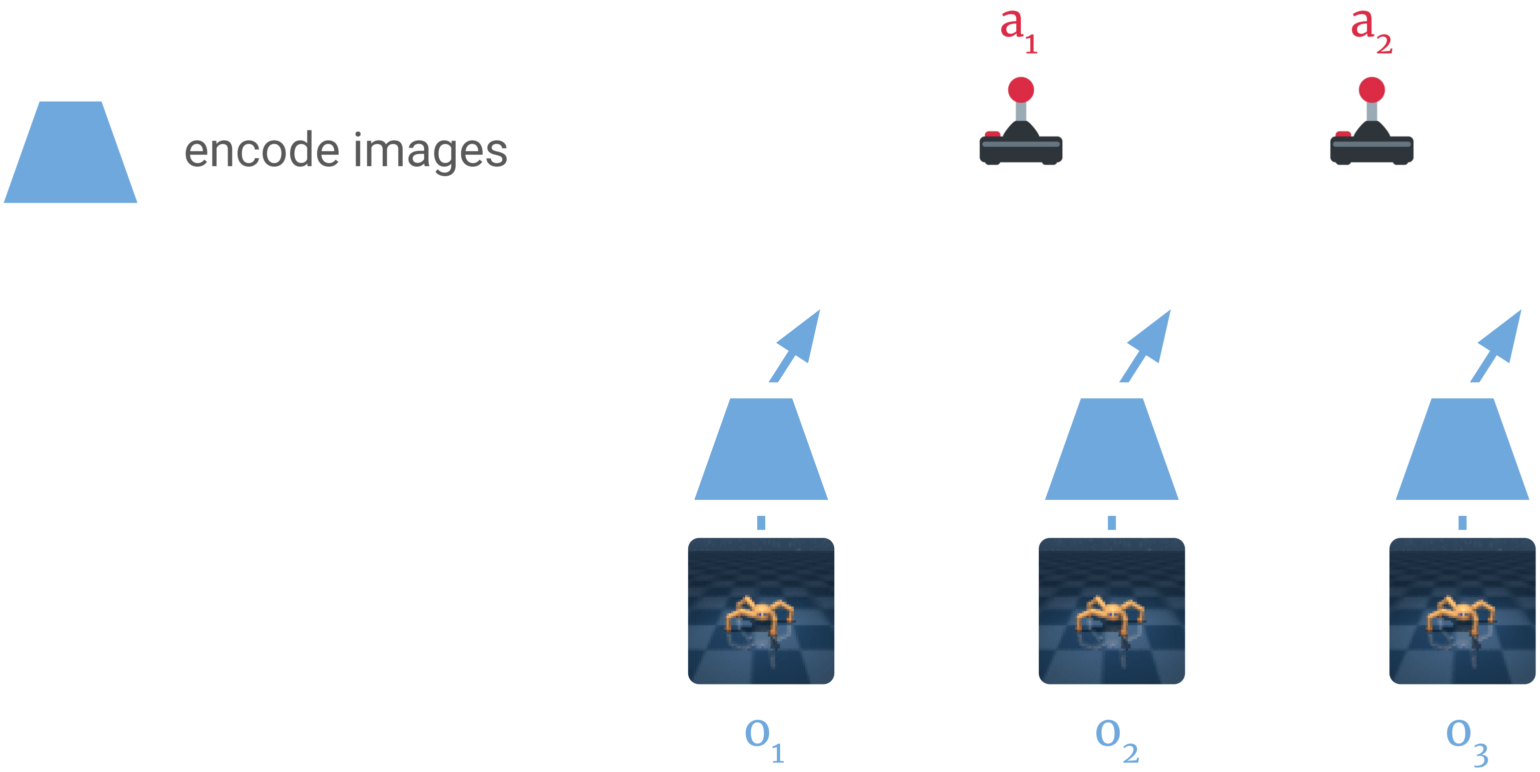
Dreamer Agent Overview



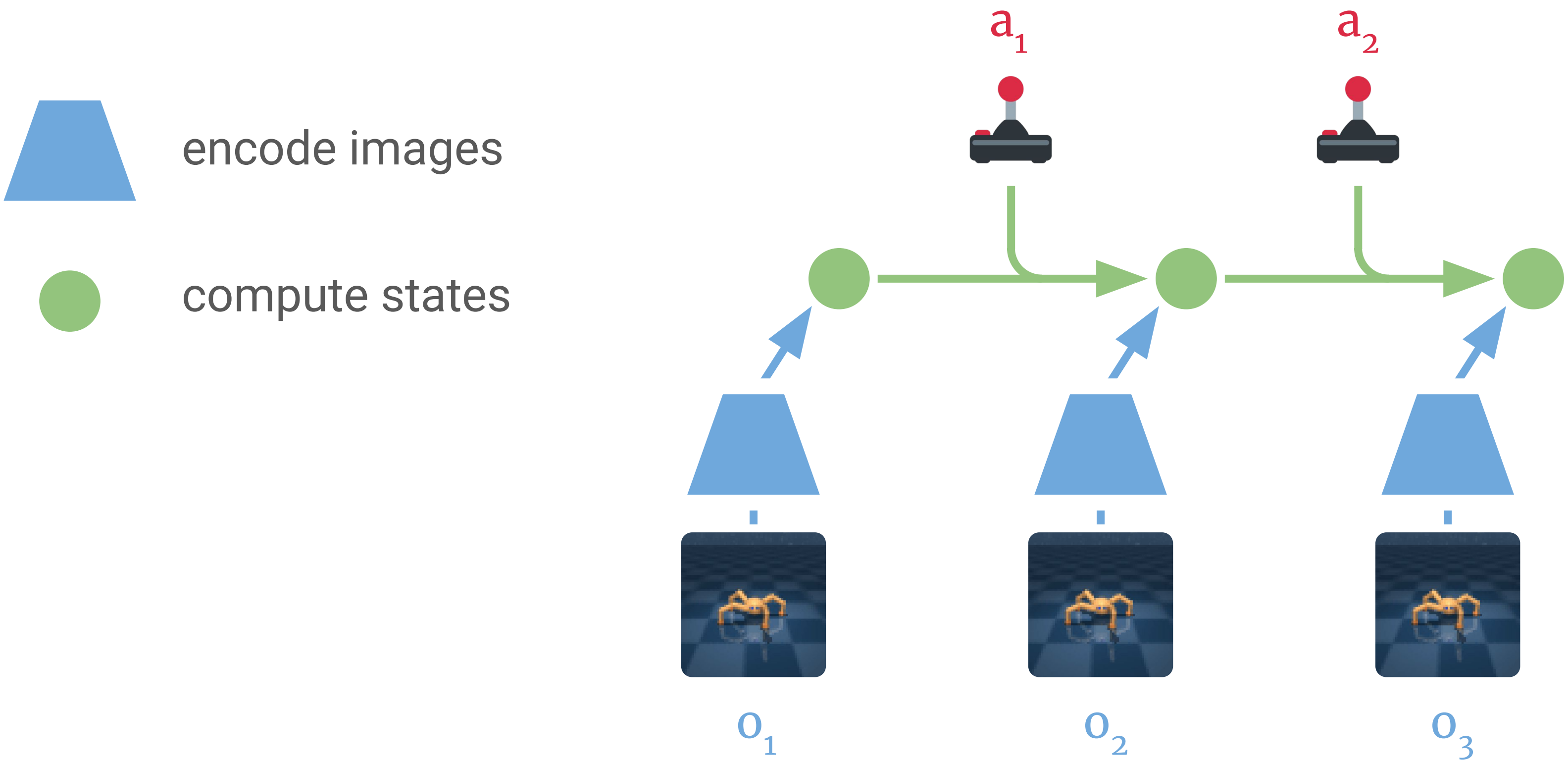
World Model with Latent States



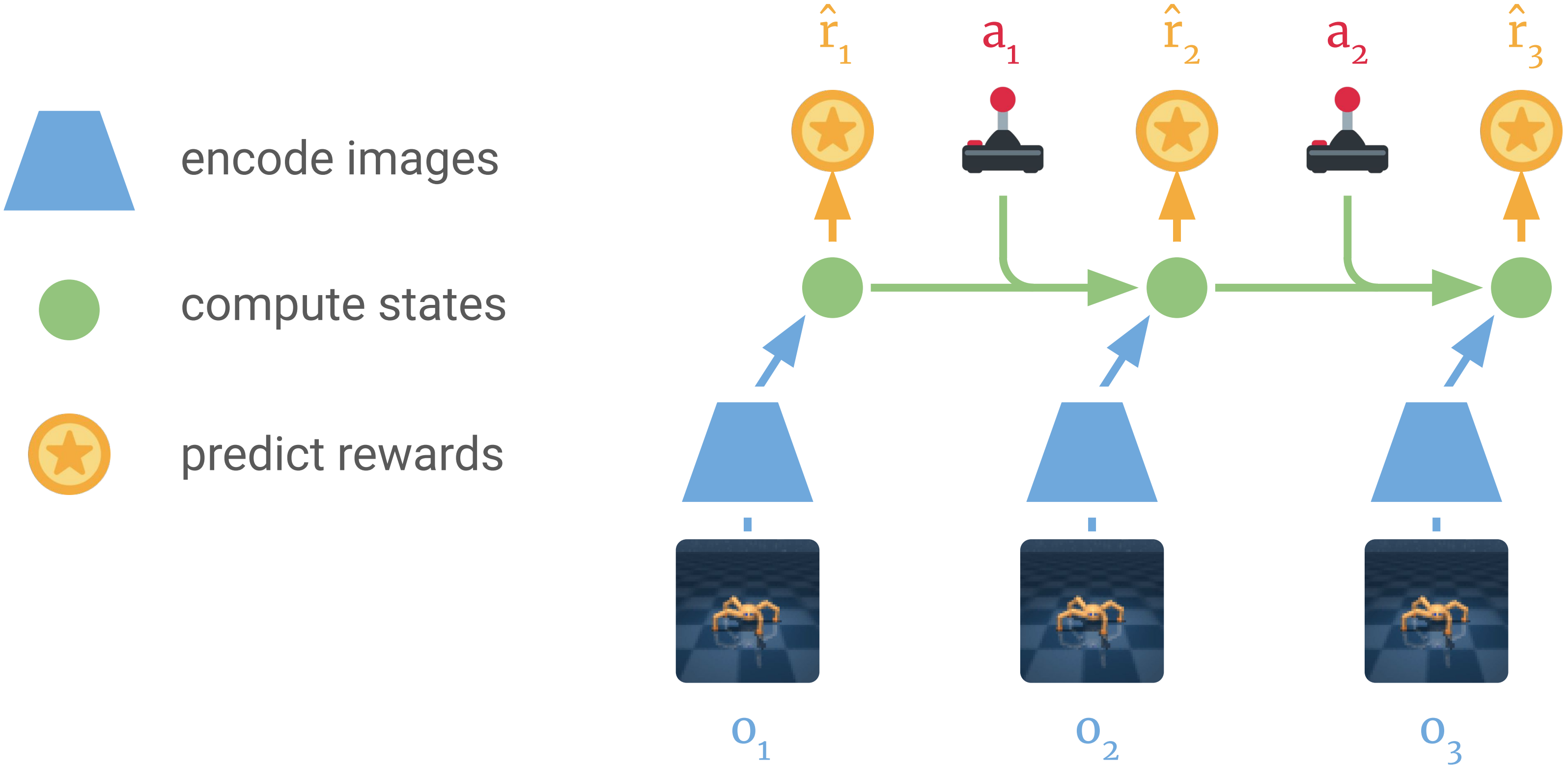
World Model with Latent States



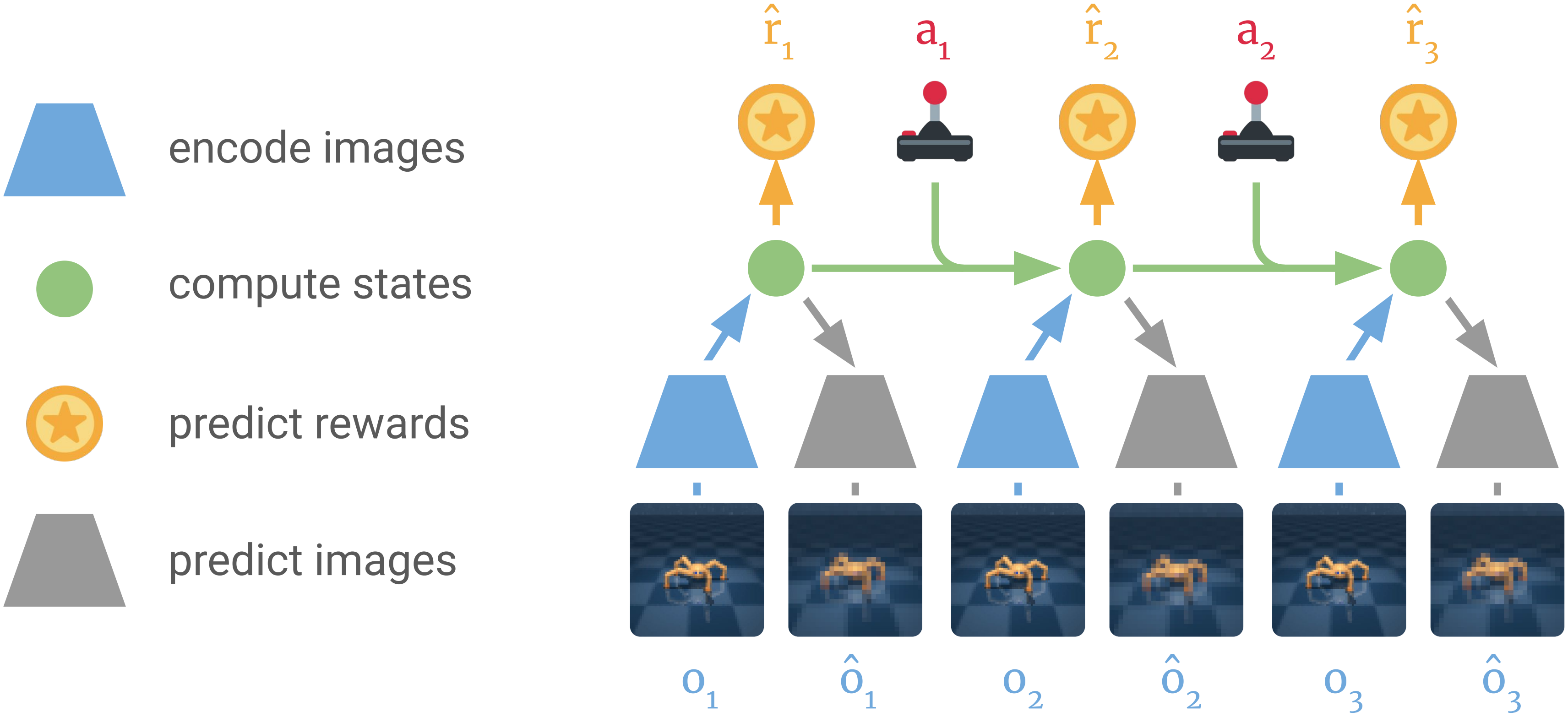
World Model with Latent States



