@doomie

World Models

Dumitru Erhan
Staff Research Scientist @ Google Brain

presenting on behalf of many colleagues!

March 1, 2021 @ University of Colorado Boulder

About this talk

- Major progress in machine learning / computer vision:
    - Stuff that works: image classification, object detection, Go, Atari etc
    - **Hypothesis**: works because of lots/infinite well-labeled data + deep nets + compute.
- But what we really want:
    - Intelligent agents that can learn from **little data**
    - And that can adapt to new settings, scenarios, goals, tasks **quickly**
- Premises:
    - By learning how the world works, agents can **imagine the effects of their actions**
    - How to learn how the world works? Use **generative models!** (this talk)
    - Another popular way: meta-learning (not covered in this talk)

Mission of “world models”

- Learn a model of the world from observation data and do useful things with it
- In other words: representation learning of **environments**.
- What are examples of “useful things”?
    - Learn RL policies from very few rewards
    - Generalize to new tasks/domains/objects/instances quickly
    - Do one/zero-shot learning of behavior
    - Learn models in sim / collected interactions, quickly adapt them to real robot
    - Add ability to “debug” agents

A “world model”?

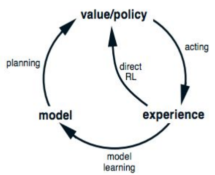
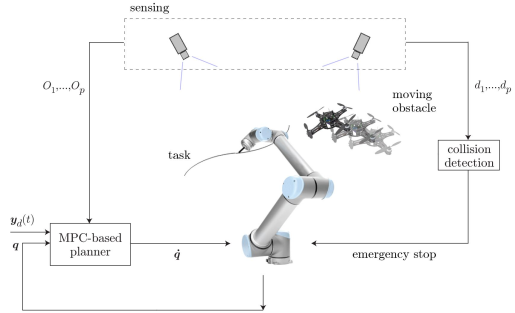


Figure 8.1: Relationships among learning, planning, and acting.



- Premise & hypothesis of model-based RL is **not new at all**
- Variety of classical control algorithms for robotics are in fact model-based.
- But they operate on ground-truth states, which cannot be observed in practice
- In a world model, we **don’t make this assumption**
- The agent:
    - observes the world (via pixels)
    - interacts with the world (via actions/torques)
    - observes the consequence of its actions
    - optionally, receives a reward.
- Premise: we can learn a lot about the world by predicting future (images/features/rewards)
- But how to do it?

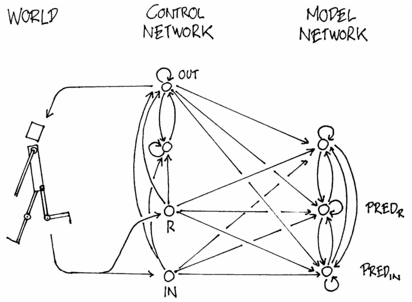
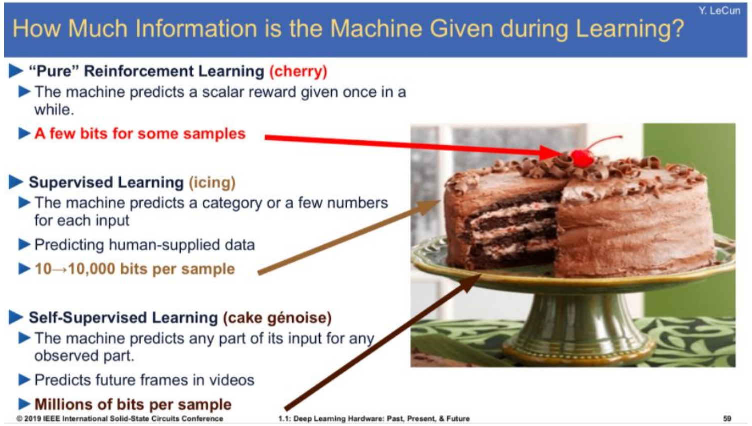


Image: Schmidhuber 1990

Predictive learning



Y. LeCun

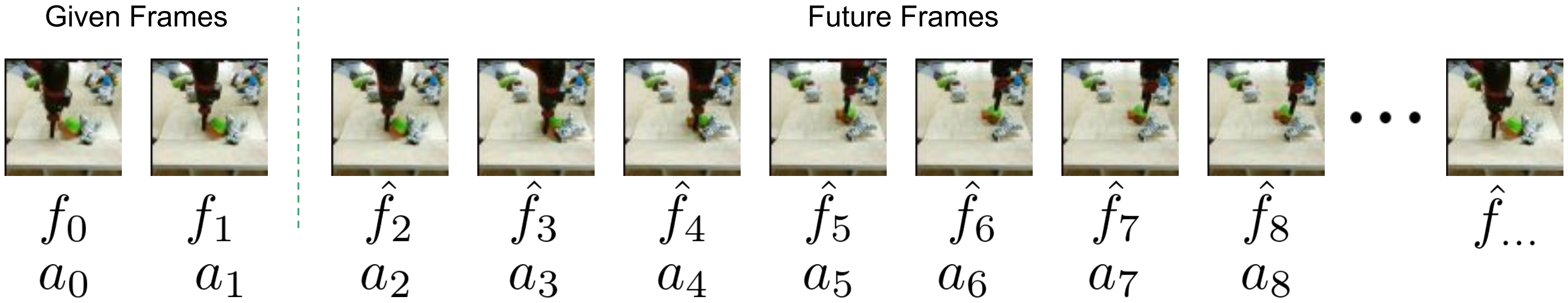
How Much Information is the Machine Given during Learning?

- **“Pure” Reinforcement Learning (cherry)**
  - The machine predicts a scalar reward given once in a while.
  - **A few bits for some samples**
- **Supervised Learning (icing)**
  - The machine predicts a category or a few numbers for each input
  - Predicting human-supplied data
  - **10→10,000 bits per sample**
- **Self-Supervised Learning (cake génoise)**
  - The machine predicts any part of its input for any observed part.
  - Predicts future frames in videos
  - **Millions of bits per sample**

© 2019 IEEE International Solid-State Circuits Conference 1.1: Deep Learning Hardware: Past, Present, & Future 59

- Yann LeCun calls this “predictive learning”
  - Also “self-supervised learning”
- (Supposedly) humans and other animals do this:
  - They predict the effect of their actions
  - e.g. “try it in your head” before acting.
- What if we had a way to do that directly? Let’s say we observe:
  - the present: a sequence of events (frames), call it **state**
  - the **action** that the agent takes (“move arm to coordinate (x,y,z)”)
  - the **future state** once the action complets.
- We could learn a model that tries to infer $\mathbf{s_{t+1}}$ from $\mathbf{[s_t, a_t]}$
- In general, this is hard: state can be a video, action can be continuous multi-dimensional, future is non-deterministic etc.

Use generative & predictive models!



