CS-461

Foundation Models and Generative AI

World Models and Generative World Modeling

Charlotte Bunne, Fall Semester 2025/26

Announcements

- Next week we have a guest lecture!

World Foundation Models for Robotics

Hang Zhao
Tsinghua University

- Lecture as usual **in PO 01**
  on Tuesday, ~~1~~ 2 - 3 pm

- **Assignment 2** on test-time learning is due next week!

  Deadline: Wednesday, December 3 at 23:59.

Last Week: **Test-Time Training**

Pre-Training

Compress world knowledge

Post-Training

Specialize in certain domains and behaviors

Test-Time-Training

Models continue to learn in deployment environments

Week 11's Exercise Sheet

No pen-and-paper exercises this week!

This Week's Code Demonstration

Improving Summarization with In-Context Learning

Code Notebook 10 · Task 1

Work on improving extreme summarization capabilities of LLMs. Showcasing instruction-following via few-shot in-context learning.

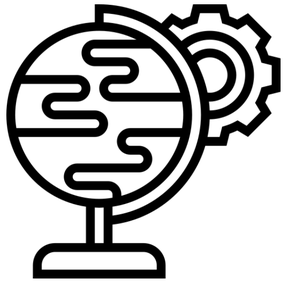
Quantization in LLMs

Code Notebook 10 · Task 2

Working on understanding quantization schemes (Affine INT8) for tensors and weights, post-training quantization (PTQ) of neural network weights, showcasing performance downgrade after PTQ, loading a pre-trained LLM and quantizing it to INT8.

More in the exercise session: discussion on current quantization schemes.

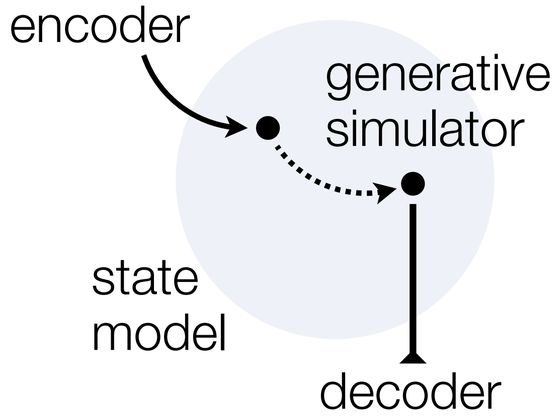
This Week: **World Models**



WORLD MODEL

A world model is an AI system that can simulate and predict how the world might change given current conditions and potential actions or events.

- World models are **generative AI systems** that learn *internal representations* of real-world environments, including their physics, spatial dynamics, and causal relationships (at least, the basic ones), from diverse input data.
- They **use these learned representations to predict future states**, simulate sequences of actions internally, and support sophisticated planning and decision-making without needing continuous real-world experimentation.



From Simple to World Models

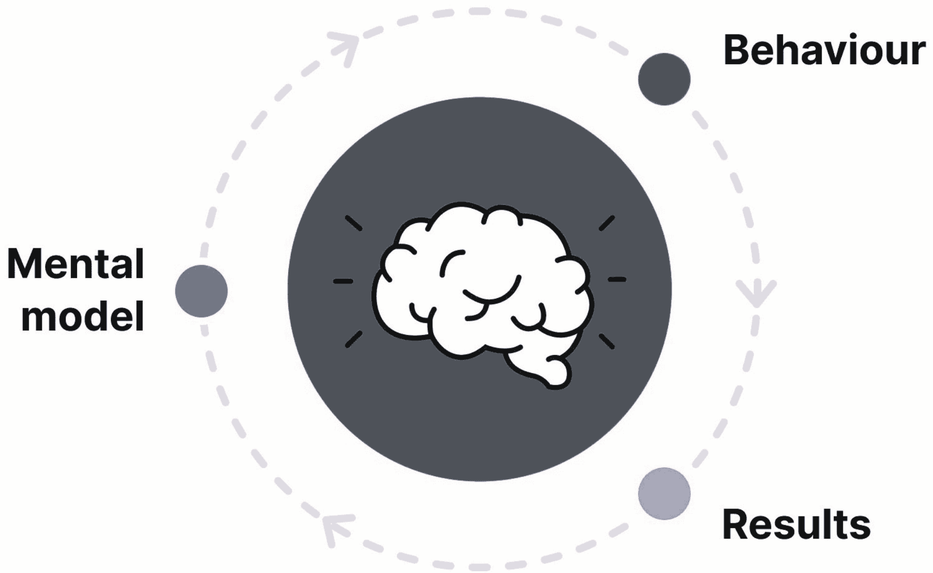
> One core issue in the LLM era is the lack of a unified framework that integrates the rich cognitive and functional components required by advanced agents. While LLMs offer exceptional language reasoning capabilities, many current agent designs remain ad hoc. They incorporate modules like perception, memory, or planning in a piecemeal fashion, failing to approximate the well-coordinated specialization seen in biological systems such as the human brain.

Liu et al., (2025)

Human Analogy to World Models

Humans naturally construct internal representations of the world, often referred to as mental models in psychology.

Humans mental models are …



… **predictive**: they anticipate what will happen next before it does.

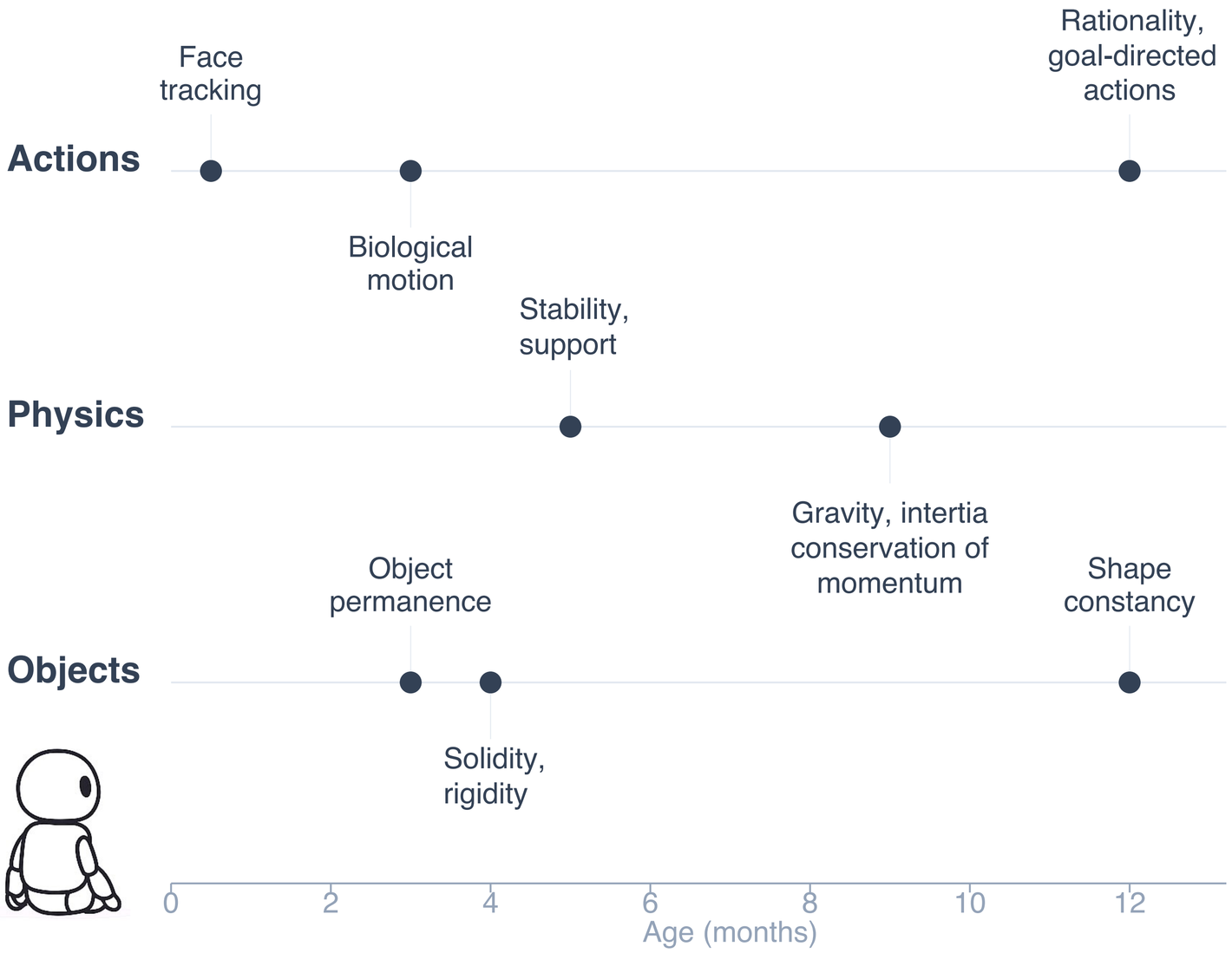
… **integrative**: fuse perception, memory, and knowledge into one coherent picture.

… **adaptive**: update when new evidence arrives or environment changes.

… **multi-scale**: span milliseconds to years, local details to global structure.

→ a human world model is not a static library of facts, but a flexible and ever-evolving mental construct, deeply rooted in perception and memory.

How do Babies Learn How the World Works?



Large Language Models

- Trained on 3×10^{13} tokens. Each token is 3 bytes.
- **Data volume**: 0.9×10^{14} bytes.
- Would take 450,000 years for a human to read.

Human Child

- 16,000 wake hours in the first 4 years.
- 2 million optical nerve fibers, carrying about 1 byte/sec each.
- Data volume: 1.1×10^{14} bytes

A four year-old child has seen more data than an LLM !

First World Models

Richard Sutton
(2019)

1991: Richard Sutton's **Dyna Algorithm**

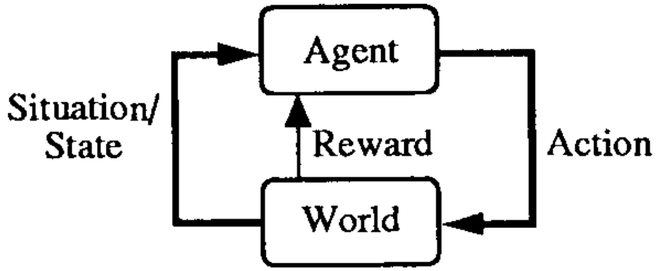


Figure 1: The Problem Formulation Used in Dyna. The agent's object is to maximize the total reward it receives over time.[1]

Sutton (1991)

The Dyna architecture attempts to integrate

- **Trial-and-error learning of an optimal *reactive policy*, a mapping from situations to actions;**
- **Learning of domain knowledge in the form of an *action model*, a black box that takes as input a situation and action and outputs a prediction of the immediate next situation;**
- **Planning: finding the optimal reactive policy given domain knowledge (the action model);**
- **Reactive execution: No planning intervenes between perceiving a situation and responding to it.**

First World Models

2018: “**World Models**” by David Ha and Jürgen Schmidhuber

> Building generative […] models of […] reinforcement learning environments. Our *world model* can be trained quickly in an unsupervised manner to learn a compressed spatial and temporal representation of the environment. By using features extracted from the world model as inputs to an agent, we can train a very compact and simple policy that can solve the required task.

Ha & Schmidhuber (2018)

First World Models

“World Models”
David Ha and
Jürgen Schmidhuber

At each time step, our agent receives an **observation** from the environment.

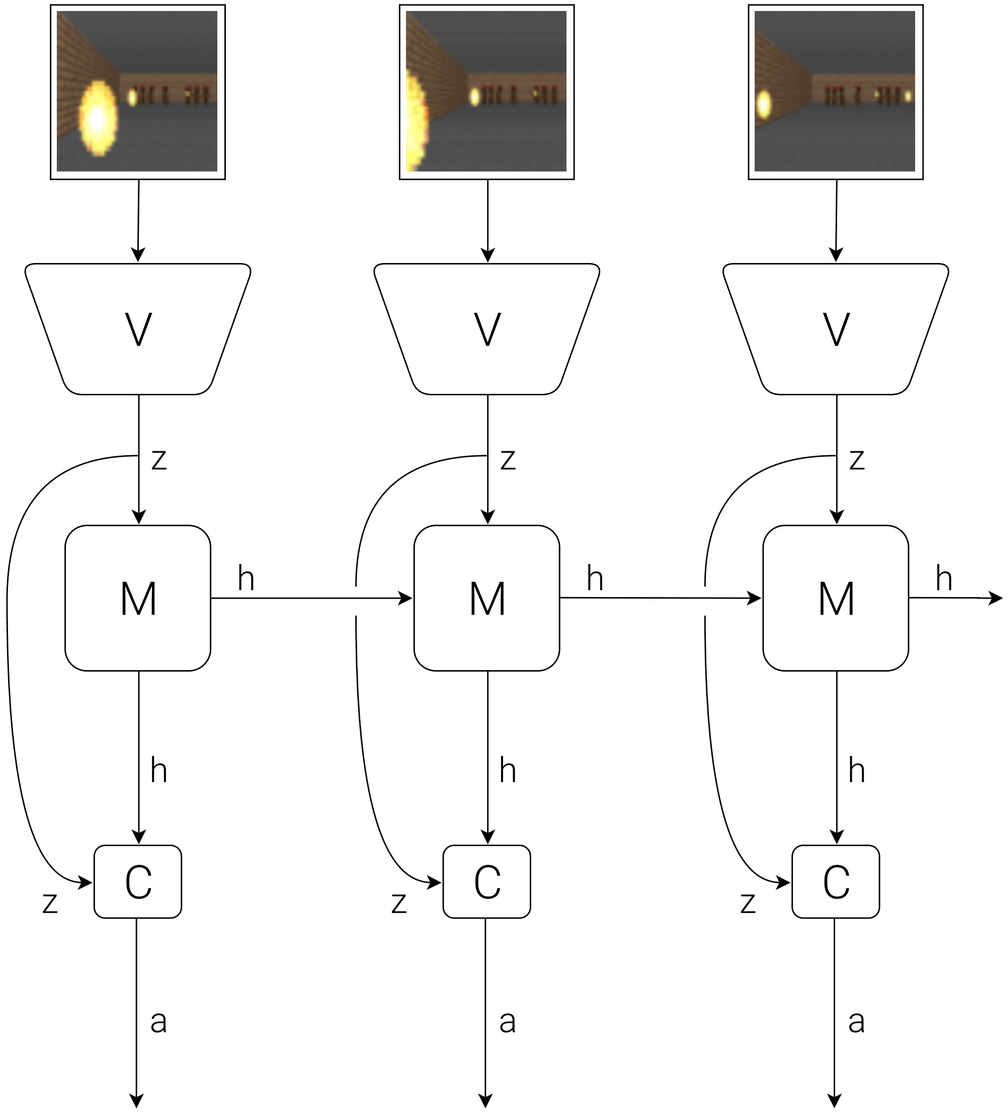
World Model

The **Vision Model (V)** encodes the high-dimensional observation into a low-dimensional latent vector.