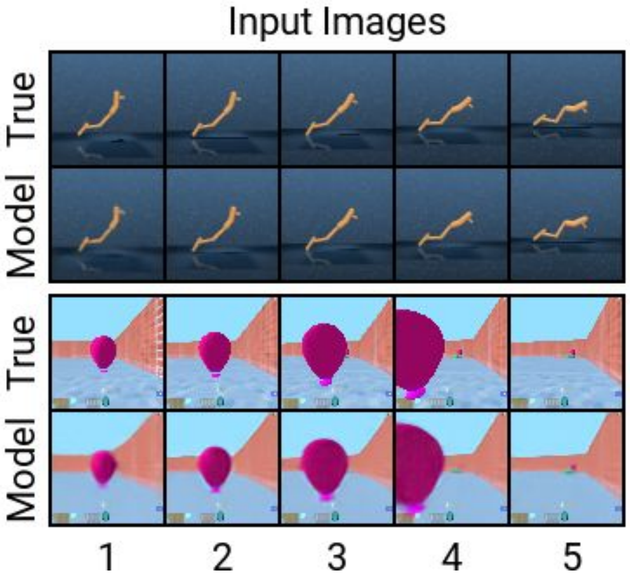
World Model with Latent States



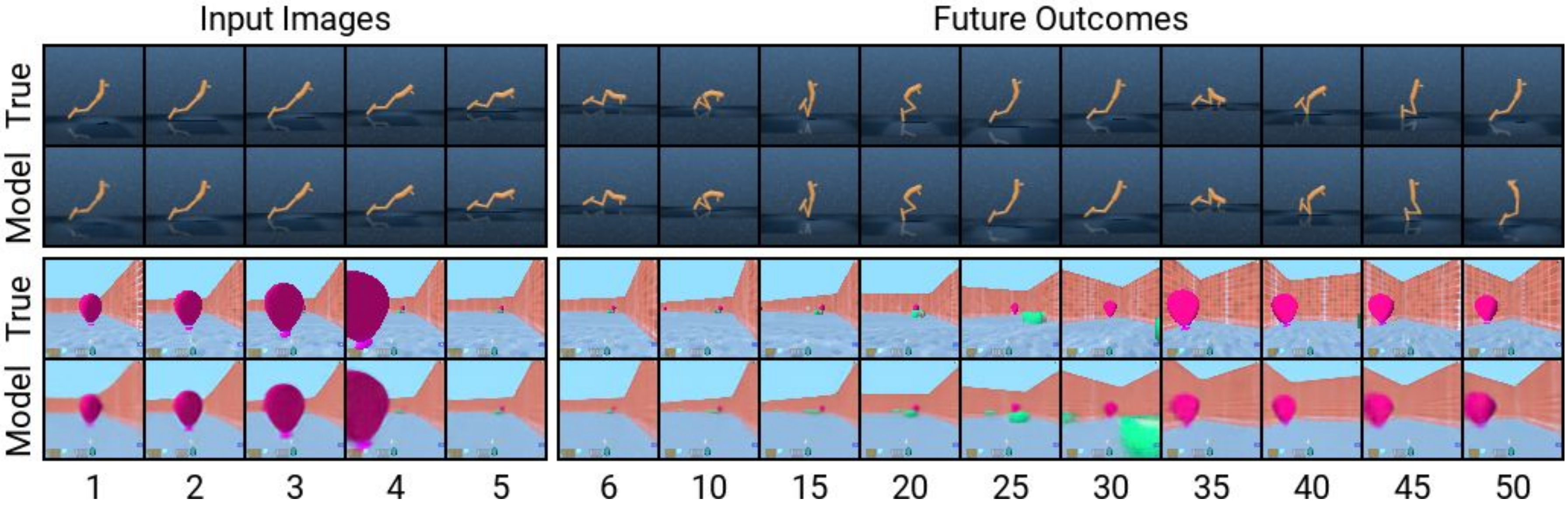
World Model with Latent States



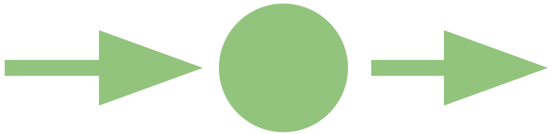
Long-Term Video Prediction



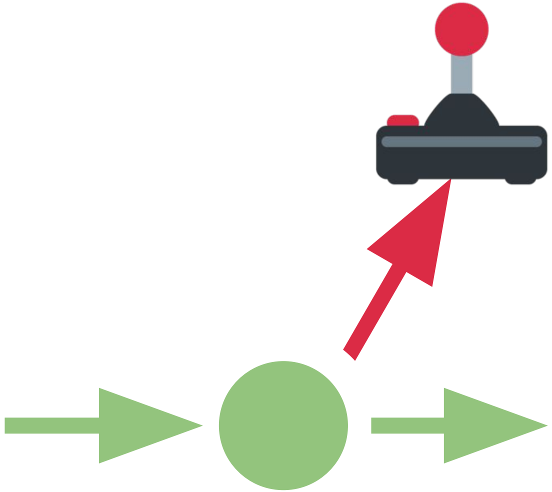