Given Frames

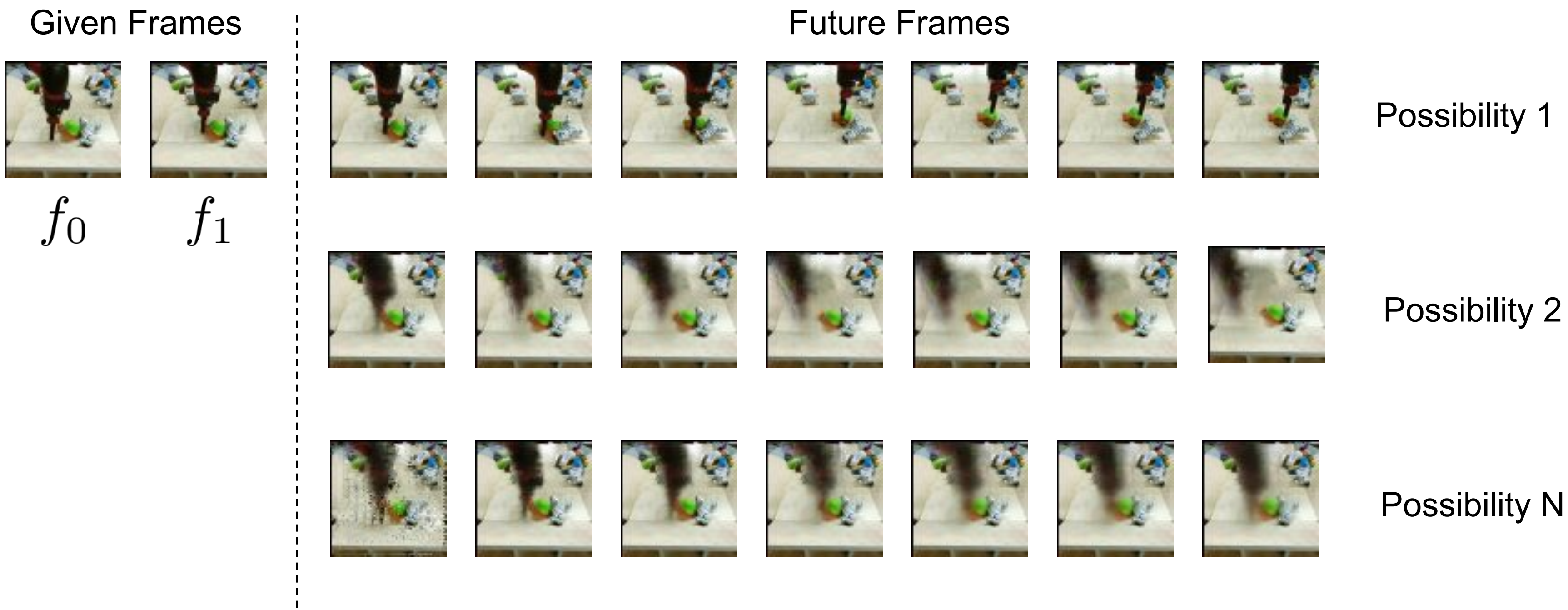
Future Frames

f_0 f_1 $\hat{f}_2$ $\hat{f}_3$ $\hat{f}_4$ $\hat{f}_5$ $\hat{f}_6$ $\hat{f}_7$ $\hat{f}_8$ $\cdots$ $\hat{f}_{\dots}$

a_0 a_1 a_2 a_3 a_4 a_5 a_6 a_7 a_8

- Predict the future: directly (video prediction) or indirectly (latent space dynamics)
- Big challenges: uncertainty, believable predictions, long-horizon predictions.
- Lots of work in our group as part of: SV2P, PlaNet, EPVA, DS-GAN, FVD
- 2019 videogen SOTA: generate 10-15 believable 64x64 frames in a contrived setting.

Problem: Stochasticity



Given Frames

Future Frames

f_0 f_1

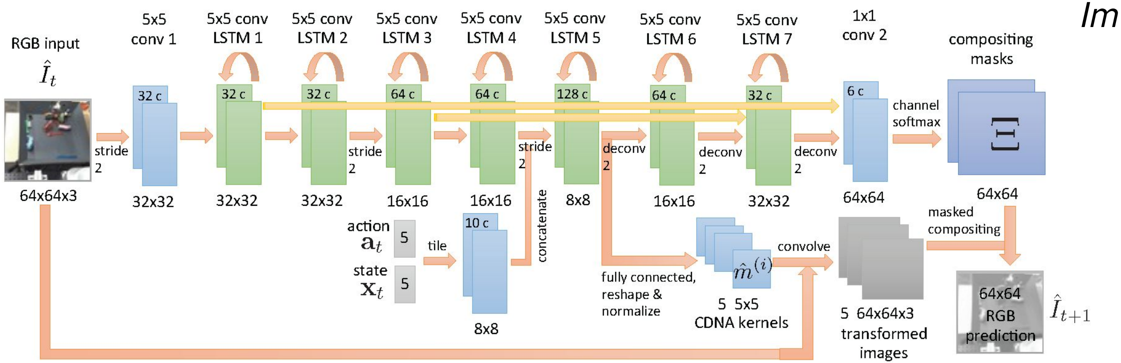
Possibility 1

Possibility 2

Possibility N

convolutional LSTMs

Image: Finn et al (2016)

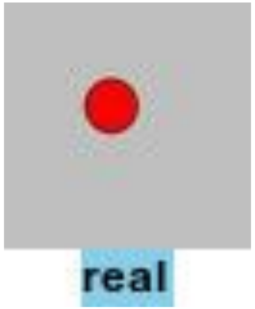


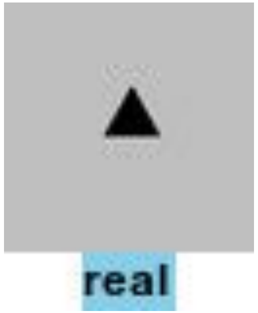
action-conditioned

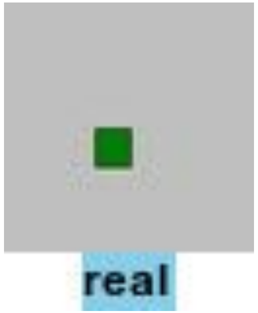
flow prediction

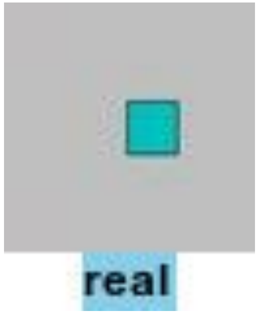
One big problem: model makes mostly deterministic predictions!