The **Memory RNN (M)** integrates the historical codes to create a representation that can predict future states.

A small **Controller (C)** uses the representations from both **V** and **M** to select good actions.

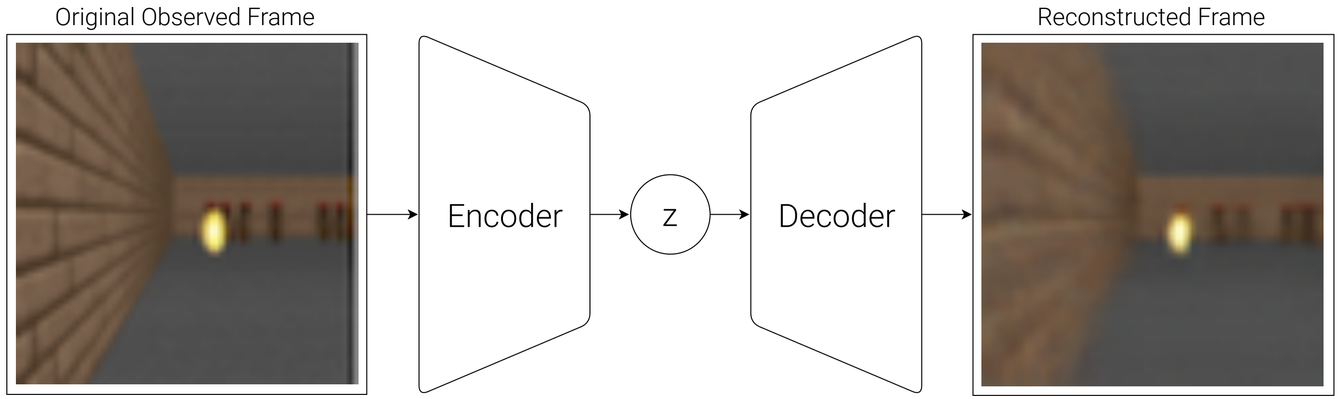
The agent performs **actions** that go back and affect the environment.



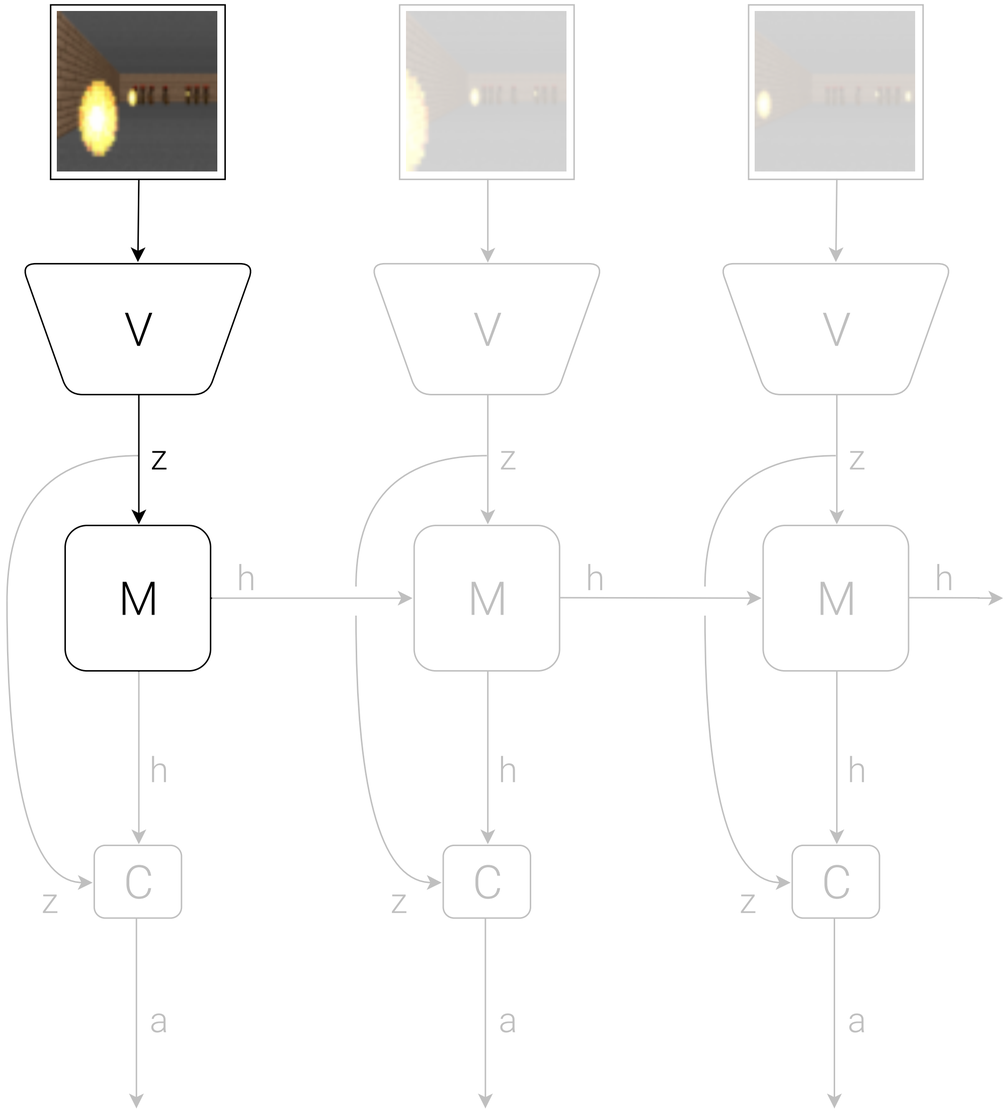
Ha & Schmidhuber (2018)

First World Models

"**World Models**" by David Ha and Jürgen Schmidhuber



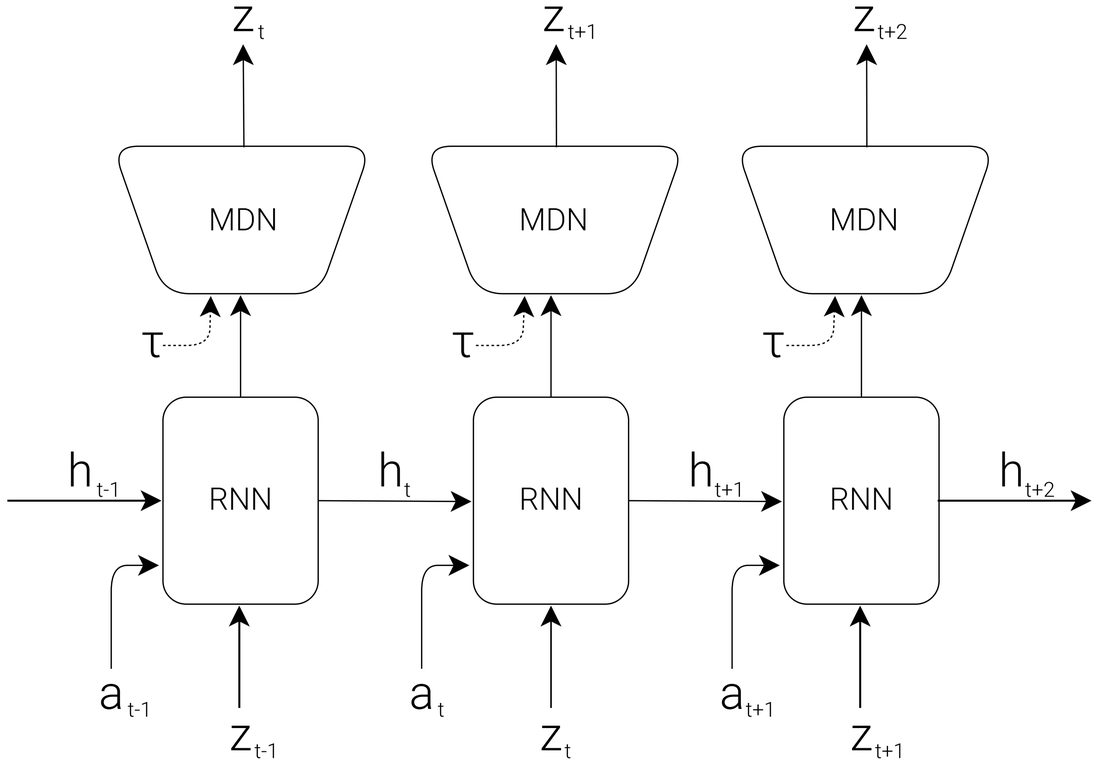
V Model: Variational Autoencoder

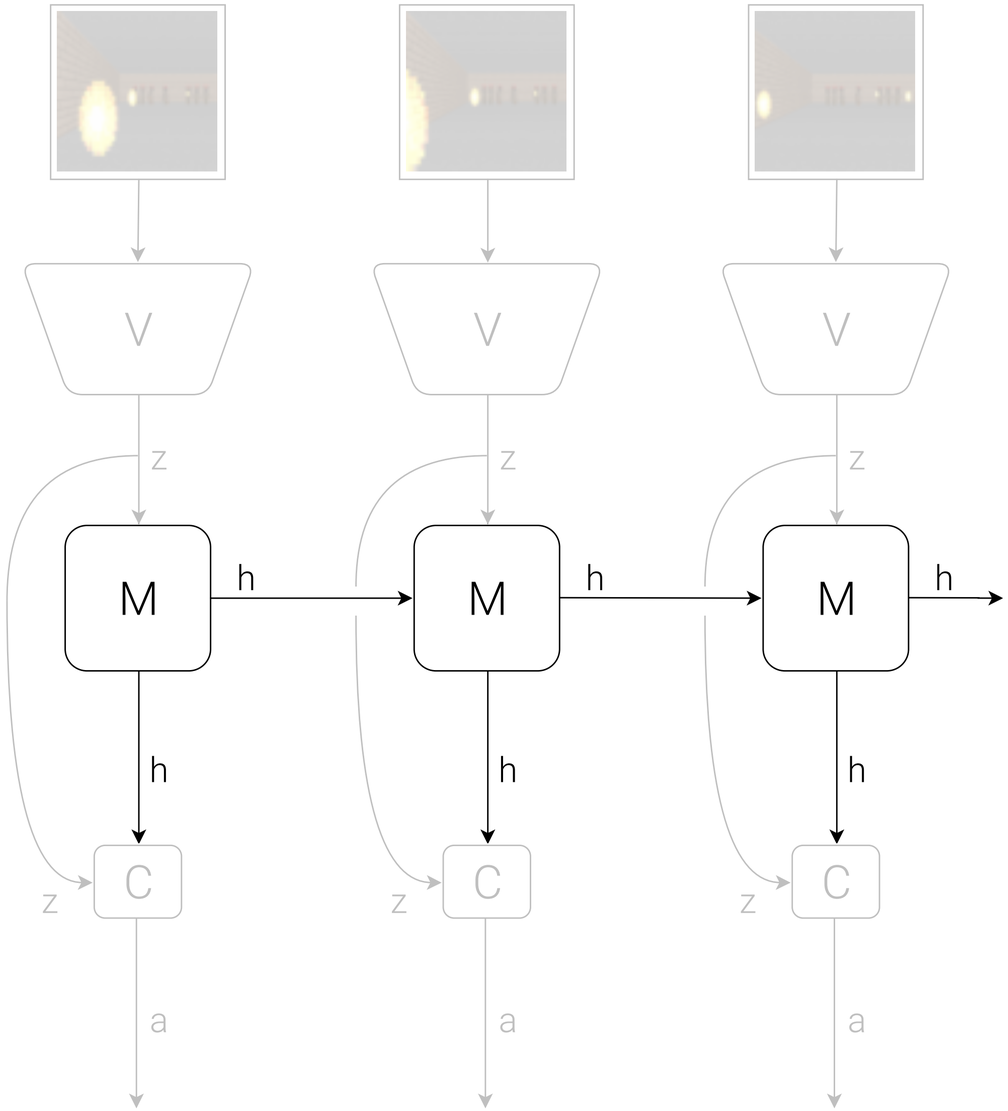


Ha & Schmidhuber (2018)

First World Models

"**World Models**" by David Ha and Jürgen Schmidhuber





M Model: RNN with a Mixture Density Network output layer

Ha & Schmidhuber (2018)