Long-Term Video Prediction



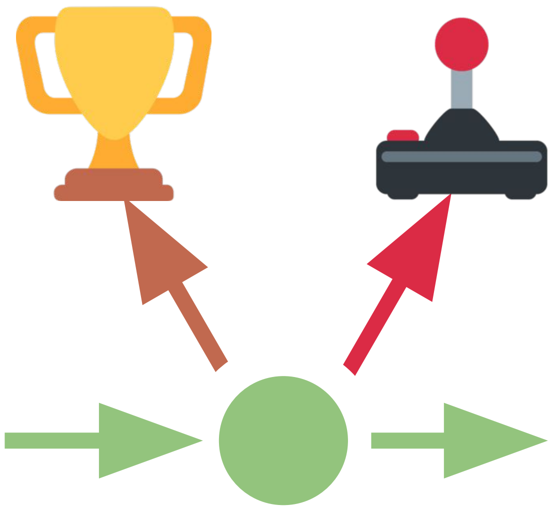
Learning Behaviors by Latent Imagination



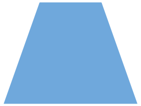
Learning Behaviors by Latent Imagination



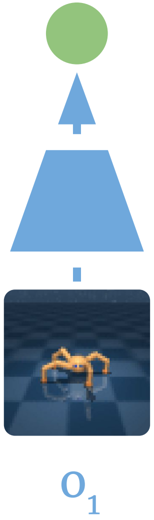
Learning Behaviors by Latent Imagination