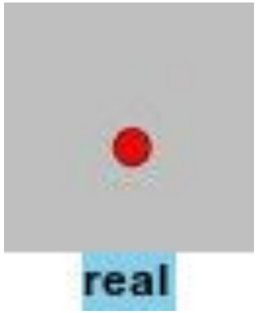
Stochastic Shapes Dataset

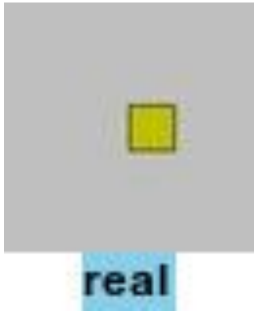




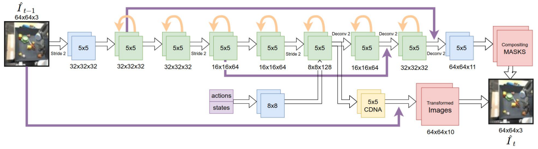


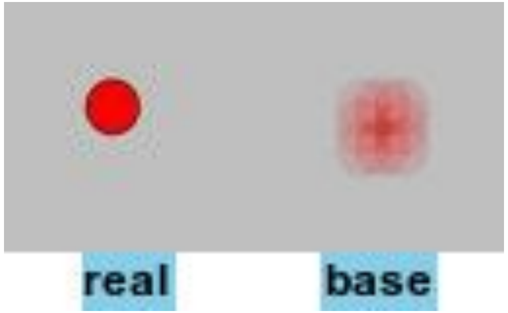


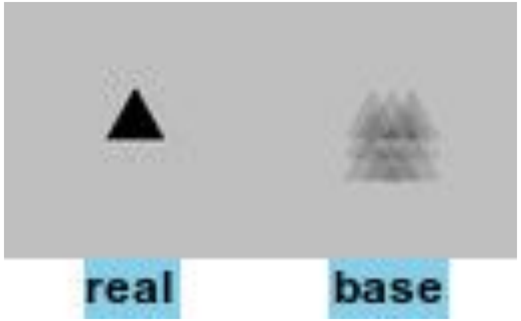


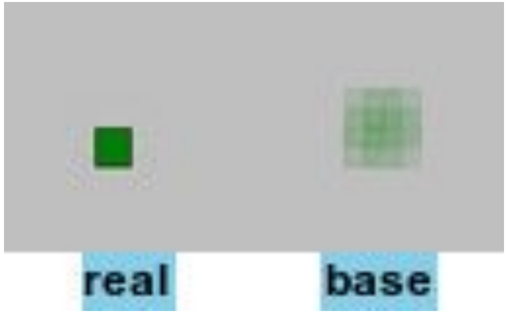


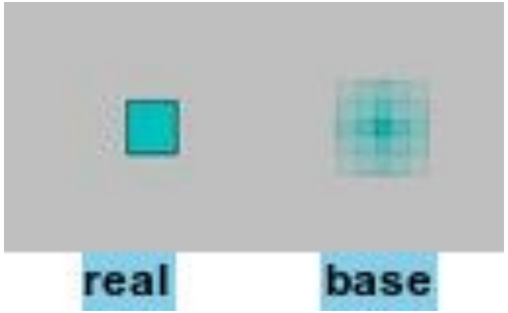
Oops: probably not very useful

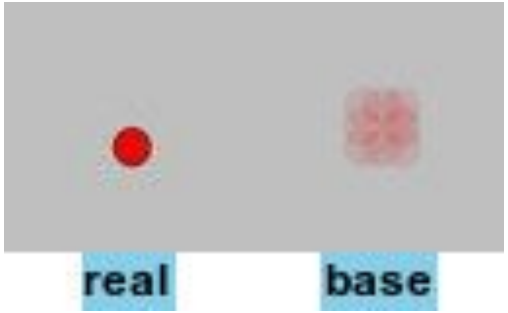


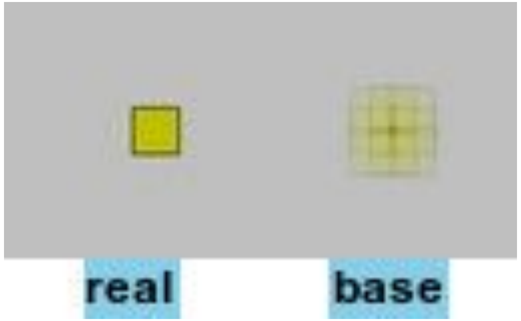






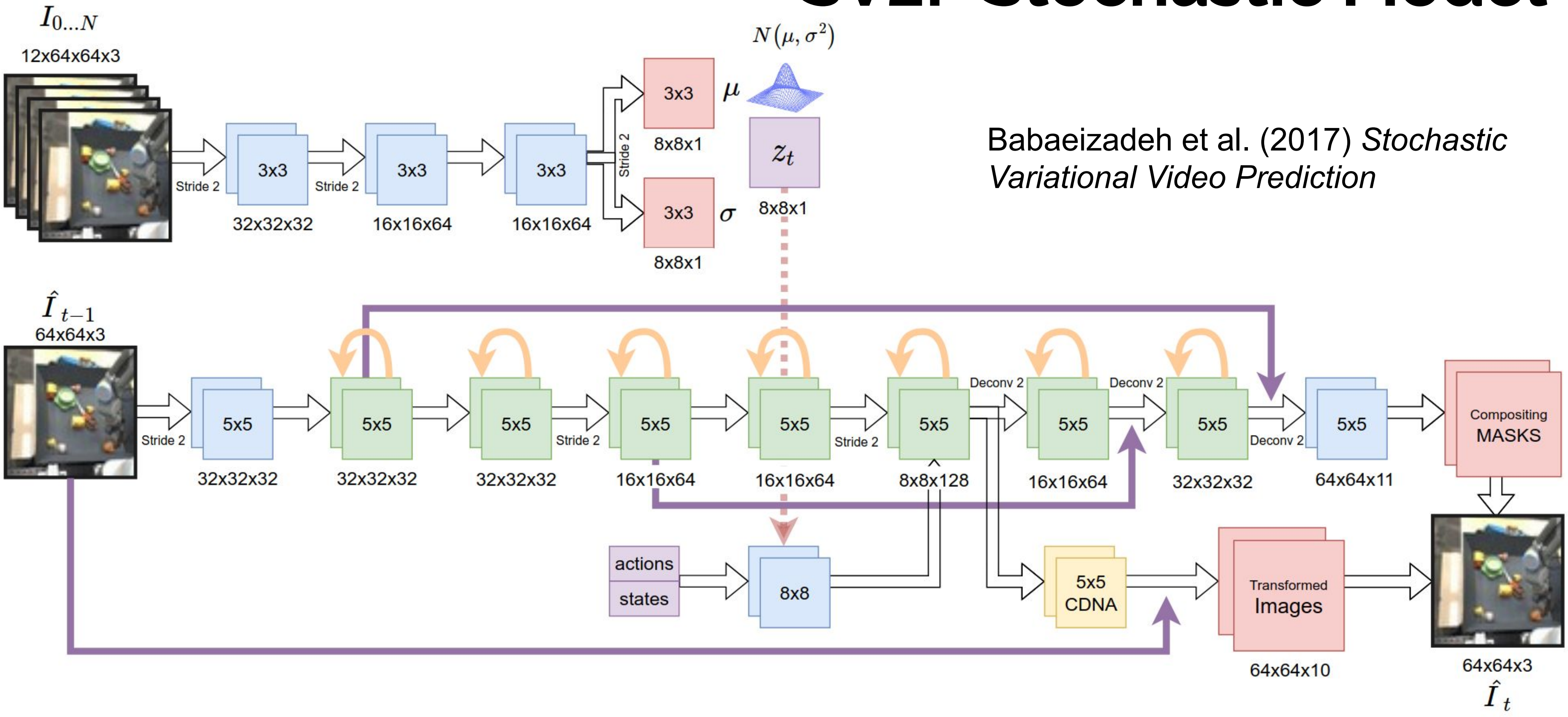






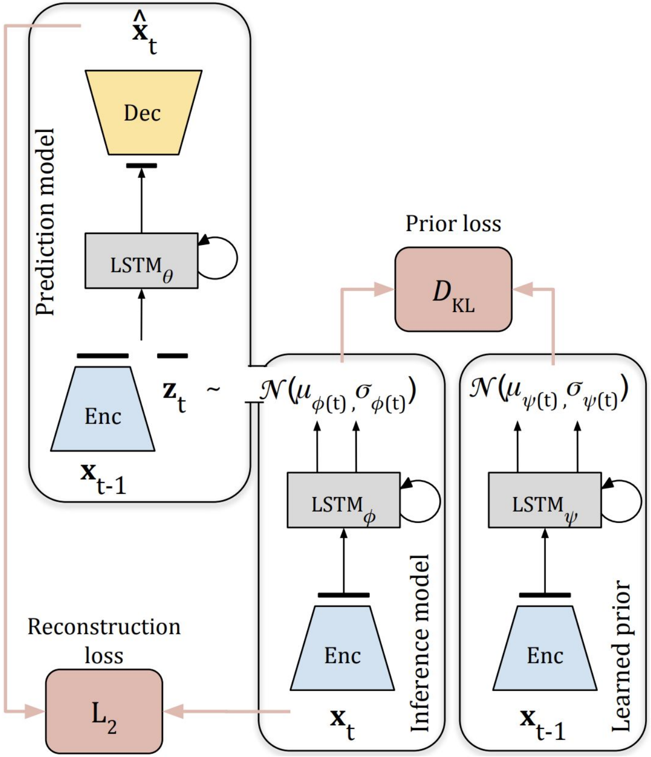
SV2P Stochastic Model

Babaeizadeh et al. (2017) *Stochastic Variational Video Prediction*



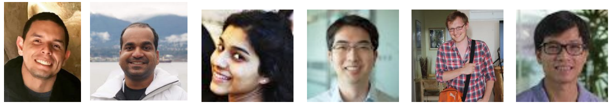
Ruben Villegas, Arkanath Pathak, Harini Kannan, Dumitru Erhan, Quoc V. Le, Honglak Lee