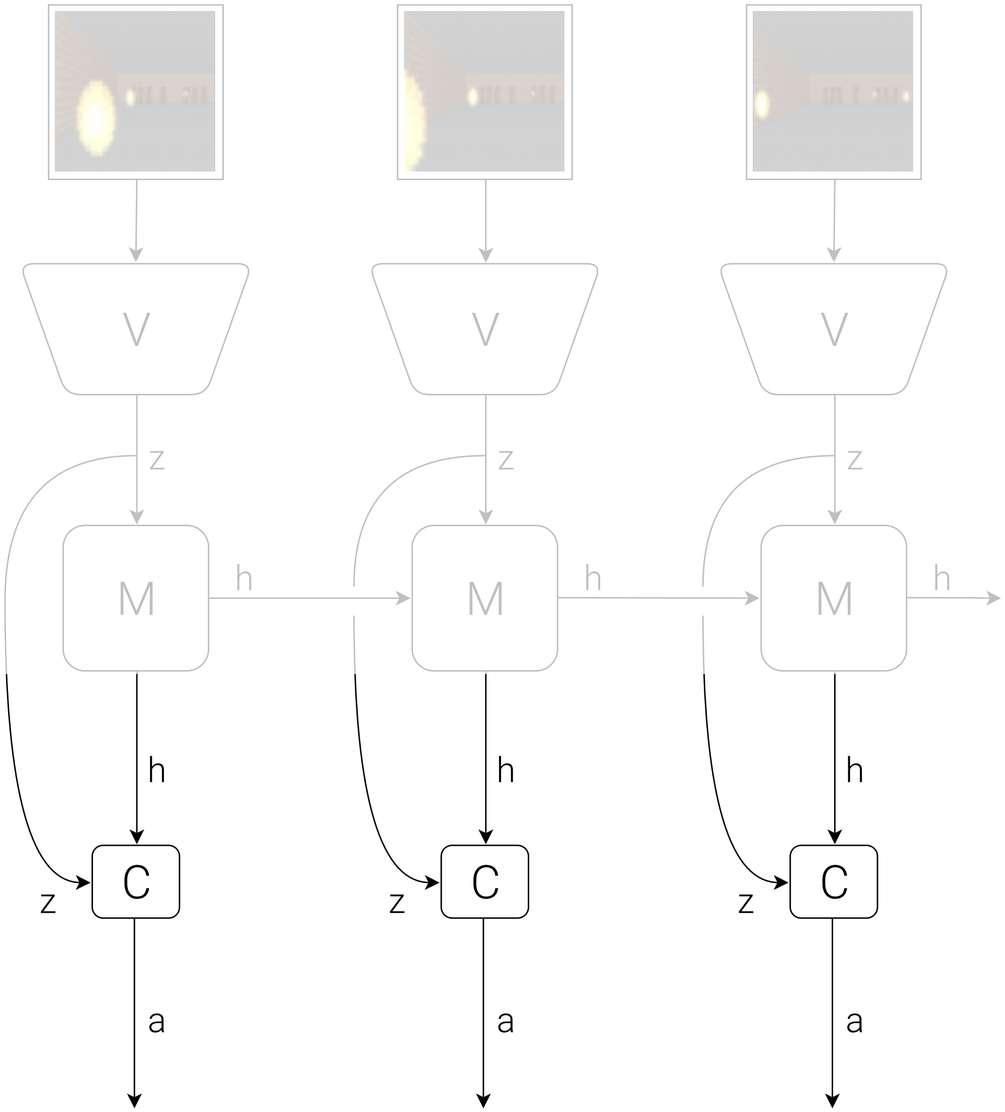
First World Models

"**World Models**" by David Ha and Jürgen Schmidhuber

C is a simple single layer linear model that maps z_t and h_t directly to action a_t at each time step:

$$a_t = W_c \, [z_t \; h_t] \; + b_c \tag{1}$$

In this linear model, W_c and b_c are the weight matrix and bias vector that maps the concatenated input vector $[z_t \; h_t]$ to the output action vector a_t.

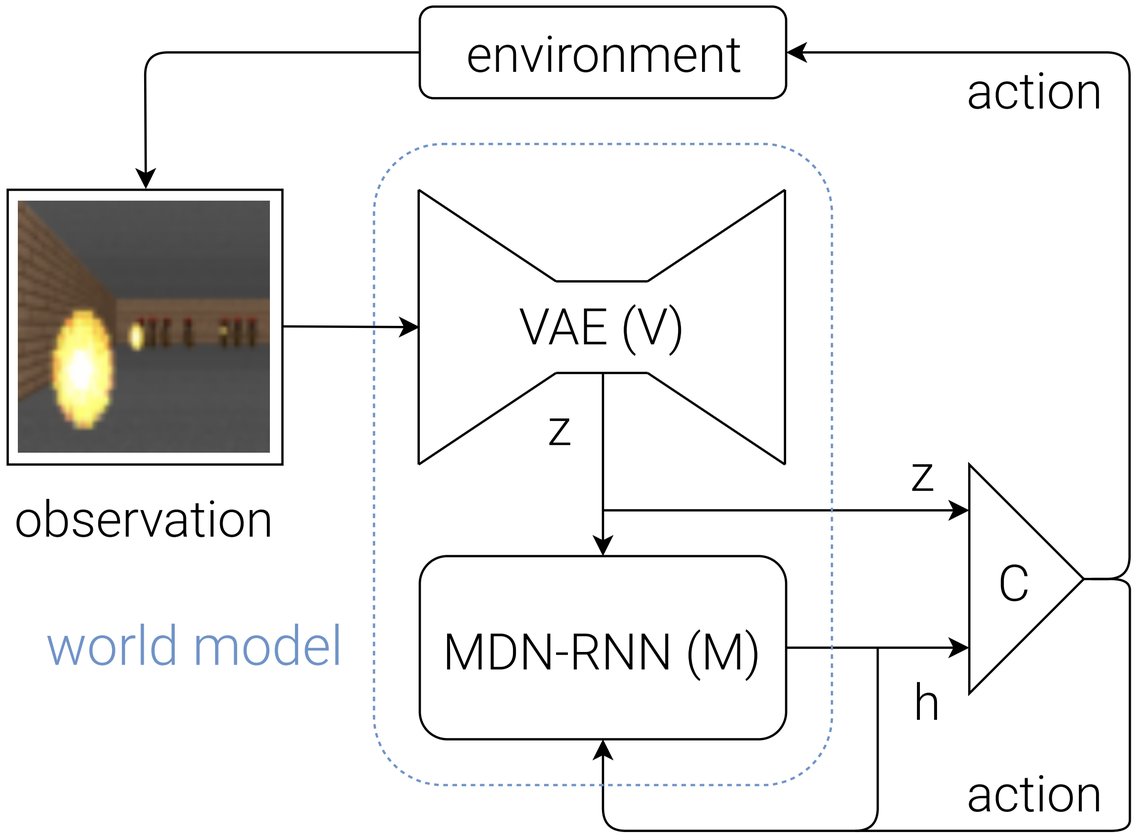


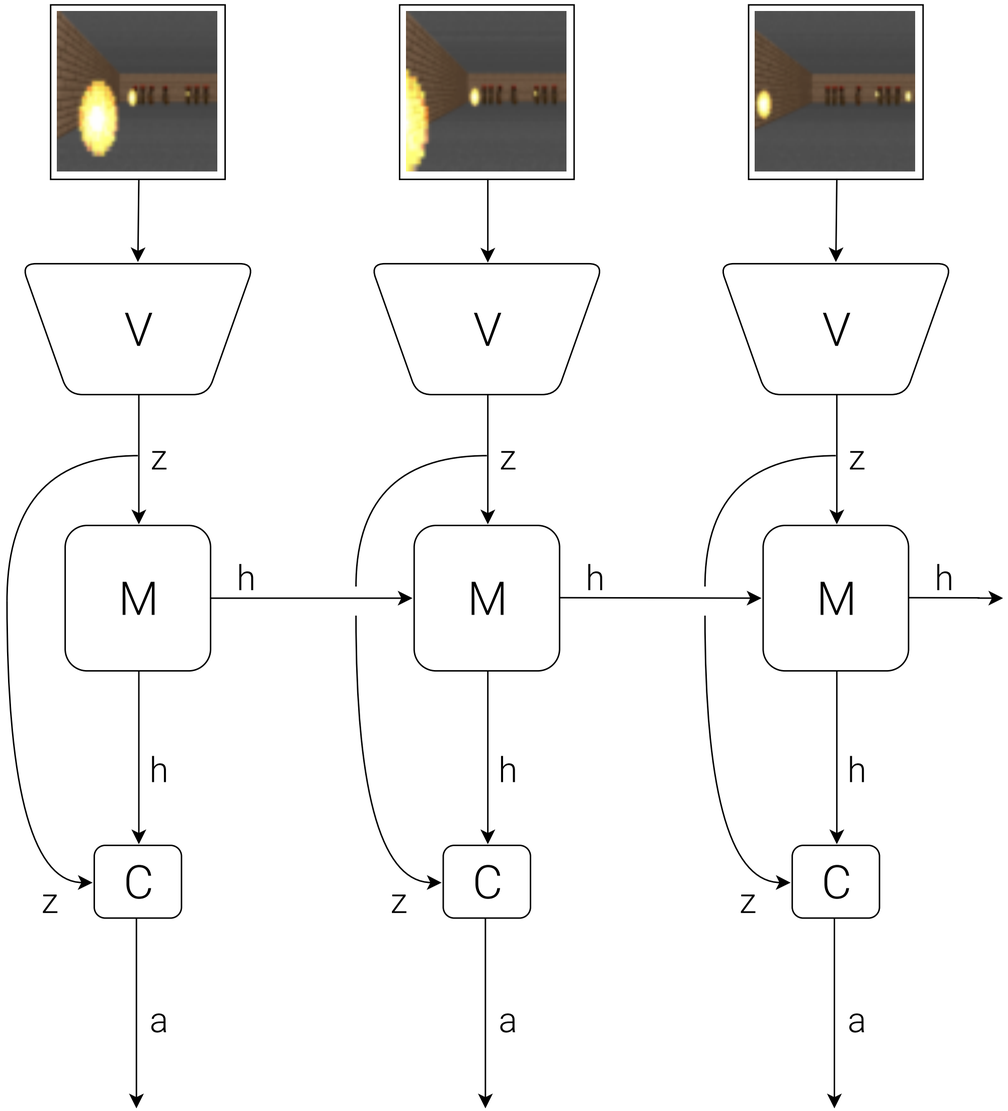
C Model: Simple single layer linear model

Ha & Schmidhuber (2018)

First World Models

"**World Models**" by David Ha and Jürgen Schmidhuber



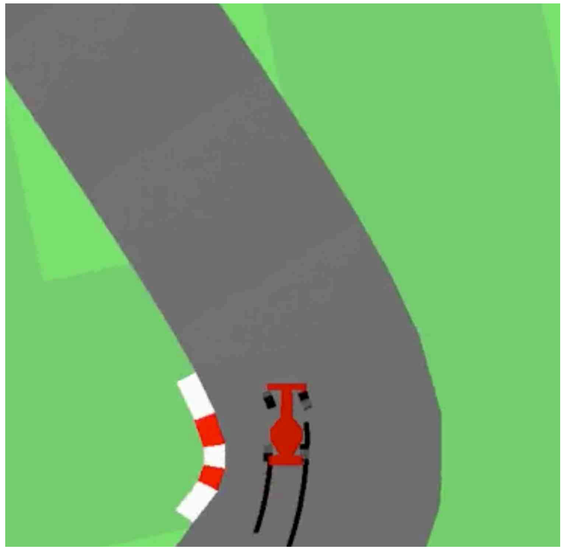


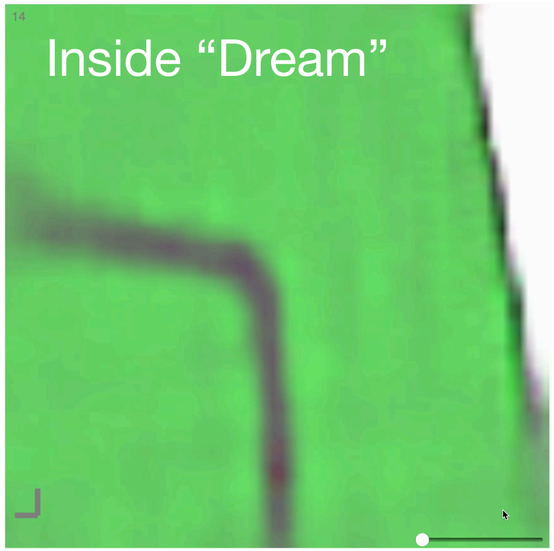
Ha & Schmidhuber (2018)

First World Models

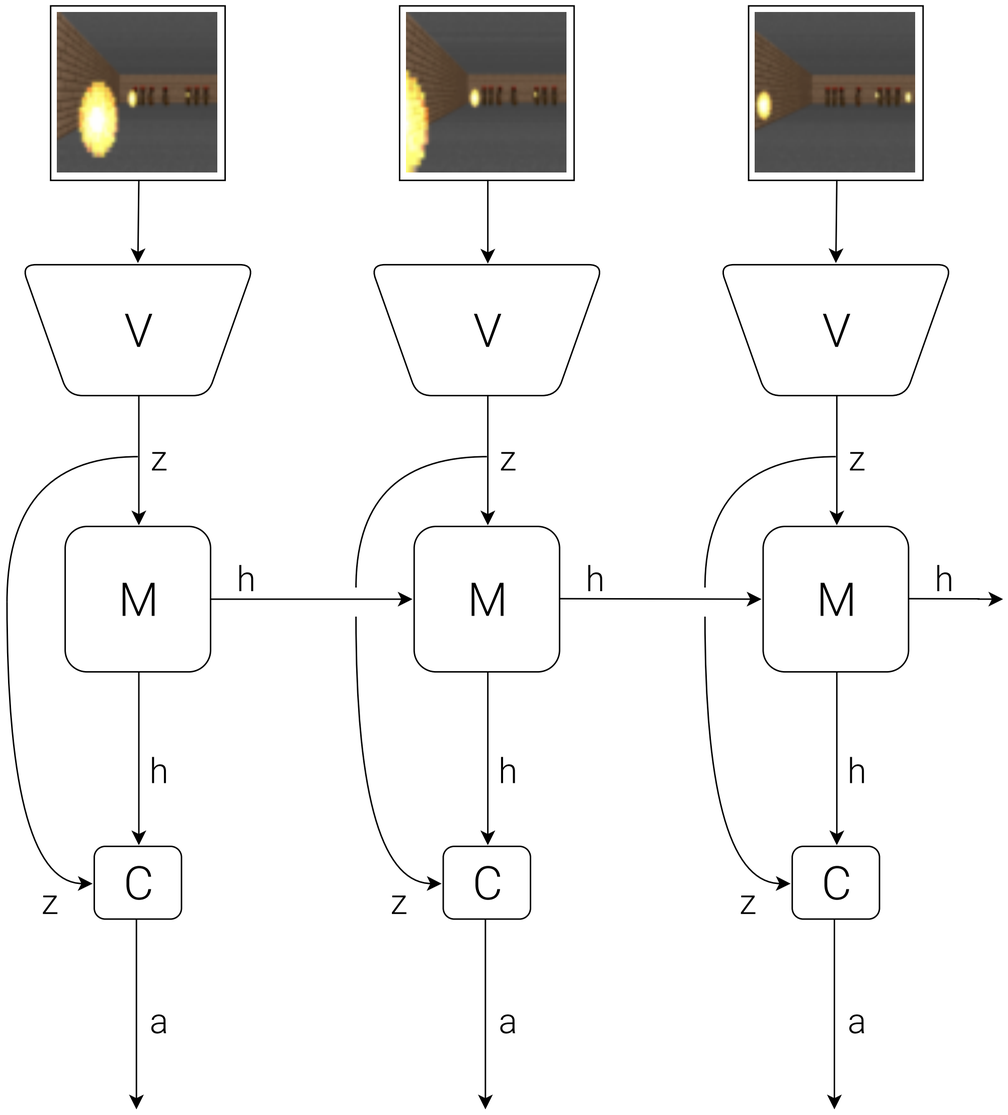
"**World Models**" by David Ha and Jürgen Schmidhuber

→ Policy (controller) could be trained entirely within the learned model's "dream" and then successfully transferred to the real game environment.