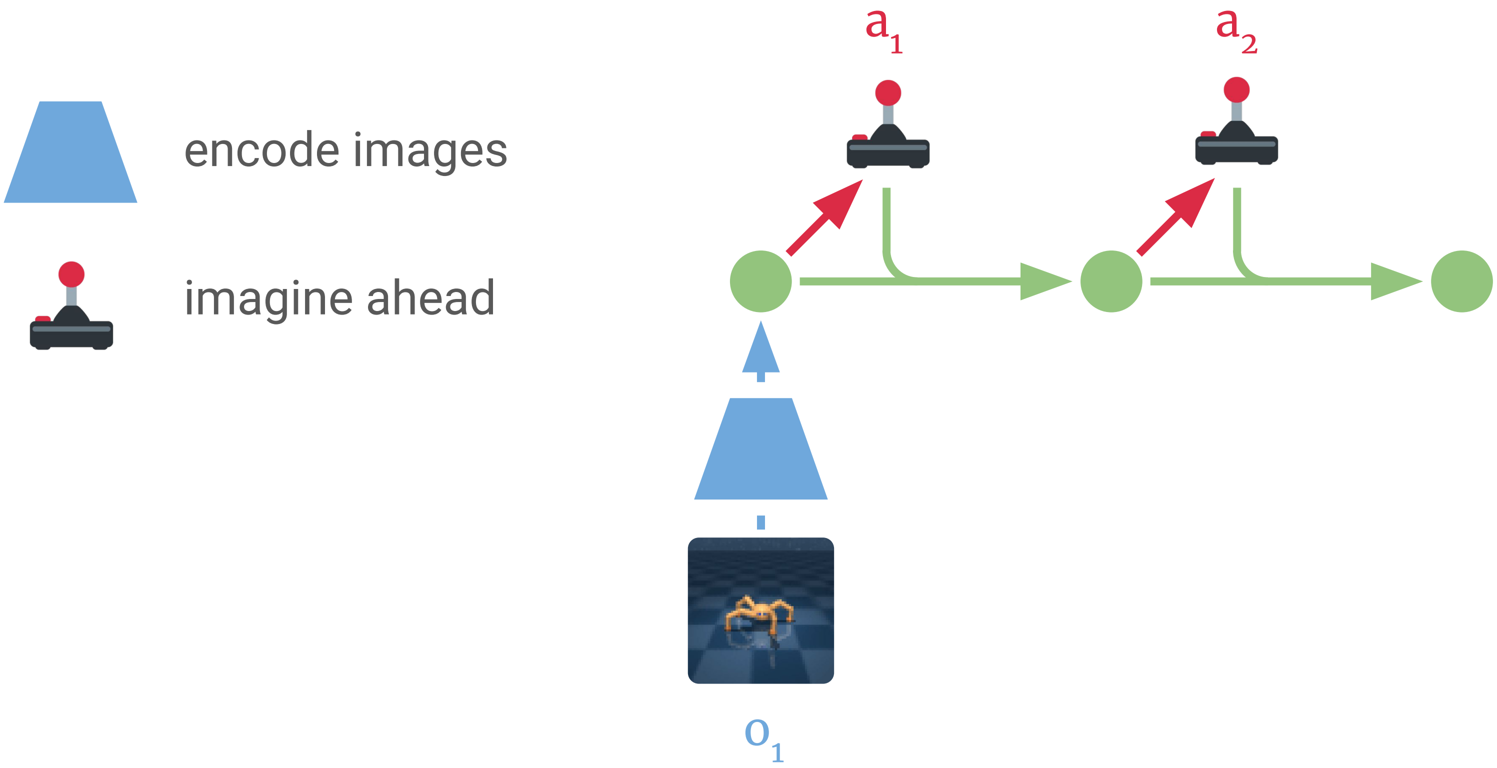
Learning Behaviors by Latent Imagination