High-fidelity video generation



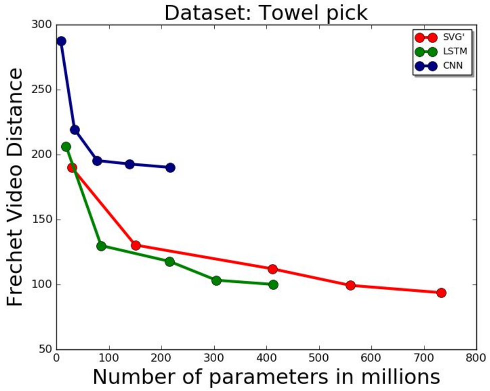
- Can we do better in terms of video quality? While leveraging our resources?
- Gist: take the Denton et al. (2018) SVG model, increase capacity
- Variations:
  - Deterministic LSTM (remove stochasticity)
  - CNN (removes recurrence)
- Evaluation:
  - Frechet Video Distance (FVD)
  - PSNR, SSIM, VGG-Cosine
  - Mturk human eval
- Biggest model: 750M params (300M in LSTMs). Baseline: 30M

Figure credit: Denton et al., 2018



High-fidelity video generation: NeurIPS 2019

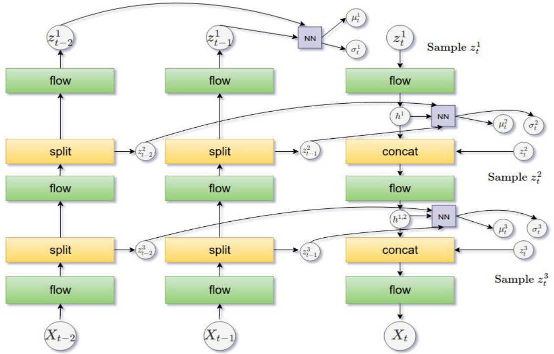
- Can we do better in terms of video quality? While leveraging our resources?
- Denton et al. (2018) model + increase capacity + make model simpler = Big WIN



Generating higher-resolution (128x128) videos

High Fidelity Video Prediction with Large Stochastic Recurrent Neural Networks Villegas et al (2019)

VideoFlow: ICLR 2020



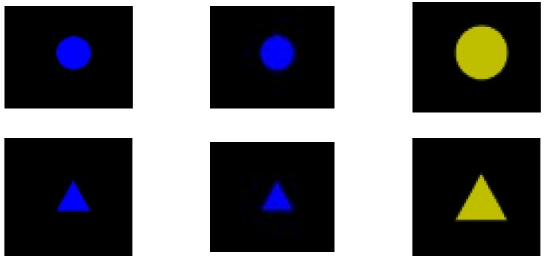
Generative model of videos based on the Glow architecture.

Normalizing flows: parallel generation, exact inference, tractable likelihood

Model	Bits-per-pixel
VideoFlow	1.87
SAVP-VAE	< 6.73
SV2P	< 6.78

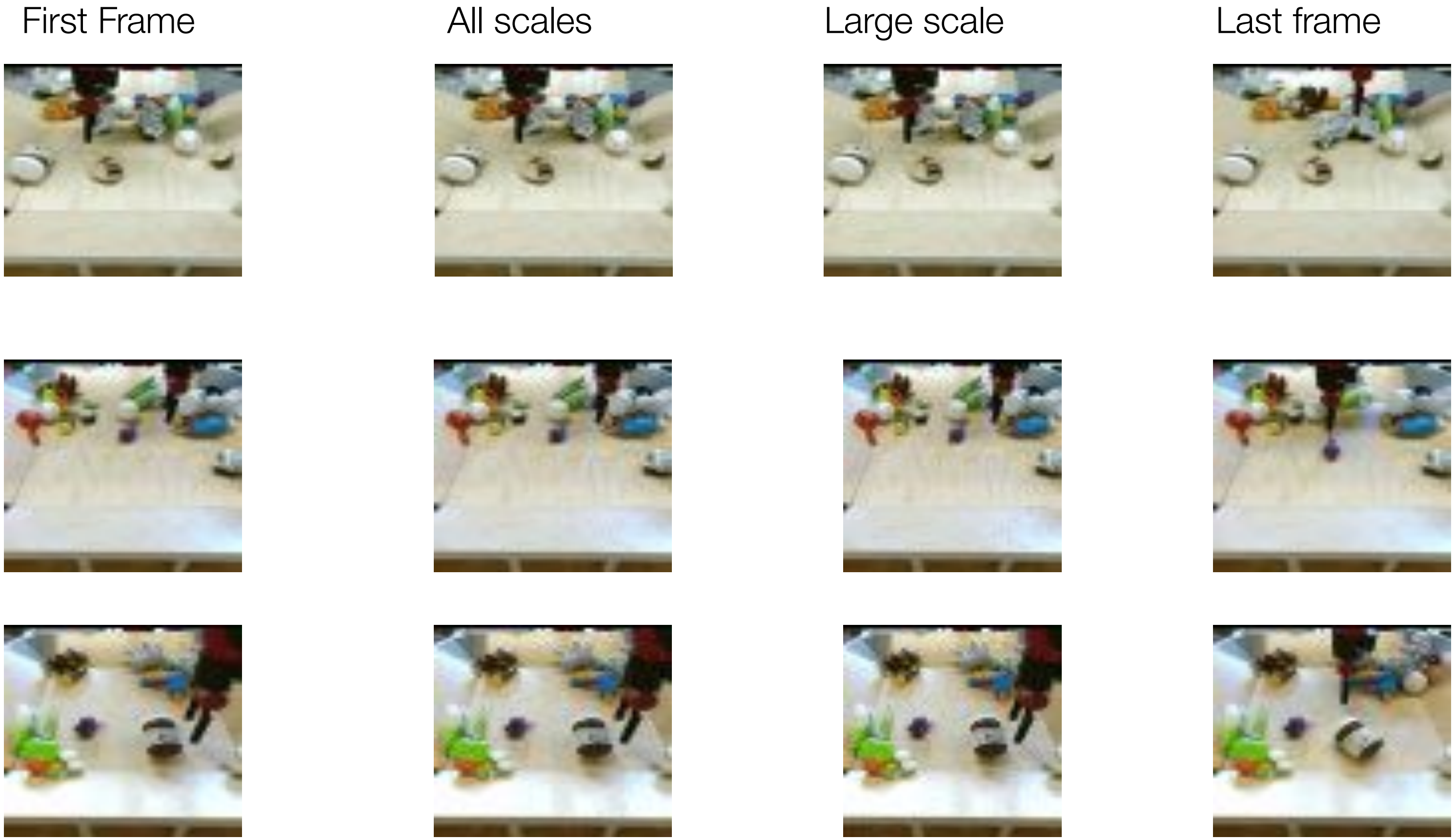
Diverse rollouts

Fun latent space interpolations!



VideoFlow: A Flow-Based Generative Model for Video by Kumar et al (2020)

Latent Space Interpolations



First Frame

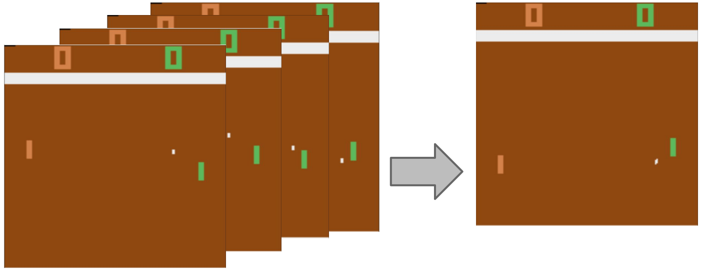
All scales

Large scale

Last frame

One useful application: Model-Based RL for Atari

- ~2018 State of the art on solving Atari
  - PPO: 8M frames = **92 days**
  - RAINBOW: 1.2M frames = **14 days**
- Sample efficiency is important!
  - Collecting experience can be costly
  - Robots are expensive, break, etc.
- We want RL to work in online scenarios with little experience and offline.
- How can we use every pixel of every frame for supervision?
  - Action-conditional video prediction as "world model"