→ use generated $\hat{z}_{t+1}$ as real observation at $t+1$



Ha & Schmidhuber (2018)

Where Does *Generation* Enter World Models?

From Static Representations → Dynamic Simulators:

So far we looked at foundation models that either to learn good representations, model static states. Generation was mostly a self-supervised learning principle.

In world models, **generation means simulating how the world evolves over time**.

- **Generate the next step:**
  From what the agent currently sees and does,
  the world model generates the next observation and reward.
- **Generate whole futures:**
  By rolling forward, it generates entire trajectories so the agent can “imagine” life ahead.
- **Generate counterfactuals and plans:**
  It generates alternative futures for different actions, letting the agent compare options, choose plans, and even create new training worlds.

Ways to Build World Models

> A world model enables an agent to predict and reason about future states without direct trial-and-error in reality.

Designing an AI world model involves determining how an AI agent acquires, represents, and updates its understanding of the environment's dynamics.

While implementations vary, most approaches fall into **four broad paradigms**:

implicit | explicit | simulator-based | instruction-driven models

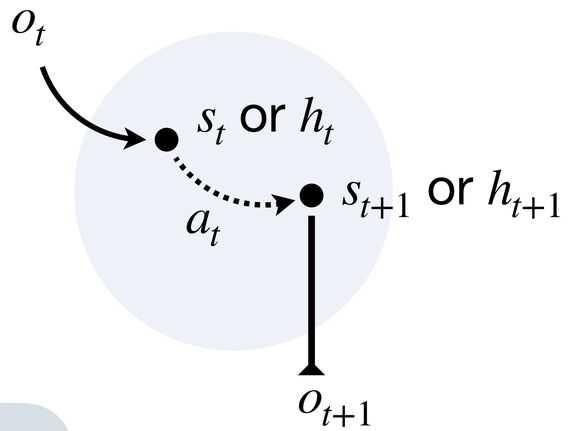
Liu et al., (2025)

Preliminaries and Notation

Let $\mathcal{S}$ denote the **set of possible environment states**,

$\mathcal{A}$ the **set of actions**,

and $\mathcal{O}$ the **set of observations**



Notation

- state at time t: s_t
- action at time t: a_t
- observation at time t: o_t
- latent state at time t: z_t or h_t

implicit

e.g.,

MuZero, Dreamer, V-JEPA 2

explicit

e.g.,

DINOv2 World Model, Diffusion World Models,

simulator-based

e.g.,

SAPIEN

instruction-driven

e.g.,

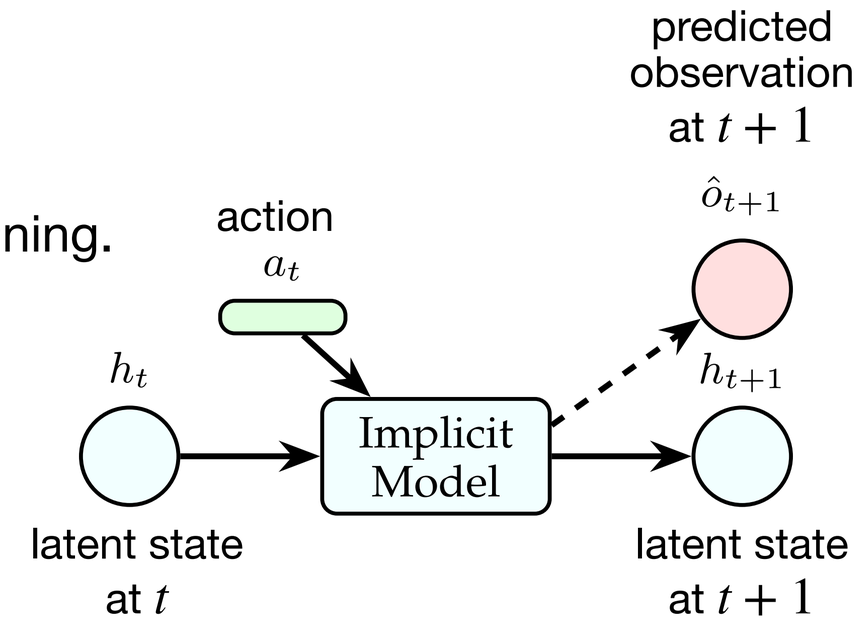
Genie 1, 2, 3

Implicit World Models

Key idea:

- Simulate the future entirely in a **latent state space**.
- Agent does not reconstruct future observations during planning.

- Decoder, if present, is only for
  - auxiliary training losses or
  - visualization, not for decision-making.



An implicit world model retains a state $h_t \in \mathscr{H}$, updated as

$$h_{t+1} = f_\theta\left(h_t, a_t\right), \quad \hat{o}_{t+1} = g_\theta\left(h_{t+1}\right) \text{ (optional, not used in rollout)}$$

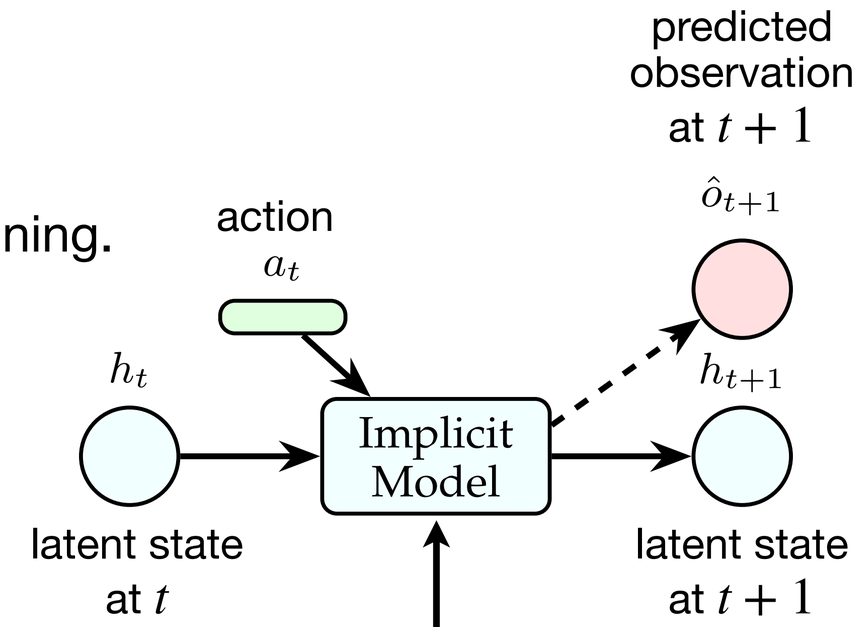
Decoder

Implicit World Models

Key idea:

- Simulate the future entirely in a **latent state space**.
- Agent does not reconstruct future observations during planning.

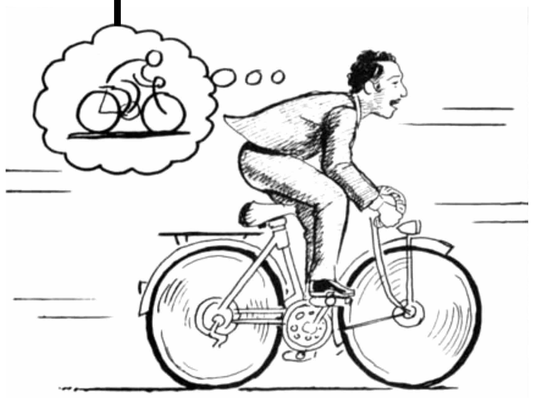
- Decoder, if present, is only for
  - auxiliary training losses or
  - visualization, not for decision-making.



Analogy:

Thinking in abstract symbols about the game state,

without ever “rendering” a picture of the board.

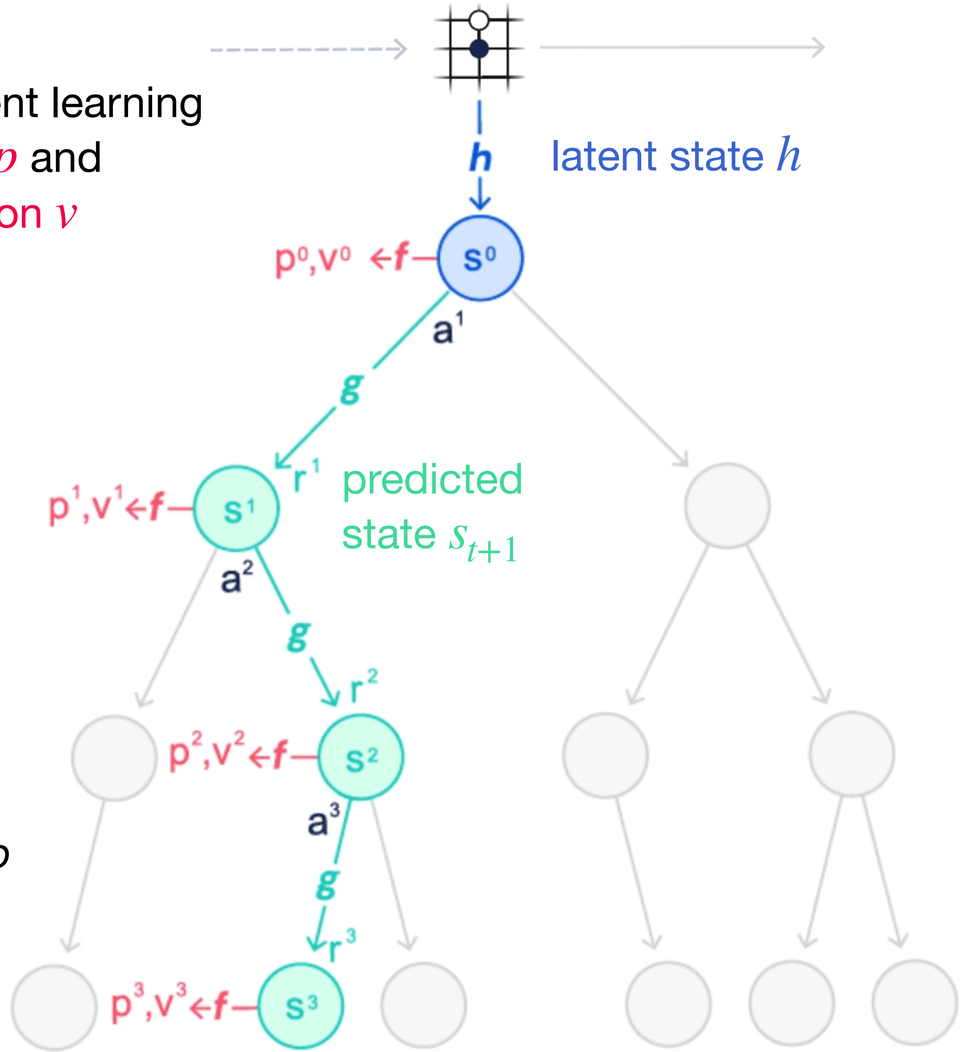


Examples: World Models, MuZero, Dreamer, V-JEPA2, etc.

Implicit World Models: **MuZero**

Schrittwieser et al., (2020)

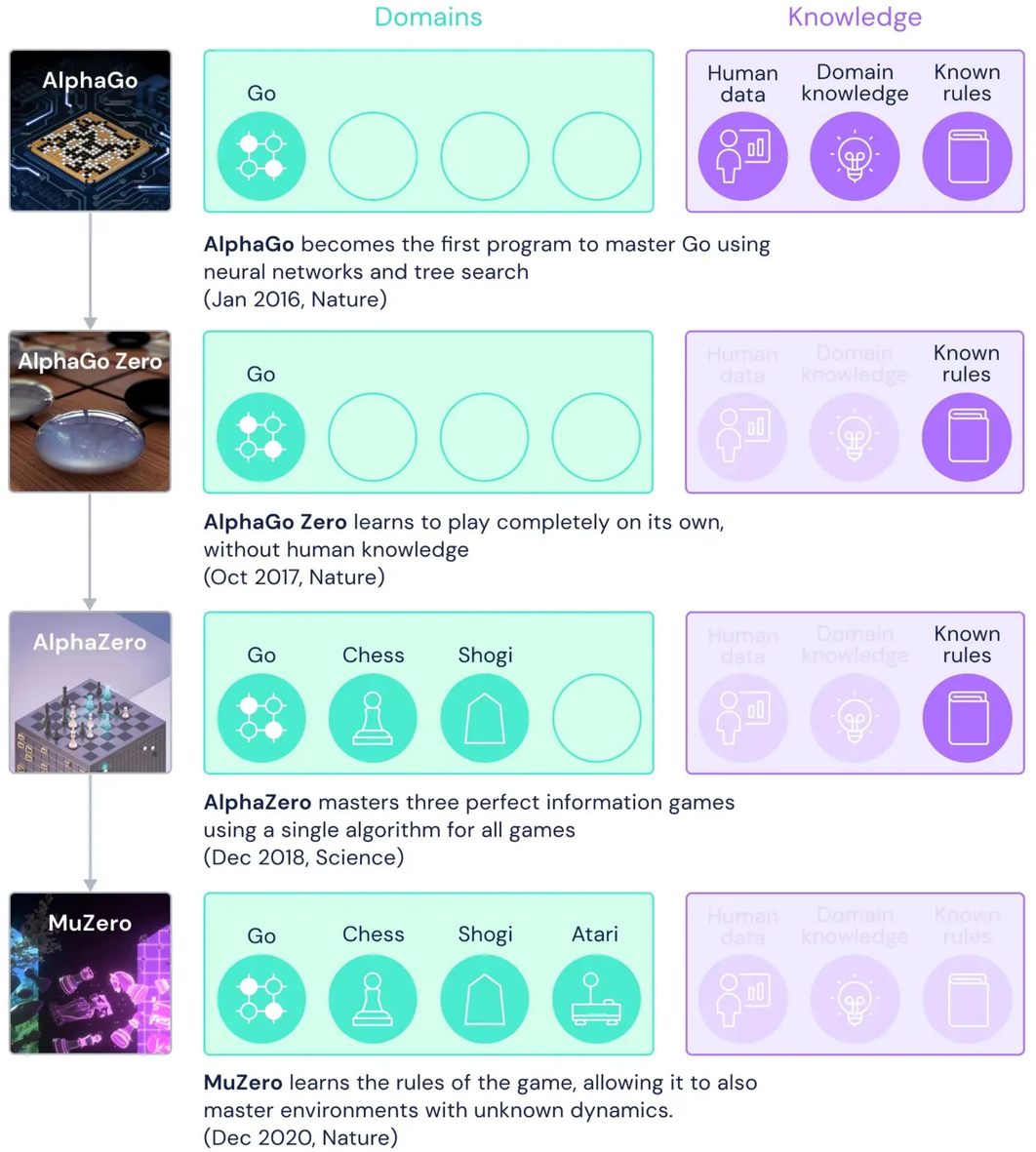
Based on reinforcement learning with policy p and value function v