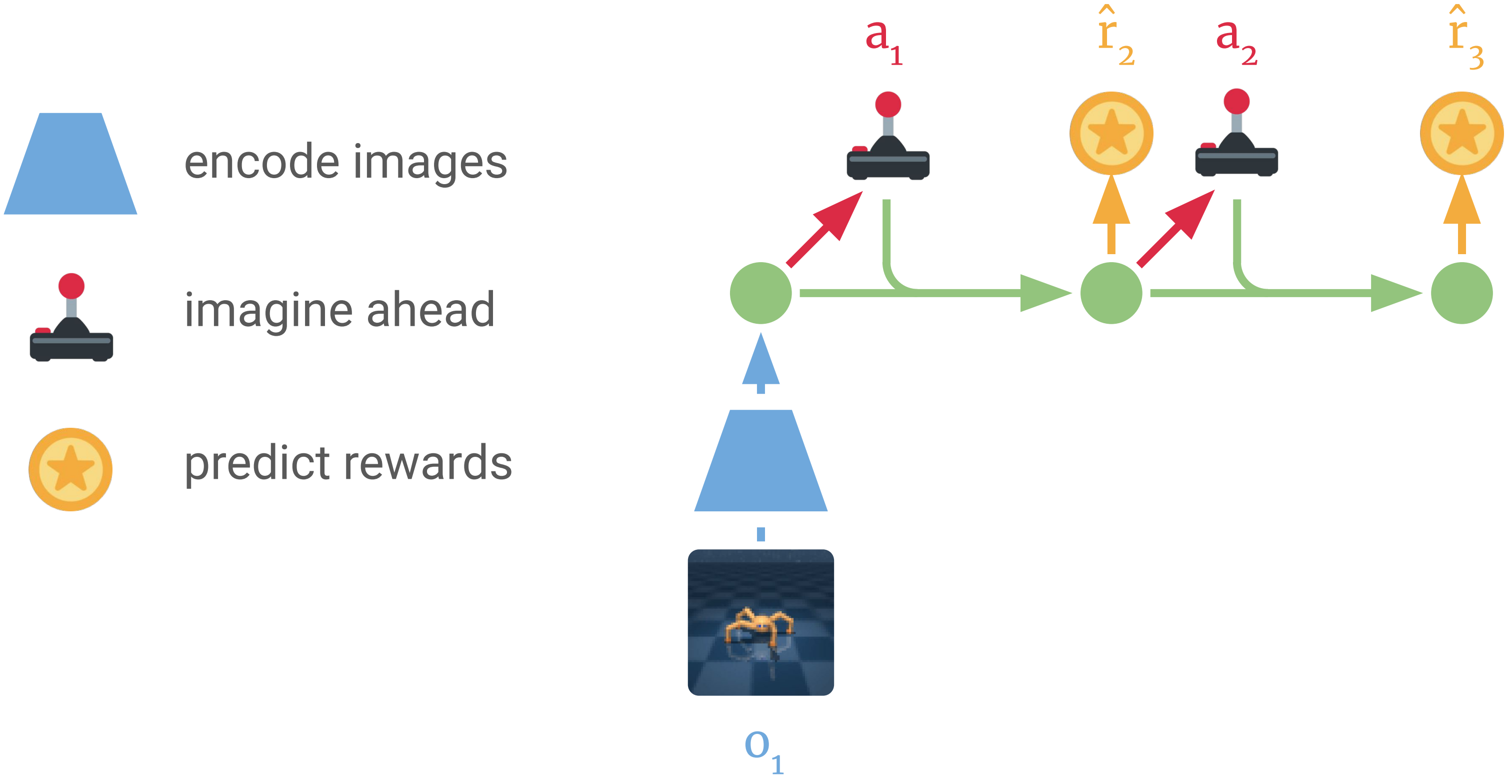
encode images



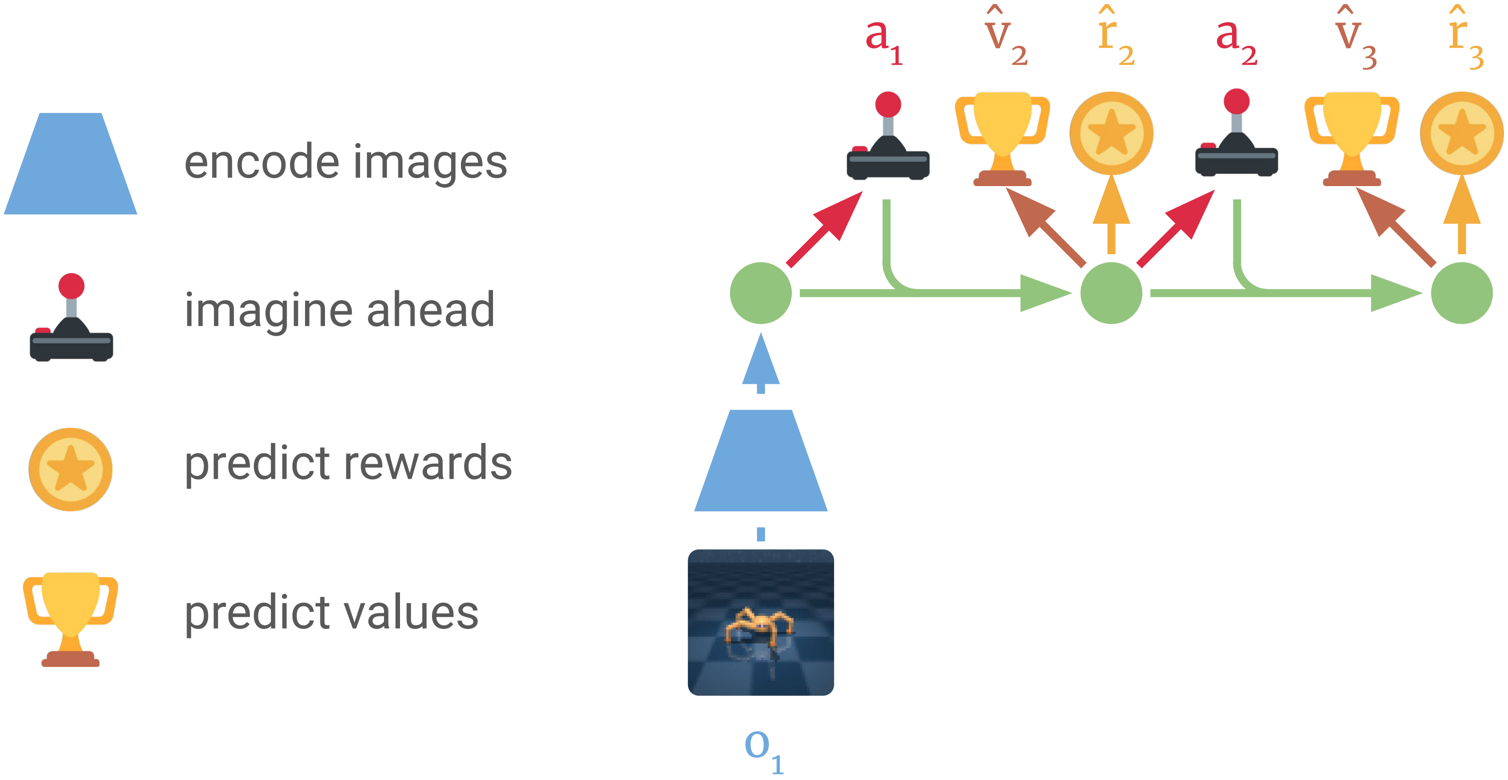
Learning Behaviors by Latent Imagination



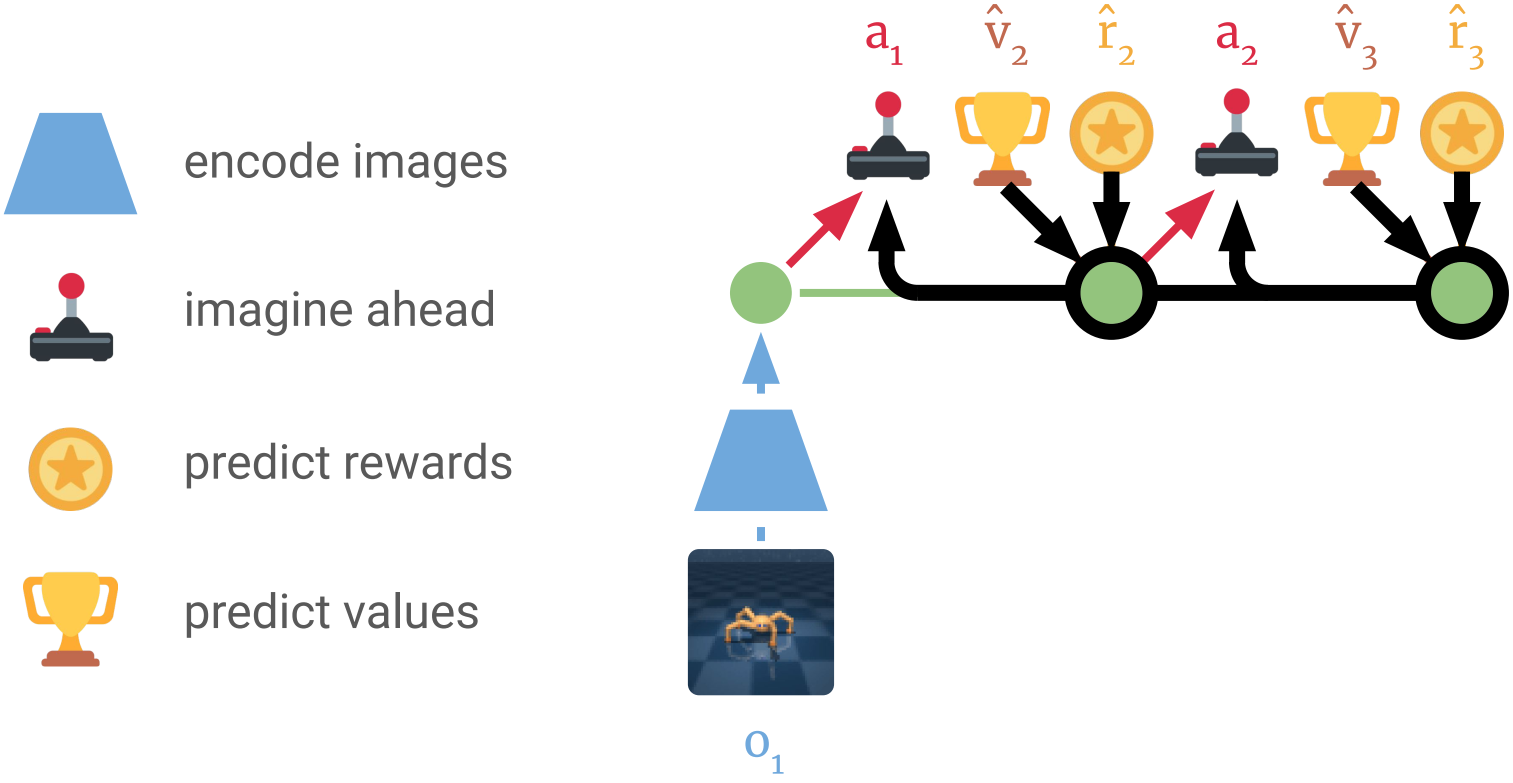
Learning Behaviors by Latent Imagination