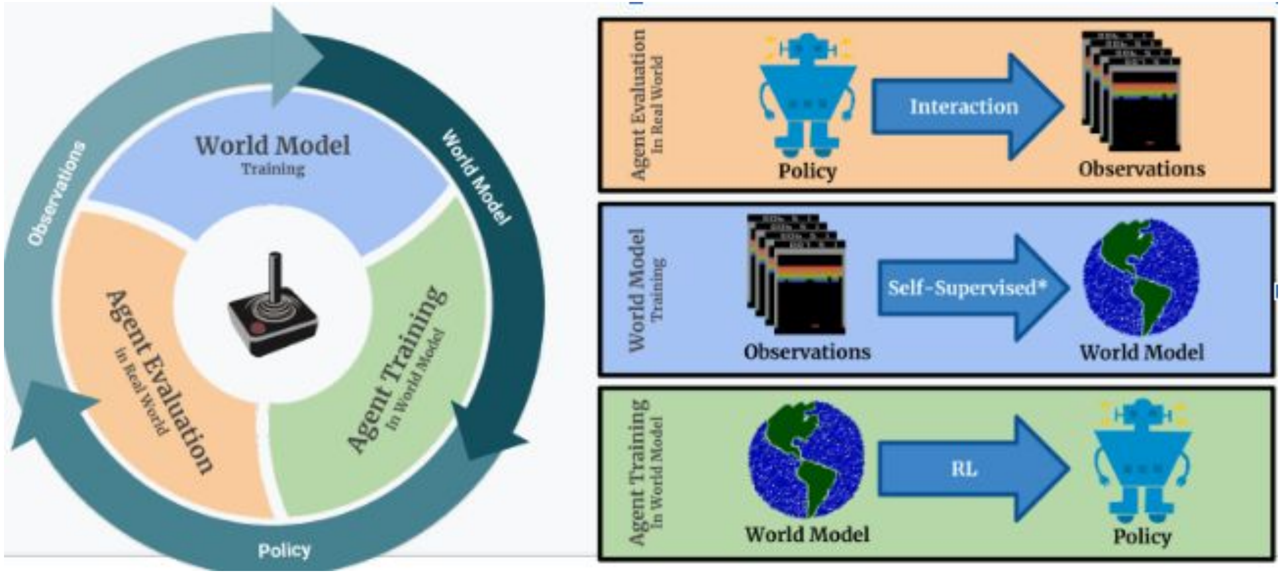
Goal: sample efficiency

Łukasz Kaiser, Mohammad Babaeizadeh, Piotr Miłos, Błazej Osinski, Roy H Campbell, Konrad Czechowski, Dumitru Erhan, Chelsea Finn, Piotr Kozakowski, Sergey Levine, Afroz Mohiuddin, Ryan Sepassi, George Tucker, Henryk Michalewski

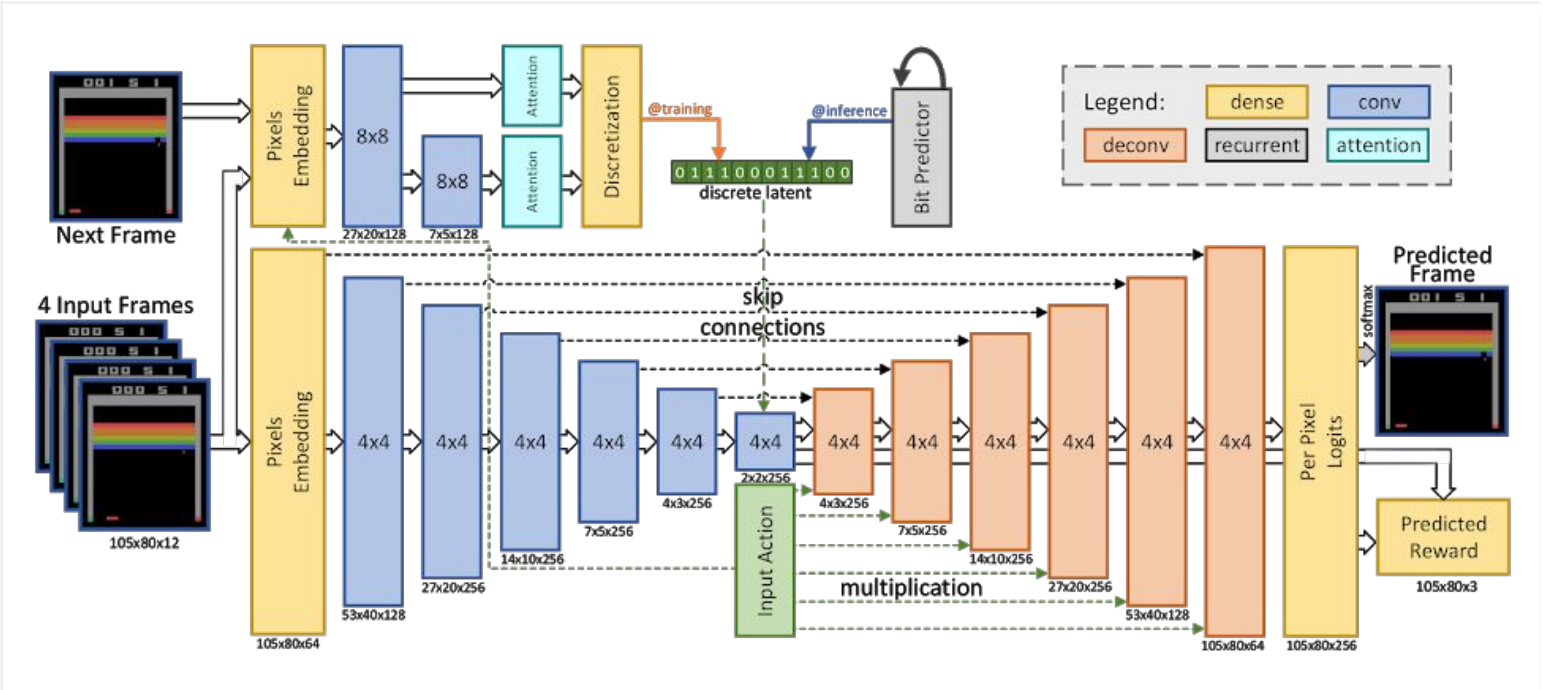
Model-Based Reinforcement Learning for Atari by Kaiser et al (2020)

SimPLe (ICLR 2020): do well on Atari

- **Precise goal**: do well enough on Atari with 100k rewards
- **Main premise** of SimPLe algorithm is to alternate between
  - learning a world model of how the game behaves
  - using that model to optimize a policy (with model-free RL) within the simulated game environment