prediction function f

dynamics function g

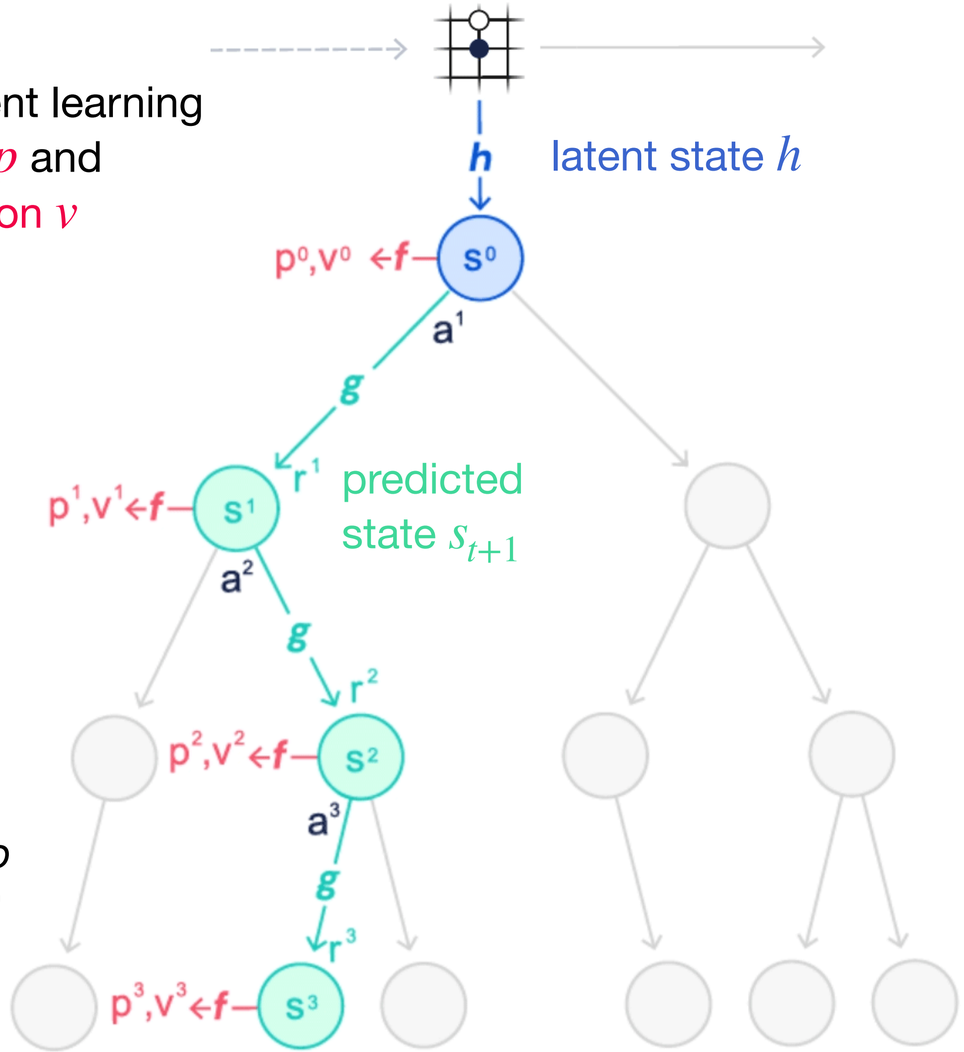
Monte Carlo Tree Search



Implicit World Models: **MuZero**

Schrittwieser et al., (2020)

Based on reinforcement learning with policy p and value function v



prediction function f

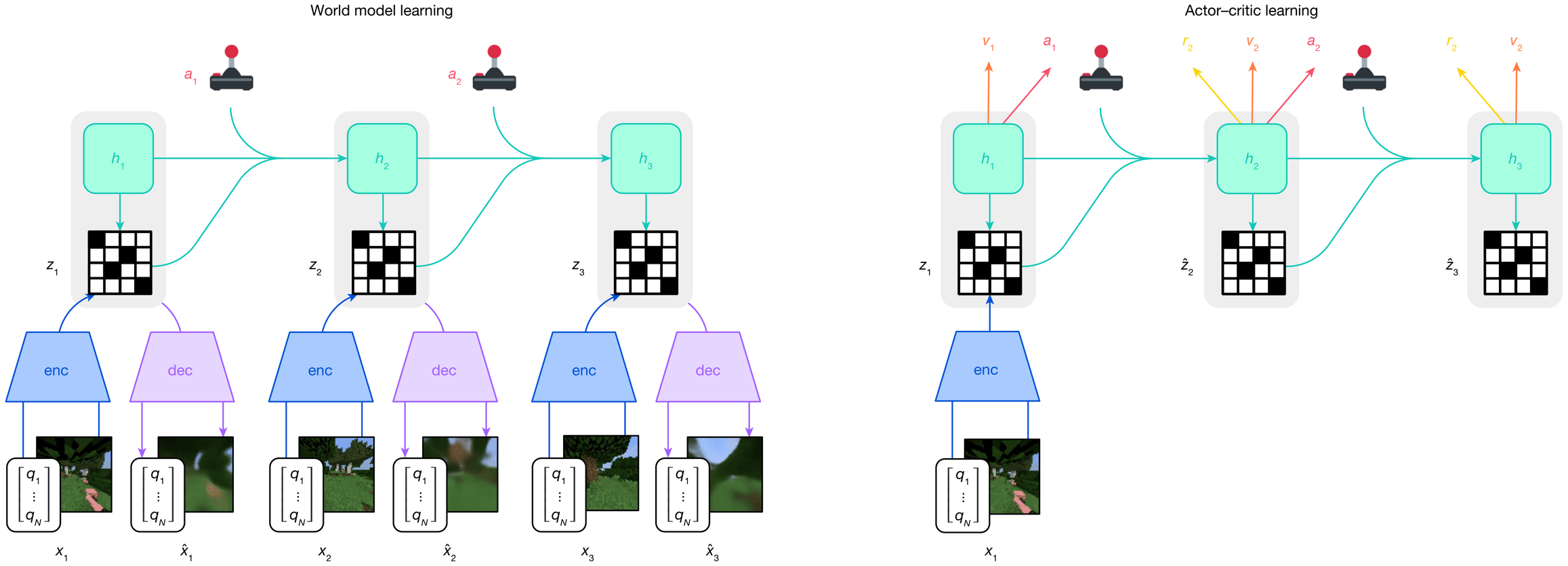
dynamics function g

Monte Carlo Tree Search

- At decision time, MuZero runs Monte Carlo Tree Search over latent states s_k.
- Each node expansion uses f and g to get new states, rewards, values, and policies.
- The final action is chosen from root visit counts.

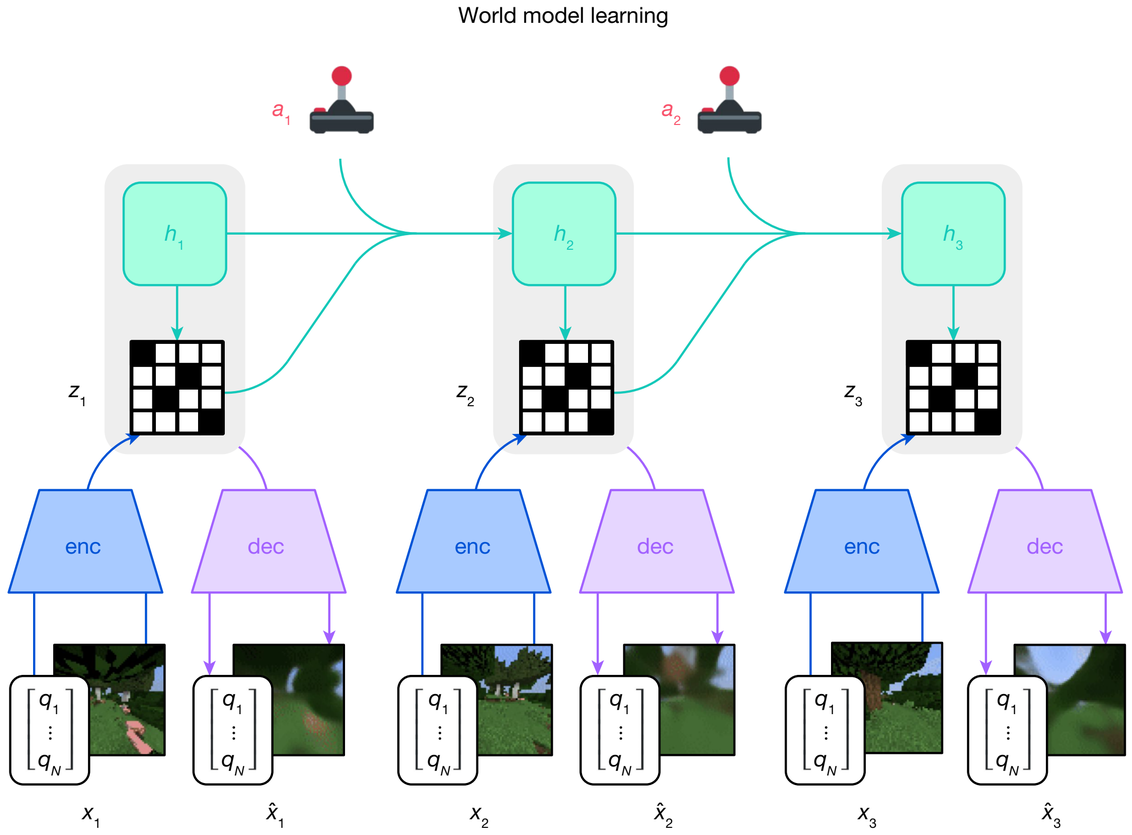
Implicit World Models: **DreamerV3**

Hafner et al., (2025)



Implicit World Models: **DreamerV3**

Hafner et al., (2025)

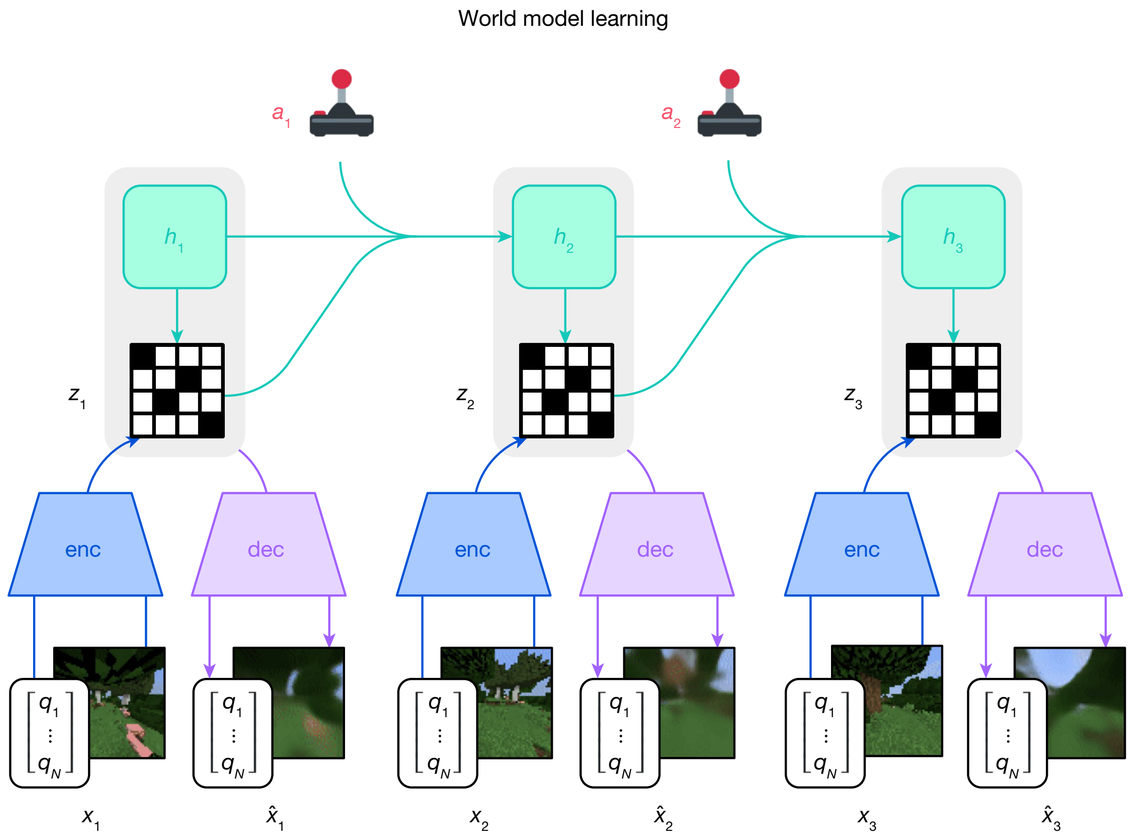


- Dreamer world model learns compact representations of sensory inputs through **autoencoding**.
- World model is implemented as as a **recurrent state-space model**.