Learning Behaviors by Latent Imagination



Learning Behaviors by Latent Imagination



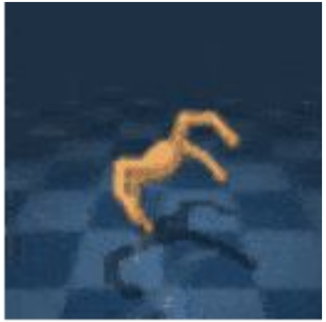
Behaviors Learned by Dreamer

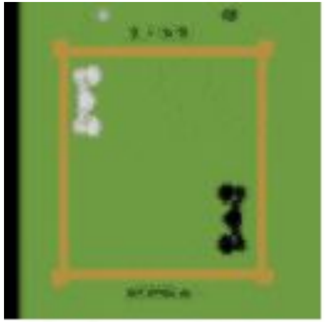

Sparse Cartpole


Acrobot Swingup

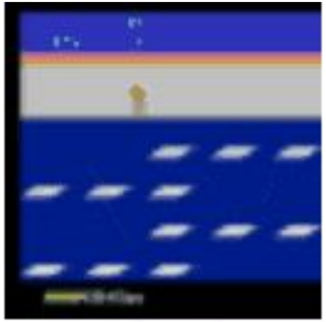

Hopper Hop

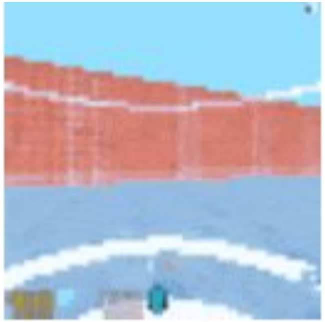

Walker Run


Quadruped Run


Boxing


Freeway


Frostbite

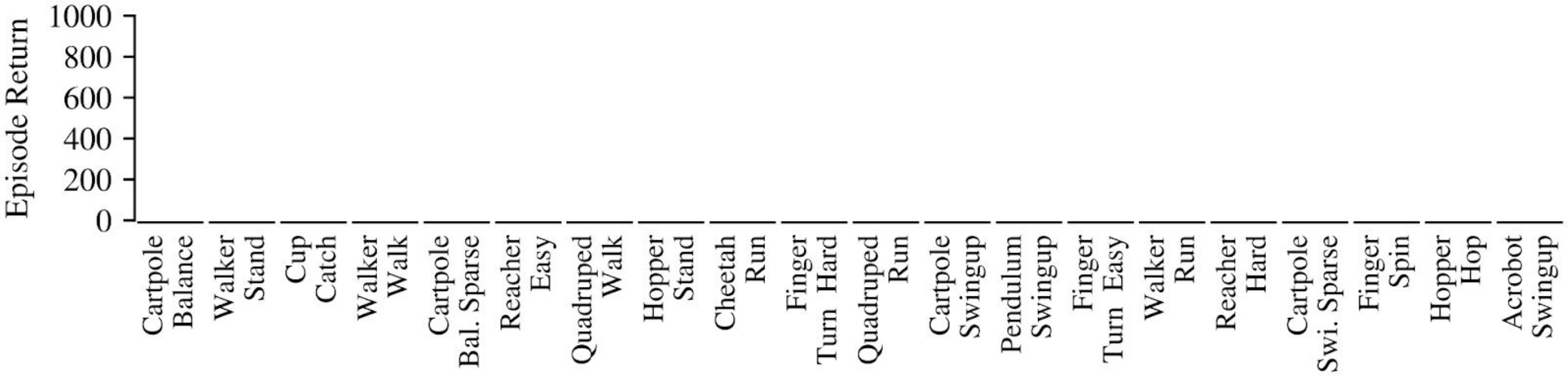

Collect Objects


Watermaze

Large-Scale Evaluation for Control from Pixels

Model-based:
28 hours of interaction

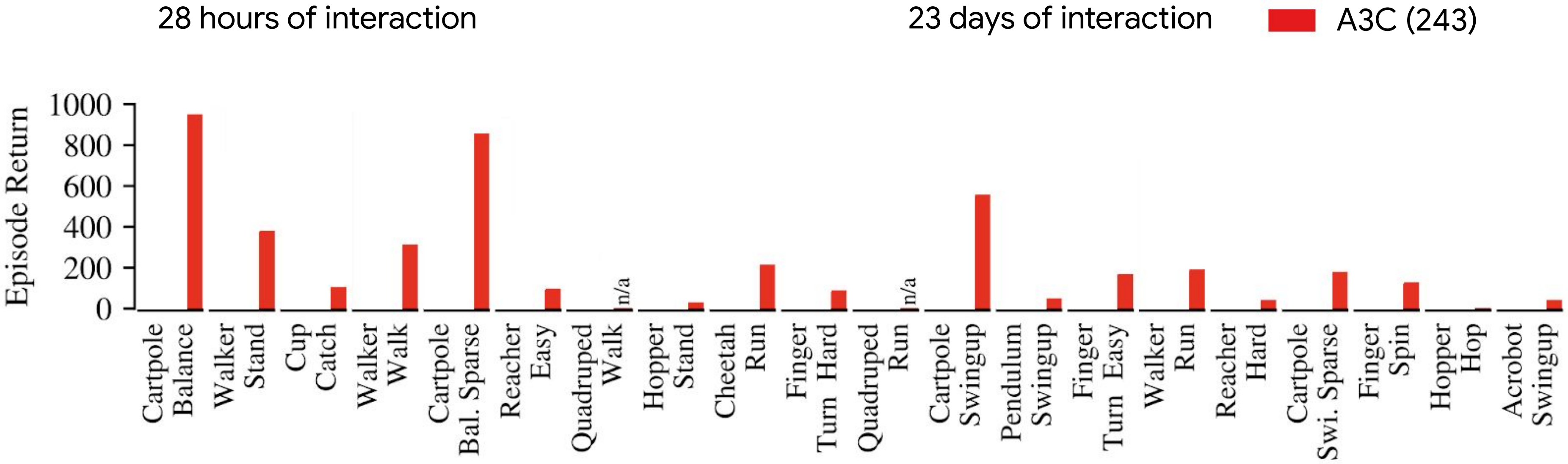
Model-free:
23 days of interaction



Large-Scale Evaluation for Control from Pixels

Model-based:
28 hours of interaction

Model-free:
23 days of interaction



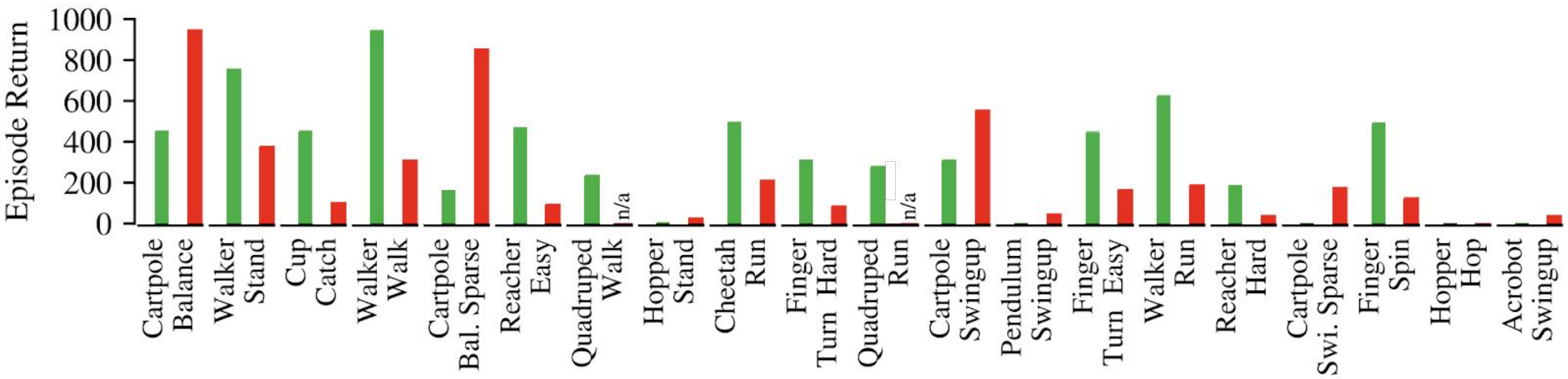
Large-Scale Evaluation for Control from Pixels

Model-based:
28 hours of interaction

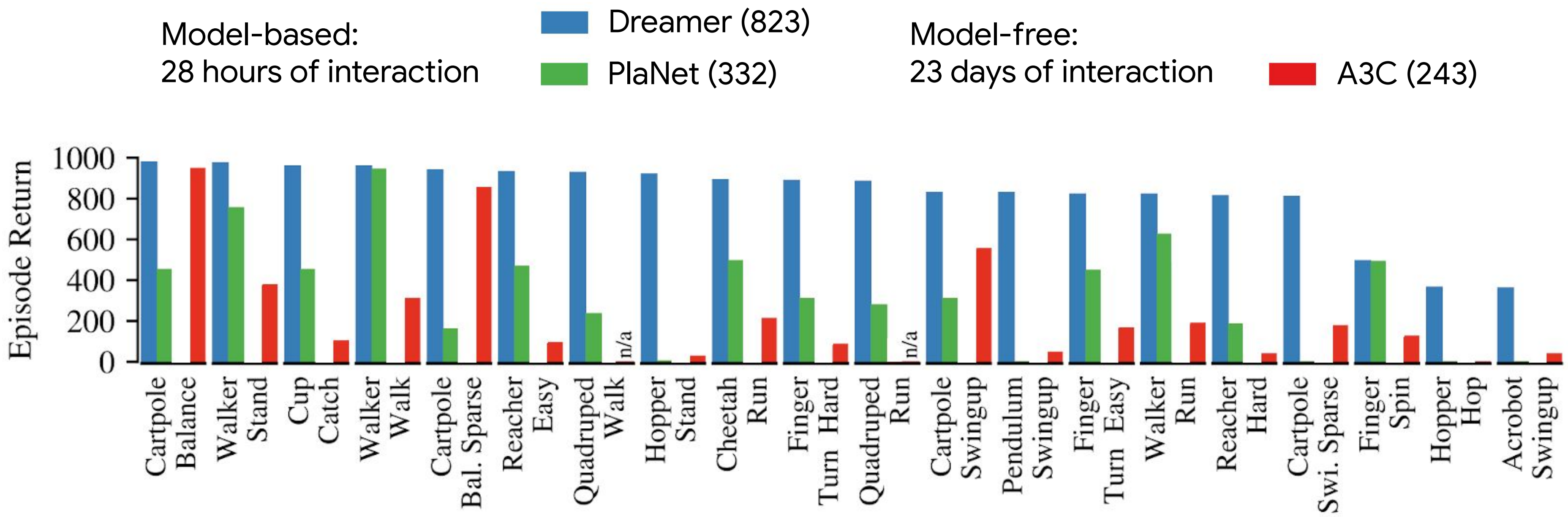
PlaNet (332)

Model-free:
23 days of interaction

A3C (243)



Large-Scale Evaluation for Control from Pixels



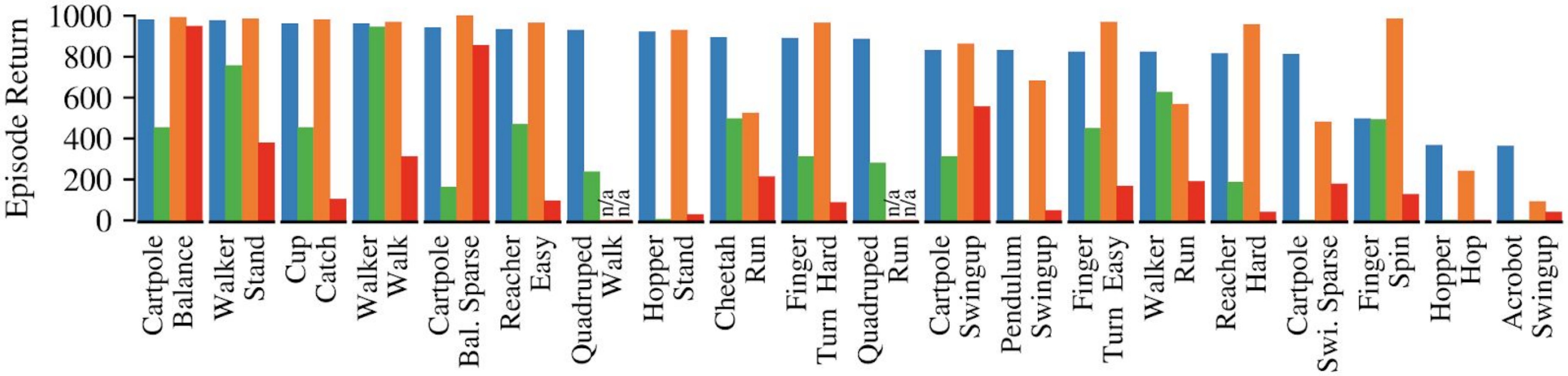
Large-Scale Evaluation for Control from Pixels

Model-based:
28 hours of interaction

Dreamer (823)
PlaNet (332)

Model-free:
23 days of interaction

D4PG (786)
A3C (243)



Google AI Blog

Introducing Dreamer: Scalable Reinforcement Learning Using World Models

Research into how artificial agents can choose actions to achieve goals is making rapid progress in large part due to the use of reinforcement learning (RL). *Model-free* approaches to RL, which learn to predict successful actions through trial and error, have enabled DeepMind's DQN to play Atari games and AlphaStar to beat world champions at Starcraft II, but require large amounts of environment interaction, limiting their usefulness for real-world scenarios.

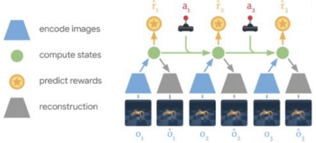
In contrast, *model-based* RL approaches additionally learn a simplified model of the environment. This *world model* lets the agent predict the outcomes of potential action sequences, allowing it to play through hypothetical scenarios to make informed decisions in new situations, thus reducing the trial and error necessary to achieve goals. In the past, it has been challenging to learn accurate world models and leverage them to learn successful behaviors. While recent research, such as our Deep Planning Network (PlaNet), has pushed these boundaries by learning accurate world models from images, model-based approaches have still been held back by ineffective or computationally expensive planning mechanisms, limiting their ability to solve difficult tasks.