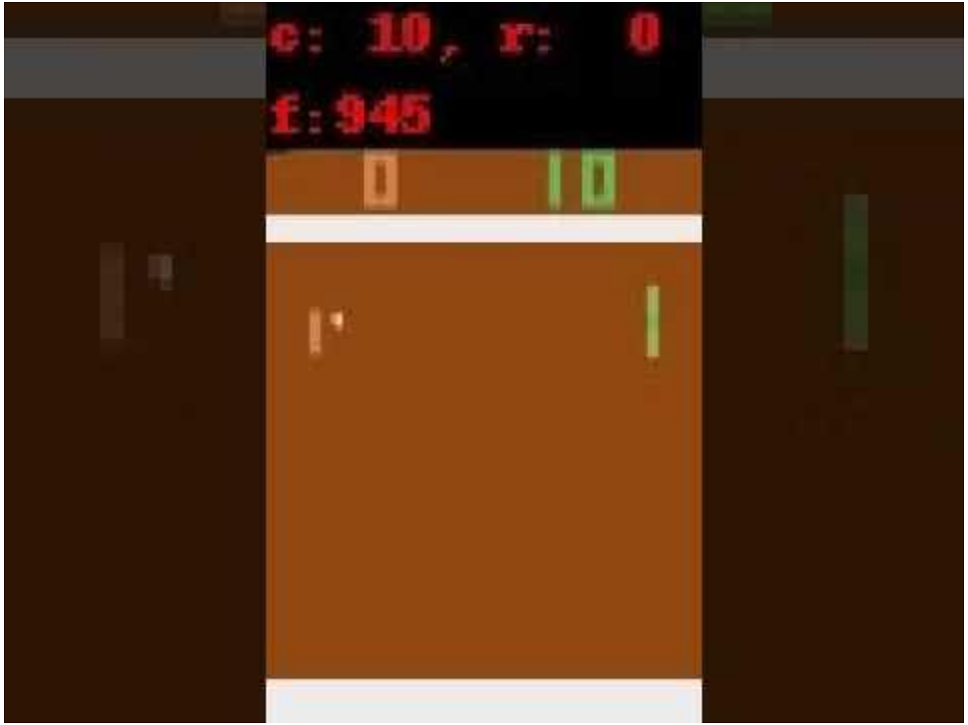


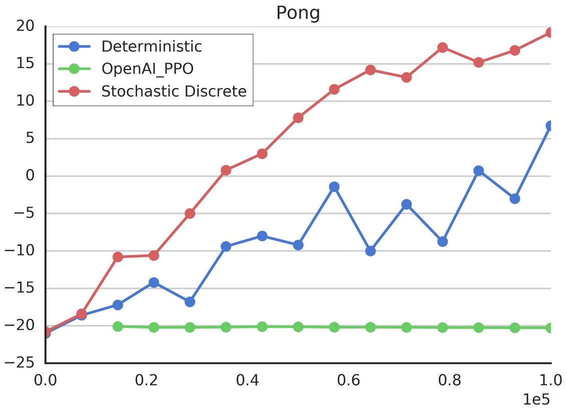
Details of the world model



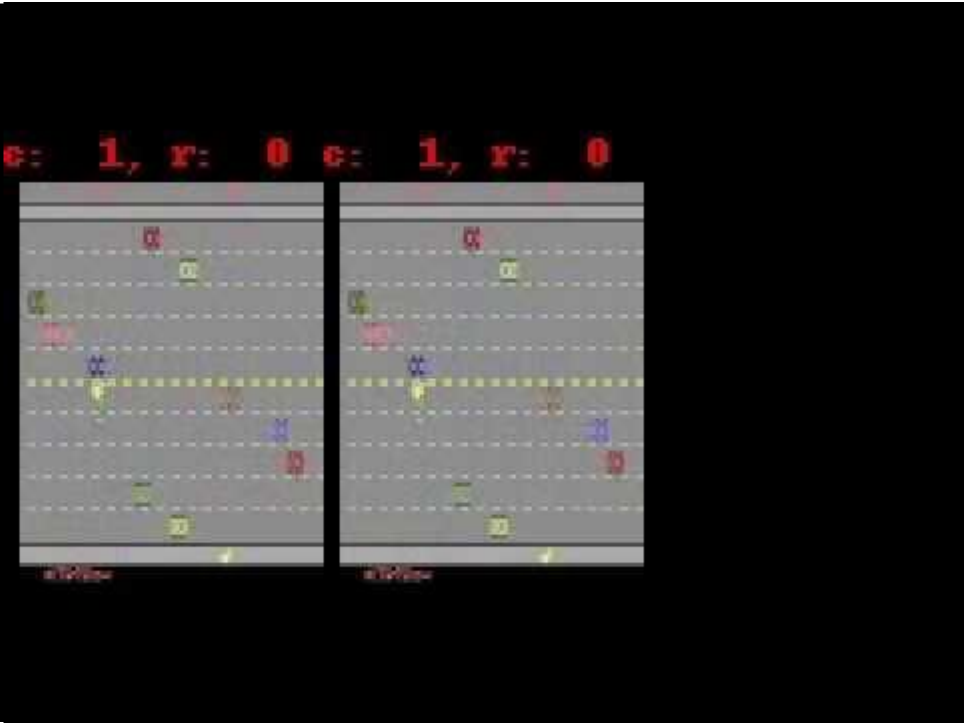
Successes: Pong

Left: predicted, middle: groundtruth, right: difference

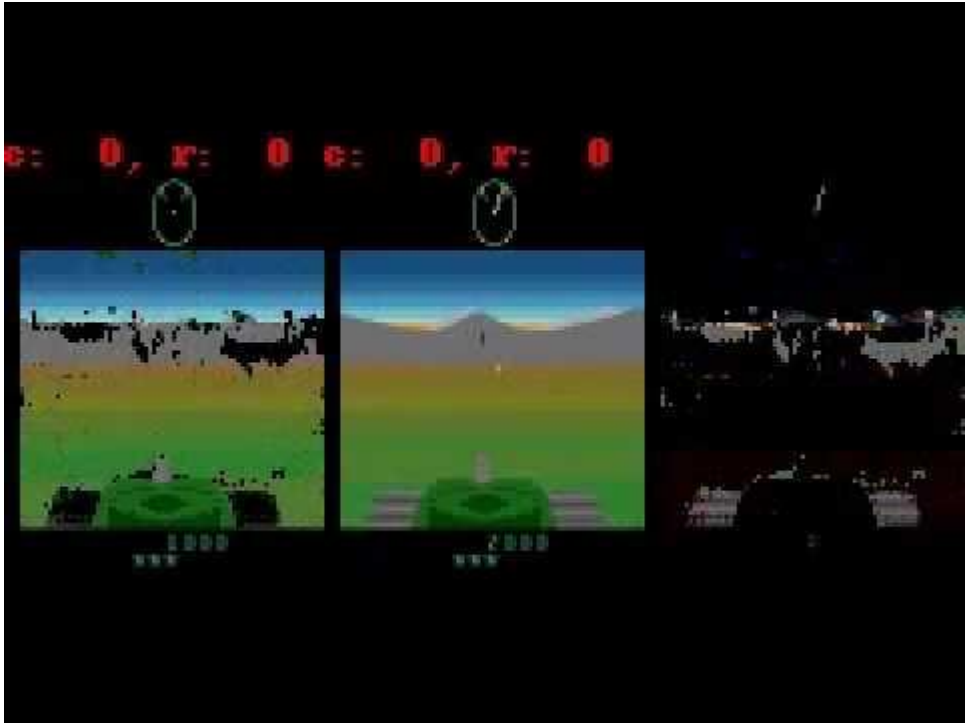




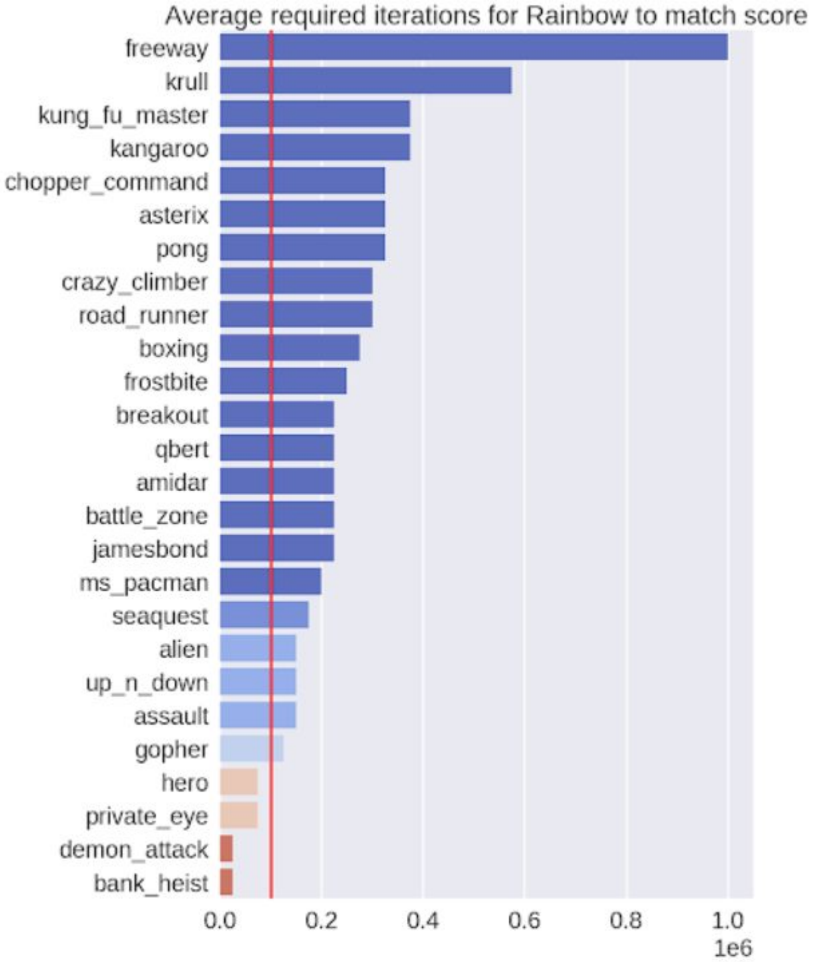
More successes

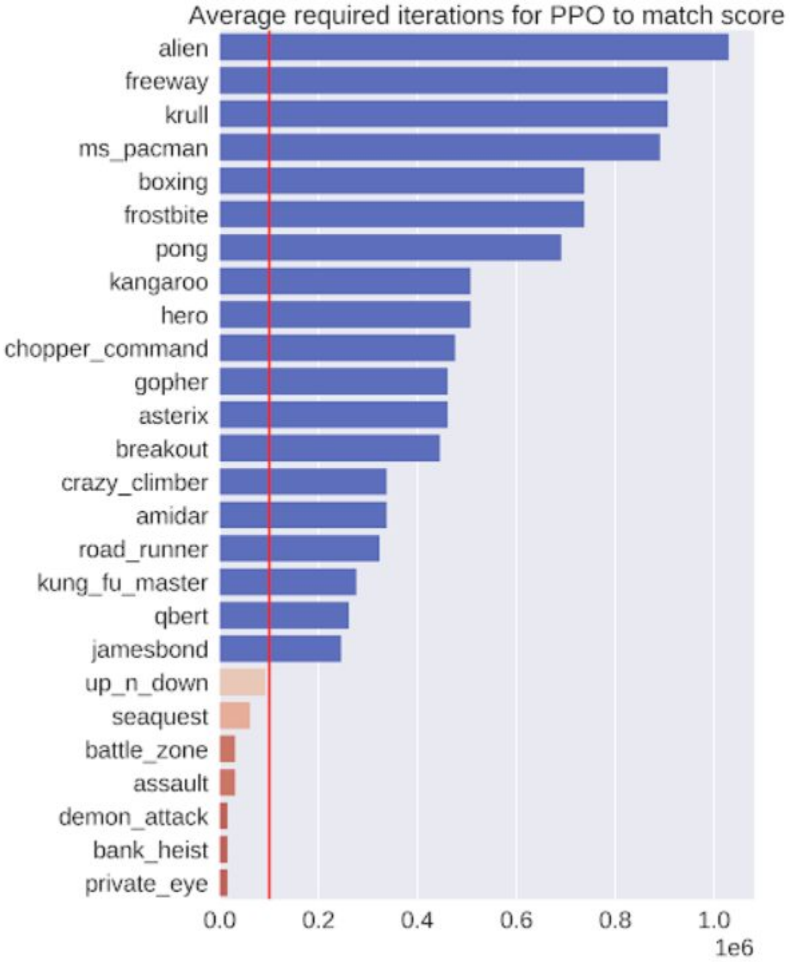


Stuff that can go wrong



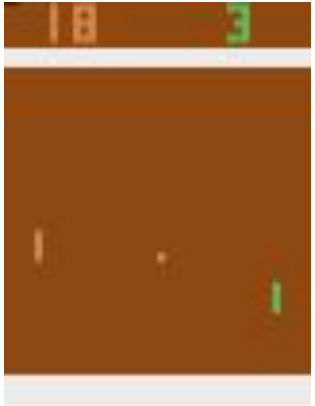
Results


Average required iterations for Rainbow to match score
freeway
krull
kung_fu_master
kangaroo
chopper_command
asterix
pong
crazy_climber
road_runner
boxing
frostbite
breakout
qbert
amidar
battle_zone
jamesbond
ms_pacman
seaquest
alien
up_n_down
assault
gopher
hero
private_eye
demon_attack
bank_heist
0.0
0.2
0.4
0.6
0.8
1.0
1e6


Average required iterations for PPO to match score
alien
freeway
krull
ms_pacman
boxing
frostbite
pong
kangaroo
hero
chopper_command
gopher
asterix
breakout
crazy_climber
amidar
road_runner
kung_fu_master
qbert
jamesbond
up_n_down
seaquest
battle_zone