1. An encoder maps sensory inputs x_t to stochastic representations z_t for each time step t in the training sequence.
2. A sequence model with recurrent state h_t predicts the sequence of these representations given past actions a_{t-1}.
   The concatenation of h_t and z_t forms the model state from which one predicts rewards r_t.

Implicit World Models: **DreamerV3**

Hafner et al., (2025)



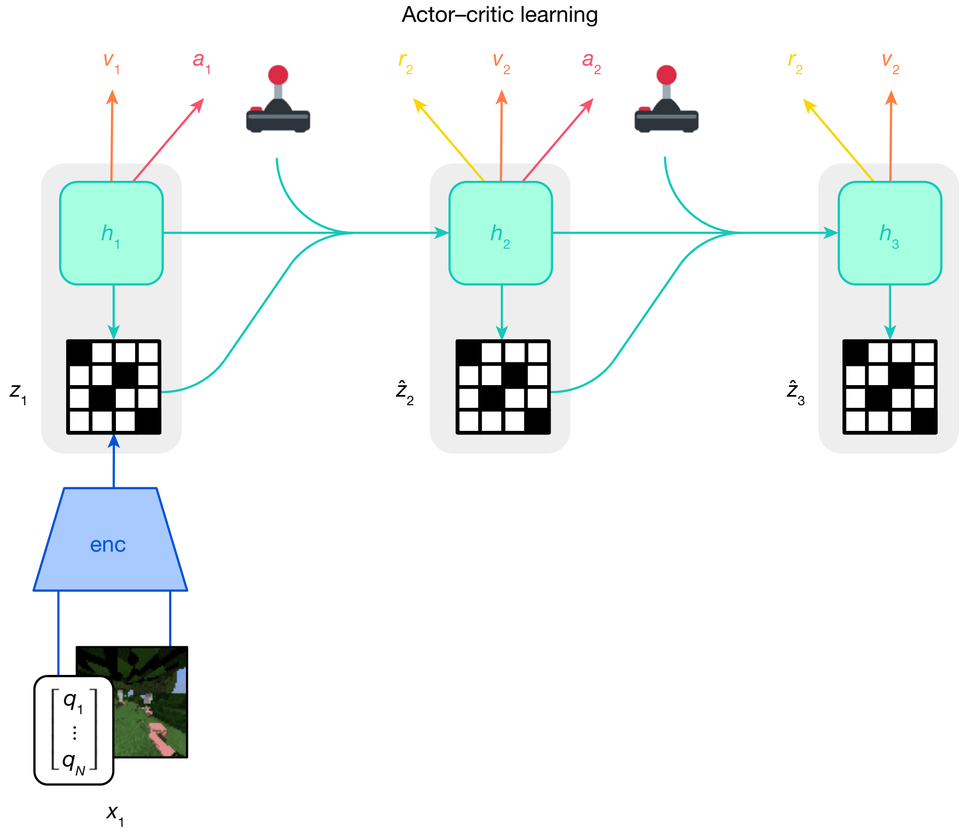
Components of DreamerV3

1. **World Model**
   Recurrent state-space model (no Transformers) that encodes observations into latents and, given an action, predicts next state, reward, and whether the episode continues.
2. **Critic**
   Distributional value network that evaluates imagined trajectories from the world model, with strong normalization and EMA-averaged parameters for stability under sparse or noisy rewards.
3. **Actor**
   Policy network that chooses actions from imagined futures, maximizing predicted return while maintaining exploration so it does not get stuck in local optima.

Implicit World Models: **DreamerV3**

Hafner et al., (2025)

- **Training Loop**:
  - Collect real experience from the environment.
  - Train RSSM to model latent dynamics + reconstructions.
  - Imagine trajectories in latent space.
  - Optimize actor and critic from imagined rollouts.
  - Same hyperparameters across 150+ tasks (Atari, DM Control, Minecraft…).
- **Difference to MuZero**:
  - Dreamer: no planning tree, instead does offline imagination to train a feedforward actor.
  - Much more suitable for continuous actions and high-dimensional control.

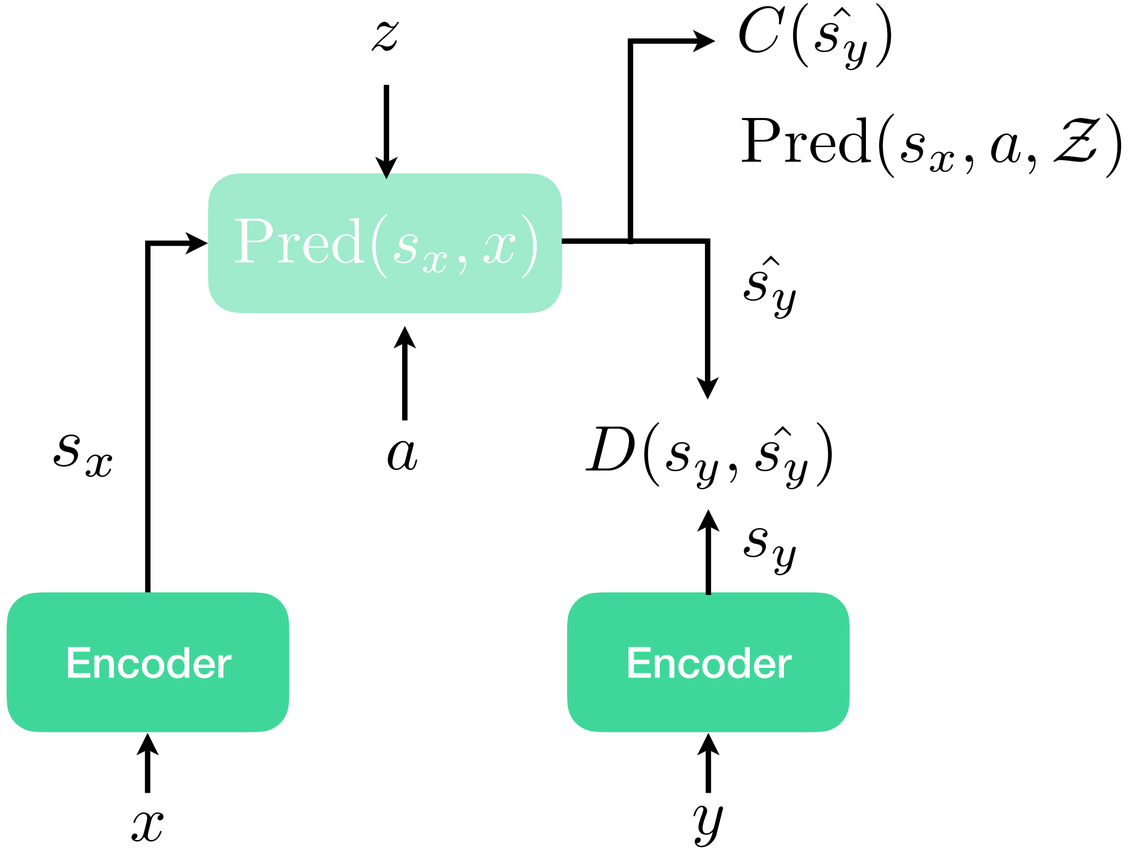


Excursus: **Joint-Embedding Predictive Architecture = JEPA**

A counterproposal to Transformers! Architectures for world models.

- x : observed past and present
- y : future
- a : action
- z: latent variable (*unknown*)
- $D(\cdot)$: prediction cost
- $C(\cdot)$: surrogate cost

→ JEPA predicts a representation of the future s_y from a representation of the past s_x.

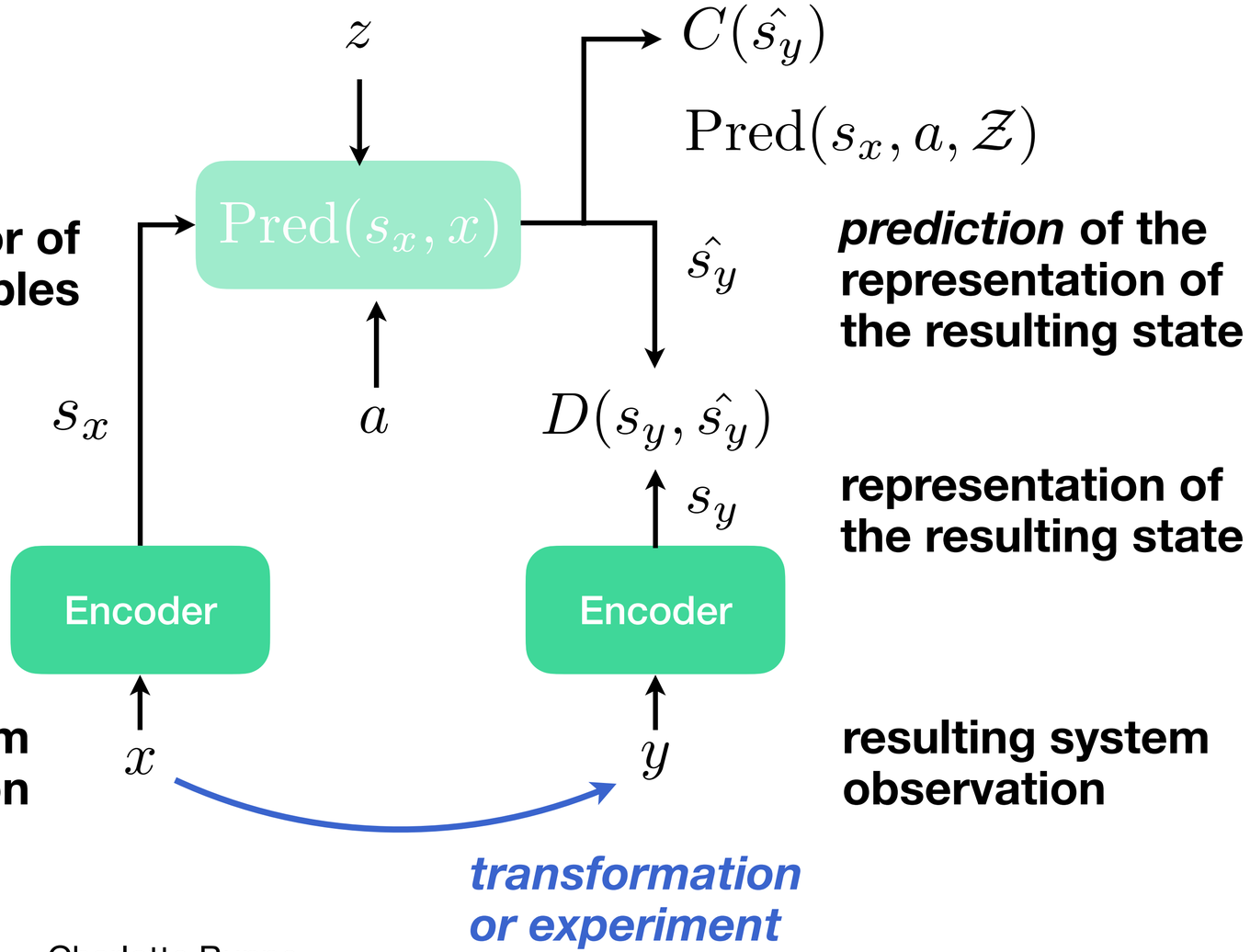


Excursus: **Joint-Embedding Predictive Architecture = JEPA**

Goal:

- Find an abstract state representation that allows to make predictions.
- Extract the state representations from observations or measurements
- Predict the outcome resulting from a perturbation or experiment y

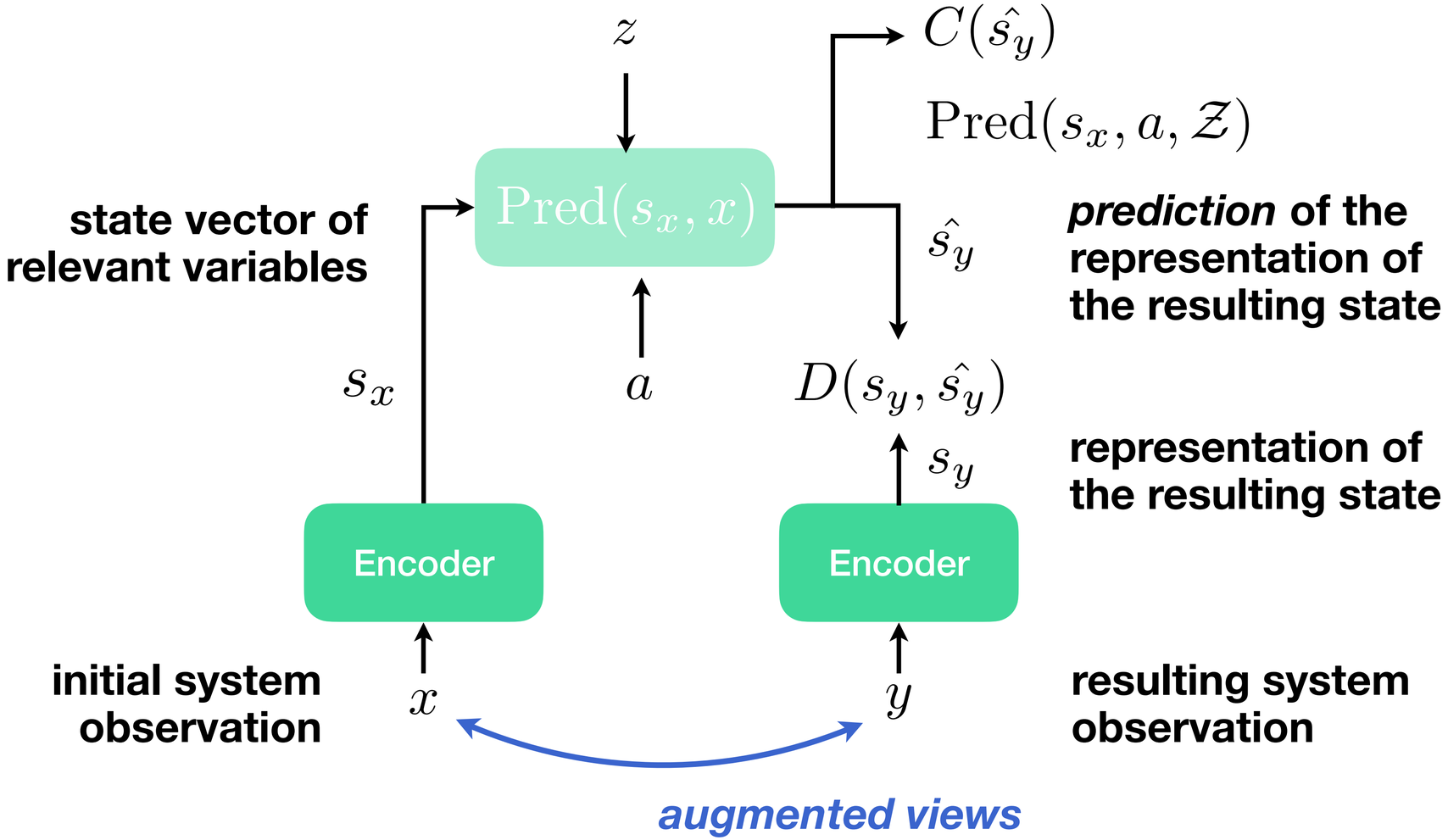
JEPA can be trained via contrastive, distillation-based objective, etc.