Today, in collaboration with DeepMind, we present Dreamer, an RL agent that learns a world model from images and uses it to learn long-sighted behaviors. Dreamer leverages its world model to efficiently learn behaviors via backpropagation through model predictions. By learning to compute *compact model states* from raw images, the agent is able to efficiently learn from thousands of predicted sequences in parallel using just one GPU. Dreamer achieves a new state-of-the-art in performance, data efficiency and computation time on a benchmark of 20 continuous control tasks given raw image inputs. To stimulate further advancement of RL, we are releasing the source code to the research community.

How Does Dreamer Work?
Dreamer consists of three processes that are typical for model-based methods: learning the world model, learning behaviors from predictions made by the world model, and executing its learned behaviors in the environment to collect new experience. To learn behaviors, Dreamer uses a value network to take into account rewards beyond the planning horizon and an actor network to efficiently compute actions. The three processes, which can be executed in parallel, are repeated until the agent has achieved its goals:

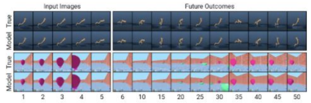
The three processes of the Dreamer agent. The world model is learned from past experience. From predictions of this model, the agent then learns a value network to predict future rewards and an actor network to select actions. The actor network is used to interact with the environment.

Learning the World Model
Dreamer leverages the PlaNet world model, which predicts outcomes based on a sequence of *compact model states* that are computed from the input images, instead of directly predicting from one image to the next. It automatically learns to produce model states that represent concepts helpful for predicting future outcomes, such as object types, positions of objects, and the interaction of the objects with their surroundings. Given a sequence of images, actions, and rewards from the agent's dataset of past experience, Dreamer learns the world model as shown:



Dreamer learns a world model from experience. Using past images (o_1–o_3) and actions (a_1–a_2), it computes a sequence of compact model states (green circles) from which it reconstructs the images ($\hat{o}_1$–$\hat{o}_3$) and predicts the rewards ($\hat{r}_1$–$\hat{r}_3$).

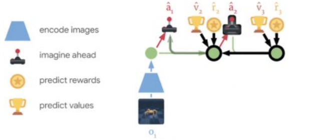
An advantage to using the PlaNet world model is that predicting ahead using compact model states instead of images greatly improves the computational efficiency. This enables the model to predict thousands of sequences in parallel on a single GPU. The approach can also facilitate generalization, leading to accurate long-term video predictions. To gain insights into how the model works, we can visualize the predicted sequences by decoding the compact model states back into images, as shown below for a task of the DeepMind Control Suite and for a task of the DeepMind Lab environment:



Predicting ahead using compact model states enables long-term predictions in complex environments. Shown here are two sequences that the agent has not encountered before. Given five input images, the model reconstructs them and predicts the future images up to time step 50.

Efficient Behavior Learning
Previously developed model-based agents typically select actions either by planning through many model predictions or by using the world model in place of a simulator to reuse existing model-free techniques. Both designs are computationally demanding and do not fully leverage the learned world model. Moreover, even powerful world models are limited in how far ahead they can accurately predict, rendering many previous model-based agents shortsighted. Dreamer overcomes these limitations by learning a value network and an actor network via backpropagation through predictions of its world model.

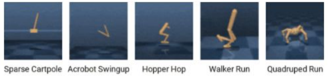
Dreamer efficiently learns the actor network to predict successful actions by propagating gradients of rewards backwards through predicted state sequences, which is not possible for model-free approaches. This tells Dreamer how small changes to its actions affect what rewards are predicted in the future, allowing it to refine the actor network in the direction that increases the rewards the most. To consider rewards beyond the prediction horizon, the value network estimates the sum of future rewards for each model state. The rewards and values are then backpropagated to refine the actor network to select improved actions:



Dreamer learns long-sighted behaviors from predicted sequences of model states. It first learns the long-term value ($\hat{v}_2$–$\hat{v}_3$) of each state, and then predicts actions ($\hat{a}_1$–$\hat{a}_2$) that lead to high rewards and values by backpropagating them through the state sequence to the actor network.

Dreamer differs from PlaNet in several ways. For a given situation in the environment, PlaNet searches for the best action among many predictions for different action sequences. In contrast, Dreamer side-steps this expensive search by decoupling planning and acting. Once its actor network has been trained on predicted sequences, it computes the actions for interacting with the environment without additional search. In addition, Dreamer considers rewards beyond the planning horizon using a value function and leverages backpropagation for efficient planning.

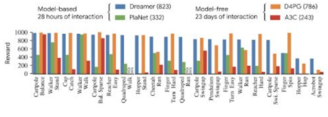
Performance on Control Tasks
We evaluated Dreamer on a standard benchmark of 20 diverse tasks with continuous actions and image inputs. The tasks include balancing and catching objects, as well as locomotion of various simulated robots. The tasks are designed to pose a variety of challenges to the RL agent, including difficult to predict collisions, sparse rewards, chaotic dynamics, small but relevant objects, high degrees of freedom, and 3D perspectives:



Dreamer learns to solve 20 challenging continuous control tasks with image inputs, 5 of which are displayed here. The visualizations show the same 64x64 images that the agent receives from the environment.

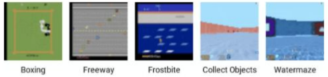
We compare the performance of Dreamer to that of PlaNet, the previous best model-based agent, the popular model-free agent, A3C, as well as the current best model-free agent on this benchmark, D4PG, which combines several advances of model-free RL. The model-based agents learn efficiently in under 5 million frames, corresponding to 28 hours inside the simulation. The model-free agents learn more slowly and require 100 million frames, corresponding to 23 days inside the simulation.

On the benchmark of 20 tasks, Dreamer outperforms the best model-free agent (D4PG) with an average score of 823 compared to 786, while learning from 20 times fewer environment interactions. Moreover, it exceeds the final performance of the previously best model-based agent (PlaNet) across almost all of the tasks. The computation time of 16 hours for training Dreamer is less than the 24 hours required for the other methods. The final performance of the four agents is shown below:



Dreamer outperforms the previous best model-free (D4PG) and model-based (PlaNet) methods on the benchmark of 20 tasks in terms of final performance, data efficiency, and computation time.

In addition to our main experiments on continuous control tasks, we demonstrate the generality of Dreamer by applying it to tasks with discrete actions. For this, we select Atari games and DeepMind Lab levels that require both reactive and long-sighted behavior, spatial awareness, and understanding of visually more diverse scenes. The resulting behaviors are visualized below, showing that Dreamer also efficiently learns to solve these more challenging tasks:



Dreamer learns successful behaviors on Atari games and DeepMind Lab levels, which feature discrete actions and visually more diverse scenes, including 3D environments with multiple objects.

Conclusion
Our work demonstrates that learning behaviors from sequences predicted by world models alone can solve challenging visual control tasks from image inputs, surpassing the performance of previous model-free approaches. Moreover, Dreamer demonstrates that learning behaviors by backpropagating value gradients through predicted sequences of compact model states is successful and robust, solving a diverse collection of continuous and discrete control tasks. We believe that Dreamer offers a strong foundation for further pushing the limits of reinforcement learning, including better representation learning, directed exploration with uncertainty estimates, temporal abstraction, and multi-task learning.

Dream to Control
Learning Behaviors by Latent Imagination

Blog post, code, videos, paper:

danijar.com/dreamer