assault
demon_attack
bank_heist
private_eye
0.0
0.2
0.4
0.6
0.8
1.0
1e6

MBRL-for-Atari thoughts

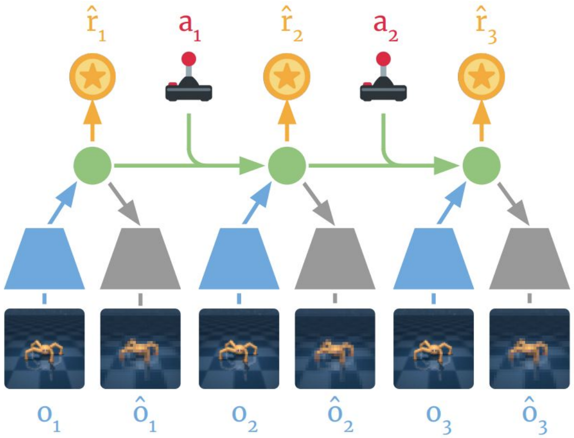
- Results could be better at convergence
- Next ambitious goal: apply the approach to domains other than Atari
- Can (should) we predict something other than pixels?
  - Figure out how to predict the important stuff



o

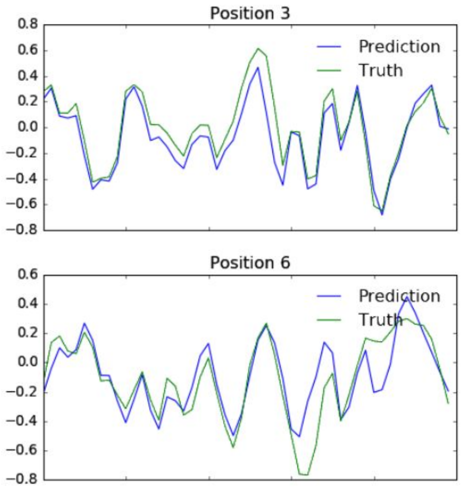
PlaNet: ICML 2019

For visual tasks, predicting forward in compact latent space reduces accumulating errors, memory footprint, computation

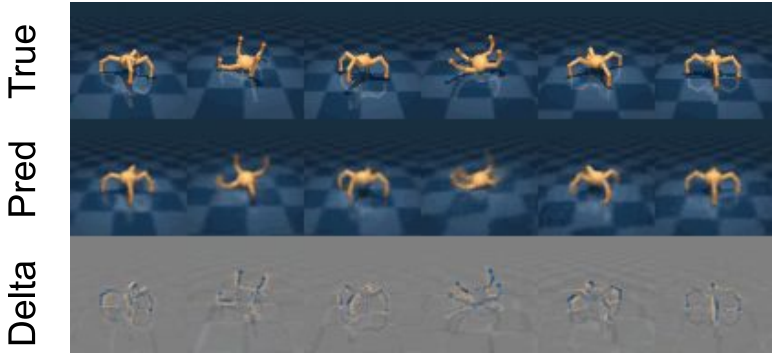


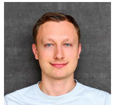
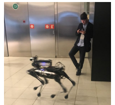
Learn $p(o_t \mid s_t)$ by pixel reconstruction or inverse $p(s_t \mid o_t)$ by CPC

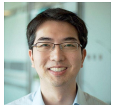
Latent Dynamics Model



Recovers true system states



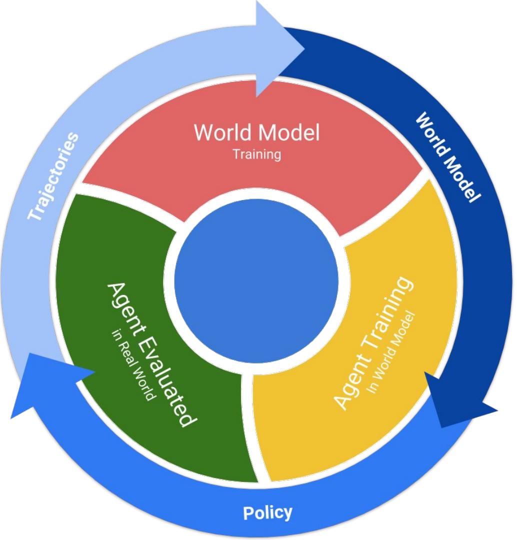


Hierarchical Visual Planning with subgoal prediction

Main idea: high-level model that **predicts** where the low-level controller will end up