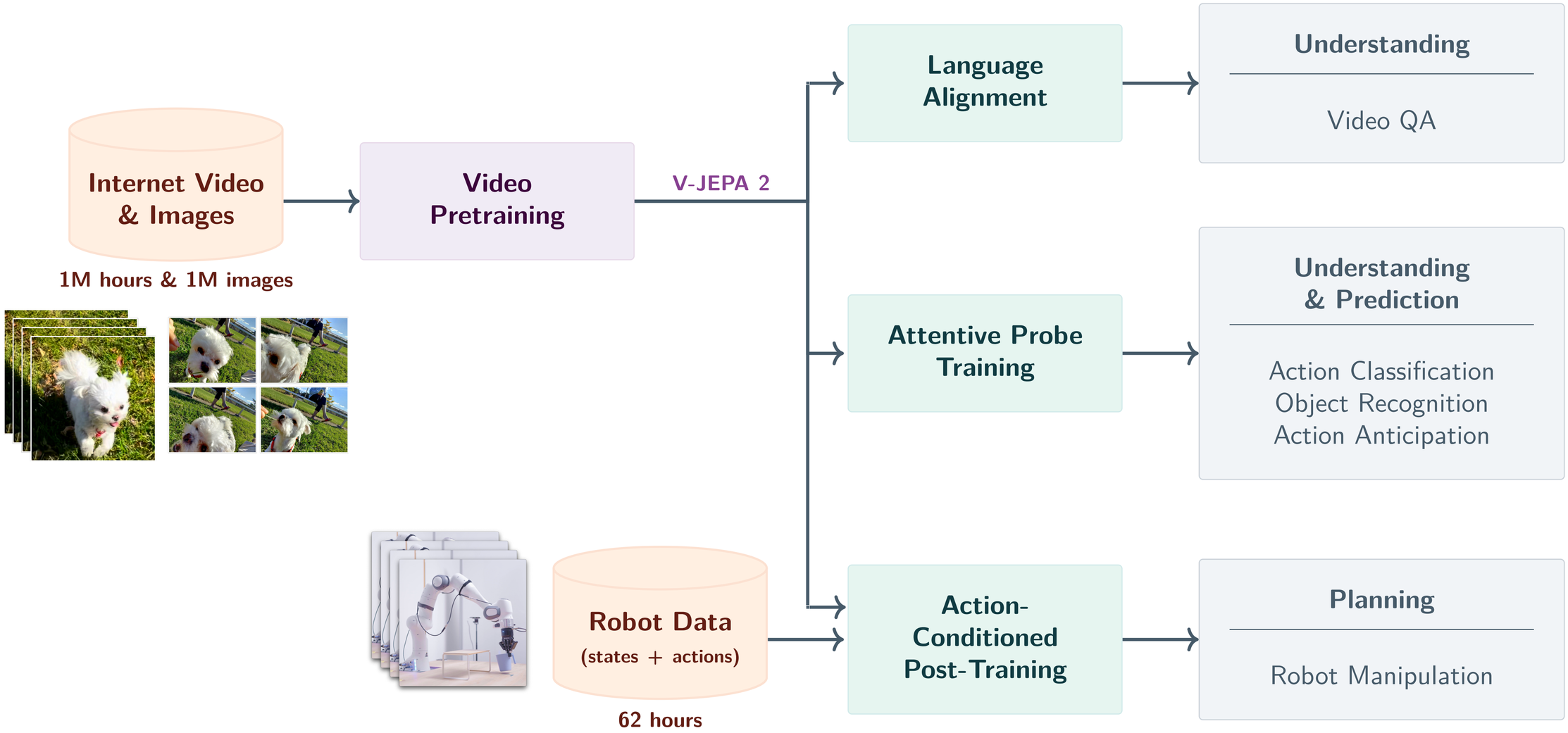
Excursus: **Joint-Embedding Predictive Architecture = JEPA**

e.g., DINOv2

- when one Encoder is the student and the other one the teacher network, both map an **augmented view** of the same data samples to embeddings.



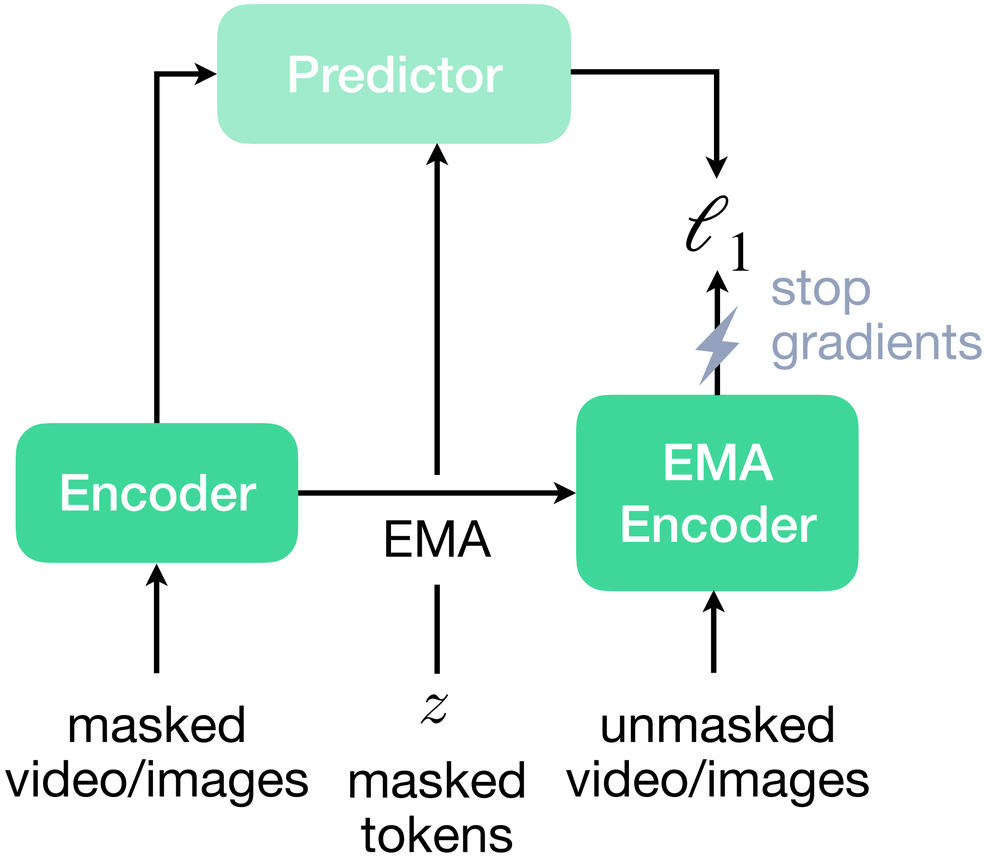
Implicit World Models: **Video World Model at Scale (V-JEPA 2)**



Implicit World Models: **Video World Model at Scale (V-JEPA 2)**

Trained on **>1M hours of internet video and images**.

- **V-JEPA 2 Architecture**:
  - Context encoder E_c processes visible frames.
  - Target encoder E_t processes masked future frames.
  - Predictor P_θ maps context embedding to predicted future embedding.
- **Loss**: predict latent features of masked video regions.
  - No pixel-level loss; prediction happens directly in representation space.

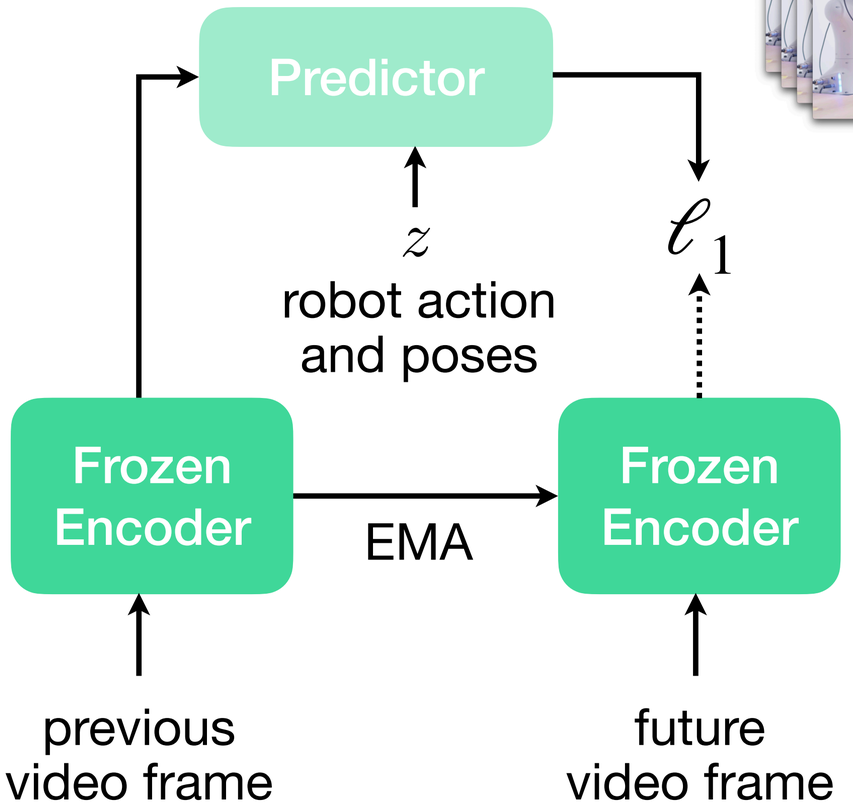


Implicit World Models: **Video World Model at Scale (V-JEPA 2)**

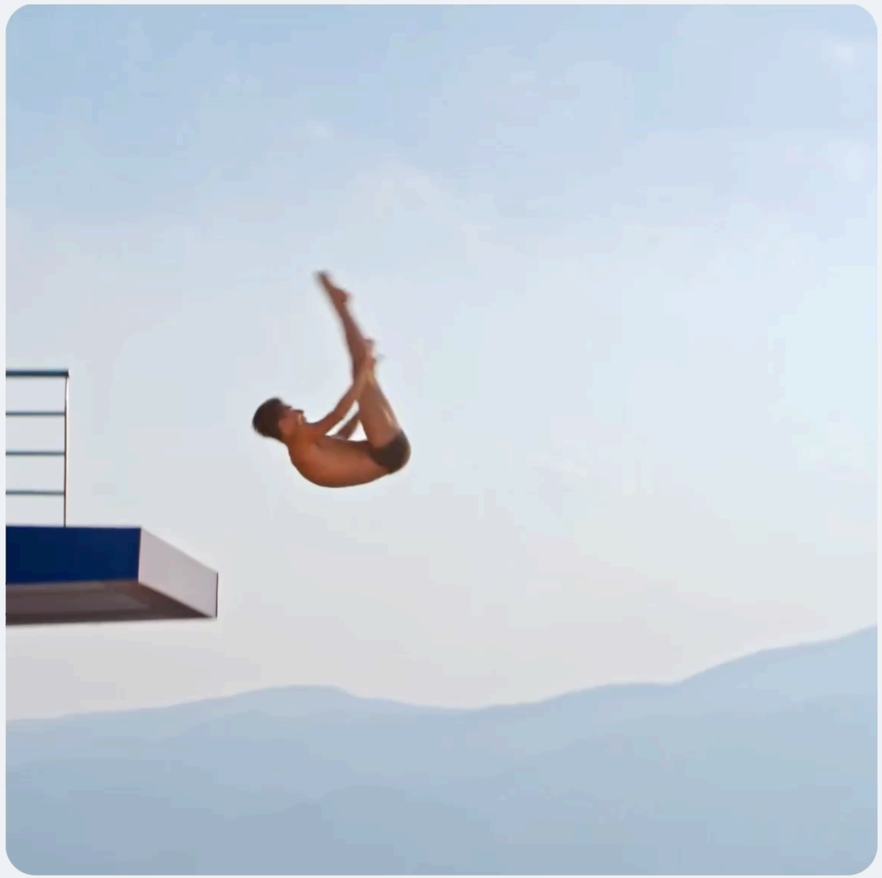
- **V-JEPA 2-AC expands architecture with an action-conditioned head**:
  - Trained on < 62h of robot videos.
  - Learn latent dynamics conditioned on robot actions.

- **Zero-shot control**:
  - Given a *goal image*, encode it into the same latent space.
  - Search for action sequences whose predicted latent future matches the goal embedding.
  - Demonstrated on Franka arms in new labs without task-specific training.

- Planning entirely in embedding space, not in pixels.



Implicit World Models: **Video World Model at Scale (V-JEPA 2)**



Unlock world understanding

V-JEPA 2 delivers exceptional motion understanding as well as leading visual reasoning capabilities when combined with language modeling.

Anticipate what's next

V-JEPA 2 can make predictions about how the world will evolve, setting a new state-of-the-art in anticipating actions from contextual cues.