HVF (Nair and Finn 2019)	55% success rate
HVPC (proposed)	**72% success rate**

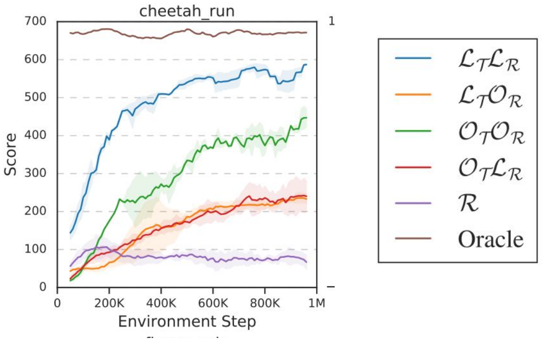
A world models playground

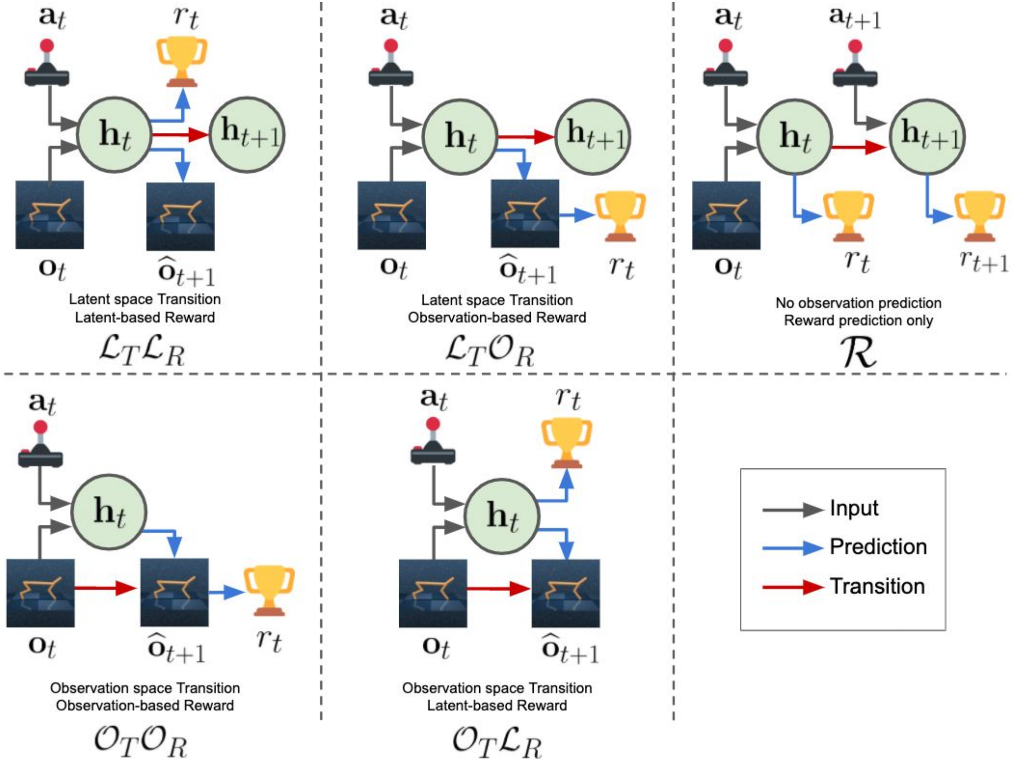


- General idea: a researcher typically has a new idea for a:
  - Model or environment or planner
  - But doesn't have the infrastructure/know-how to try the whole pipeline
- We want to enable this use-case: “bring-your-own piece of the puzzle”
- **Solution**: lightweight, platform-agnostic API
- **Goal**: simple interface that enables a researcher to use it within a day.
  - A few high-level methods: simulate(), observe(), predict(), save_trajectories()
  - Implements the above training loop!

Deconstructing World Models (in review)

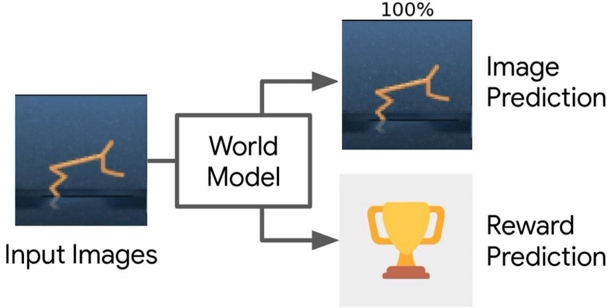
- Codebase allowing comparison of methods on equal footing.
- Major themes:
  - Is predicting images useful?
  - How important is reward prediction accuracy?
  - Modeling dynamics in latent vs observation space

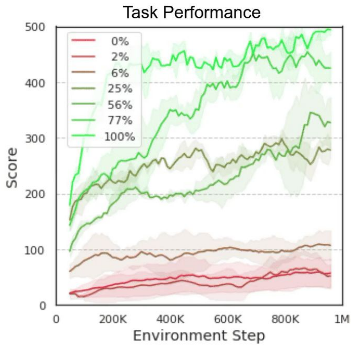




Deconstructing World Models (in review)

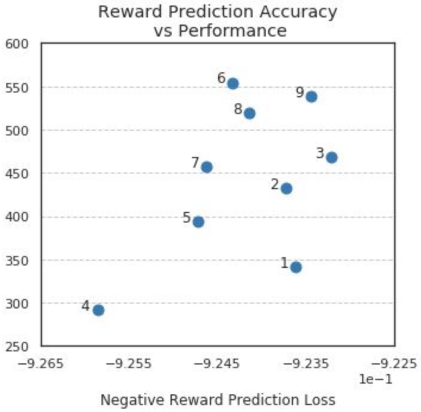
The more pixels you predict, the better the agent becomes.

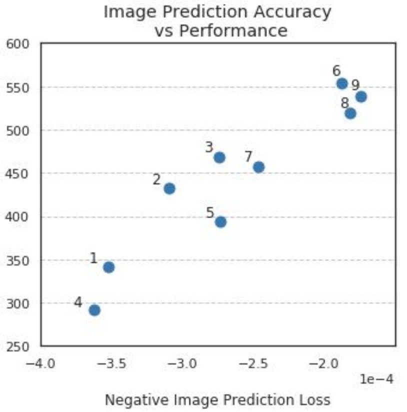




Strong correlation between image prediction accuracy and task performance.

Not so strong between reward prediction and performance!





Summary

- Despite amazing progress & hype, computer vision not really “solved”.
- Anything that deviates from basic classification template is hard.
- We should be building **predictive agents** that can learn from **self-supervised** interactions with the world.
- Building **unsupervised representations** of the world is an open problem.
- Open questions:
    - Can we deal with uncertainty?
    - How to do efficient planning under uncertainty?
- Current projects attempt to understand the role of learning a model for: generalization, domain adaptation, zero/one-shot learning, short vs long horizon; **not just end performance**

Thanks!

- Library is released @ **github.com/google-research/world_models**
  - One day: many of the collected trajectories
- **Current focus**: scaling our work beyond DeepMind Control Suite. Research-wise, the **long-term goal** is the same. We want to understand how we can create sample-efficient agents
  - that can bootstrap themselves with little expert data
  - and that can solve multiple tasks
  - in visually complex and diverse environments.
- Lots of collaborators to thank: Chelsea Finn, Sergey Levine, Harini Kannan, Mohammad Saffar, Mohammad Babaeizadeh, Danijar Hafner, Suraj Nair, Thanard Kurutach and many others!
- Reach me at: **dumitru@google.com** or 🐦 @doomie