Explicit World Models

Key idea:

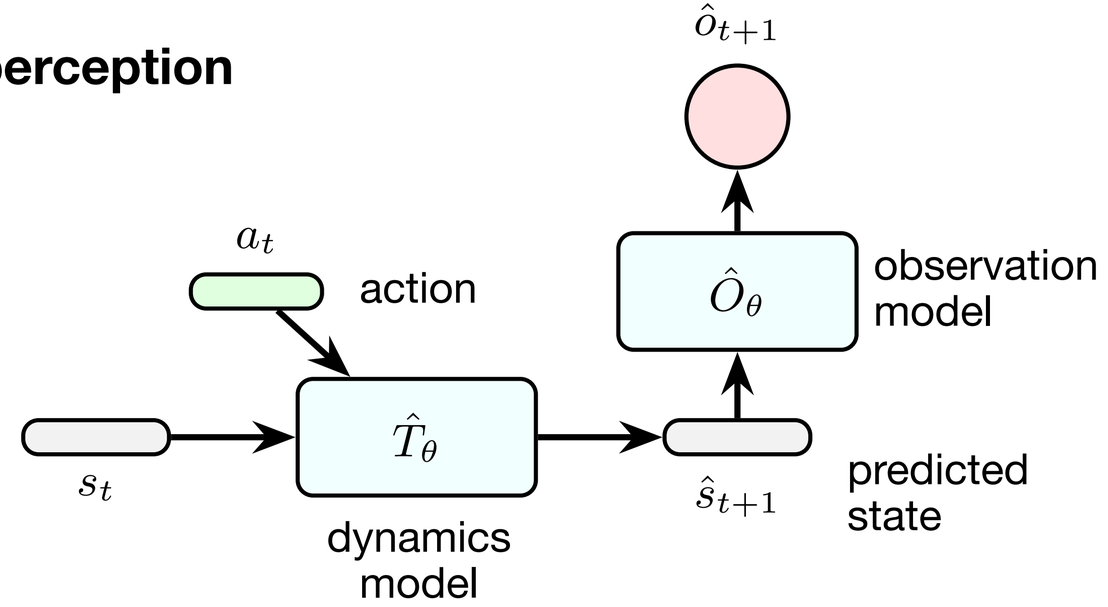
- World model **reconstructs or predicts future observations** during imagined rollouts.
- These predicted observations are used for training and/or decision-making.

Simulated trajectories are in the **same modality as perception**
→ easier to inspect and constrain.

- Factorized dynamics and observation model:

$$\hat{s}_{t+1} = T_\omega\left(s_t, a_t\right)$$

$$\hat{o}_{t+1} = O_\omega\left(\hat{s}_{t+1}\right)$$



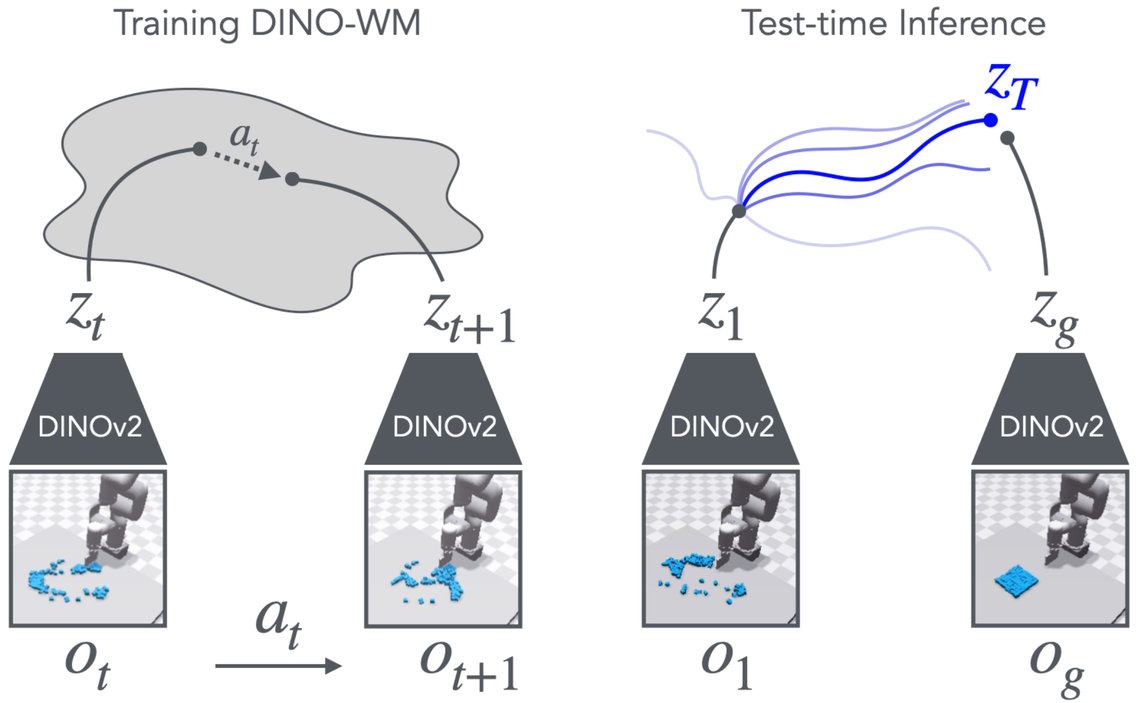
Use $\hat{o}_{t+1}$ (and subsequent frames) to evaluate actions or compute values.

Explicit World Models

Difference to Implicit World Models:

- **Capability to reconstruct future observations during imagined rollouts** distinguishes explicit models from implicit ones.
- While both may use latent representations internally, explicit models treat accurate observation prediction as central to the simulation process.

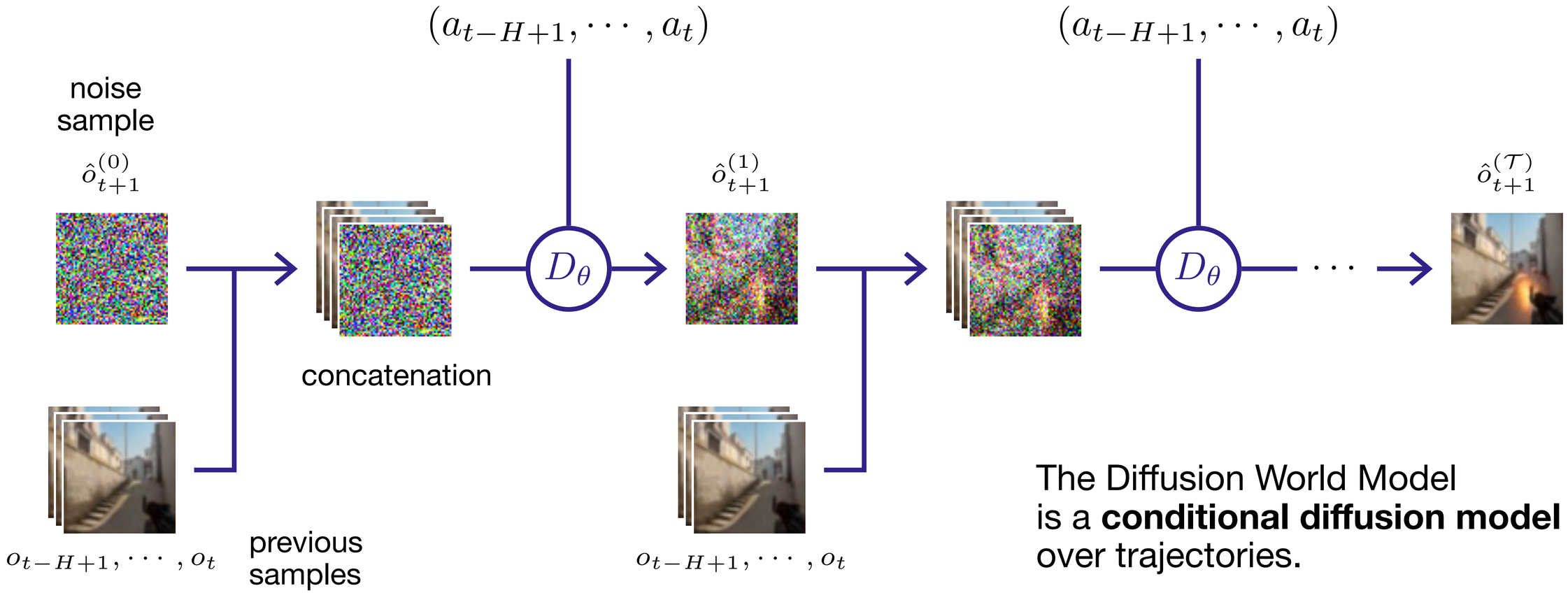
e.g., **DINOv2 World Model**



Zhou et al., (2024)

Explicit World Models: **Diffusion World Models**

Diffusion world models use a diffusion process to generate plausible future trajectories from the current state and actions, providing a **generative world model of the environment's states and dynamics**.



The Diffusion World Model is a **conditional diffusion model** over trajectories.

Explicit World Models: **Diffusion World Models**

Figure 2: Example trajectories sampled from diffusion world models in the environments tested in this paper; 2D Atari games, a modern 3D first-person shooter, and real-world motorway driving.

Simulator-Based World Models

Key idea:

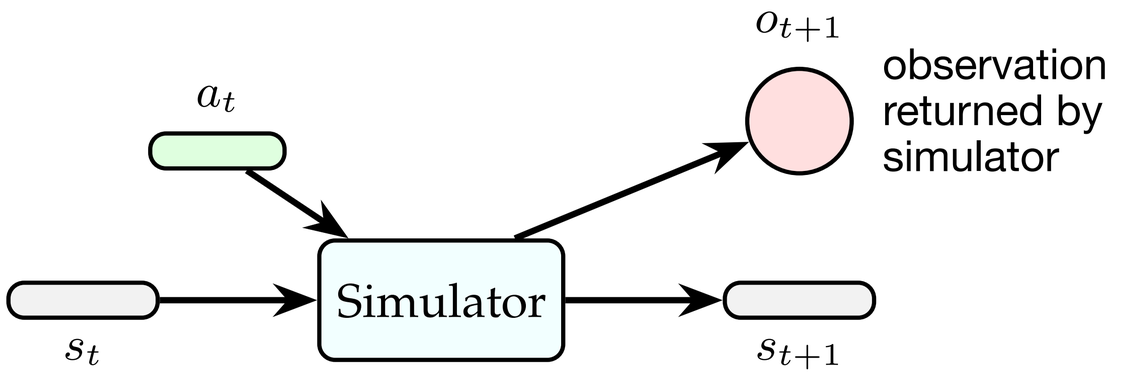
- Do not learn T and O from data; instead, use an external simulator (or physical world) as the ground-truth world model.
- Agent queries the simulator for next state and observation.

Instead of *learning from scratch*, these models use a simulator or real-world environment to test actions and outcomes:

→ bypasses the need to learn $\hat{T}_\theta$ from data.

- Environment update handled by simulator:

$$(s_{t+1}, o_{t+1}) \leftarrow \text{SIM}(s_t, a_t)$$



Simulator-Based World Models: **SimulAted Part-based Interactive ENvironment = SAPIEN**

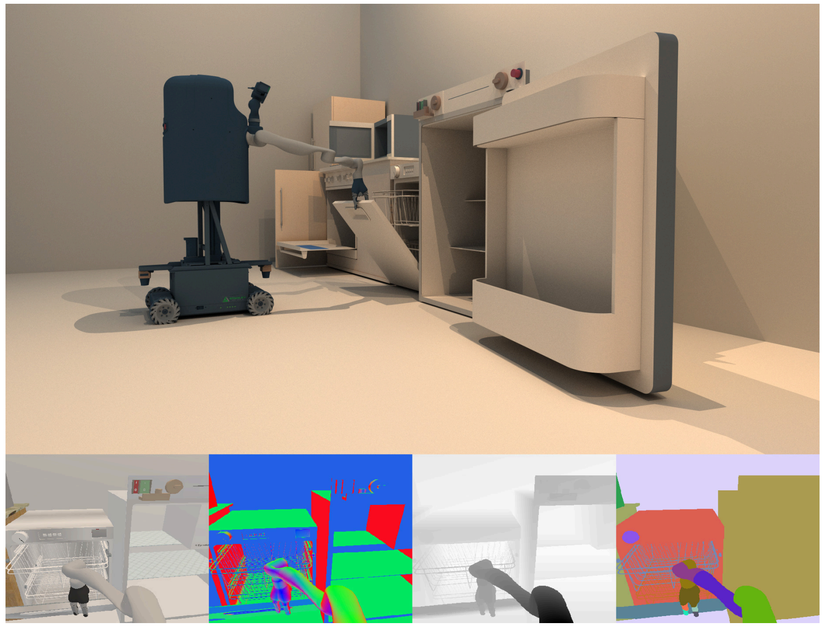
Open-source, **physics-rich simulation platform** for robotics and embodied AI.

Provides:

- High-quality rigid-body and articulated-object dynamics.
- Contact-rich interactions (grasping, pushing, tool use).
- Photorealistic rendering for vision tasks.

$$(s_{t+1}, o_{t+1}) = \mathrm{SAPIEN}\,(s_t, a_t)$$

= simulator



ray-traced scene

robot camera views

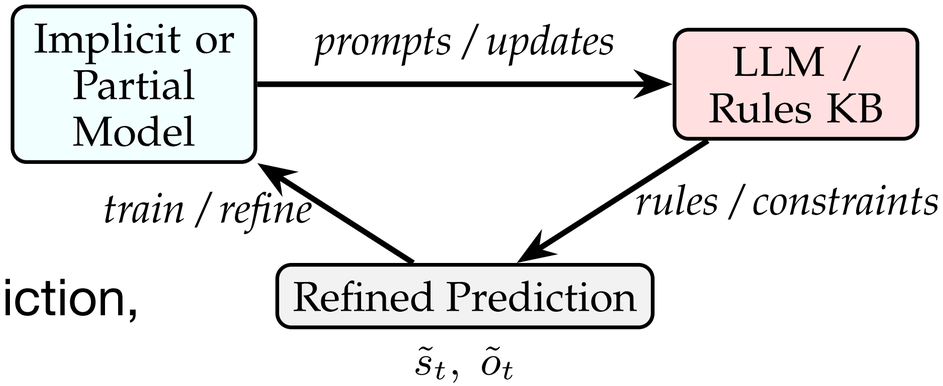
Xiang et al., (2020)

Lecture 13: FM in Robotics

Instruction-Driven World Models

Key idea:

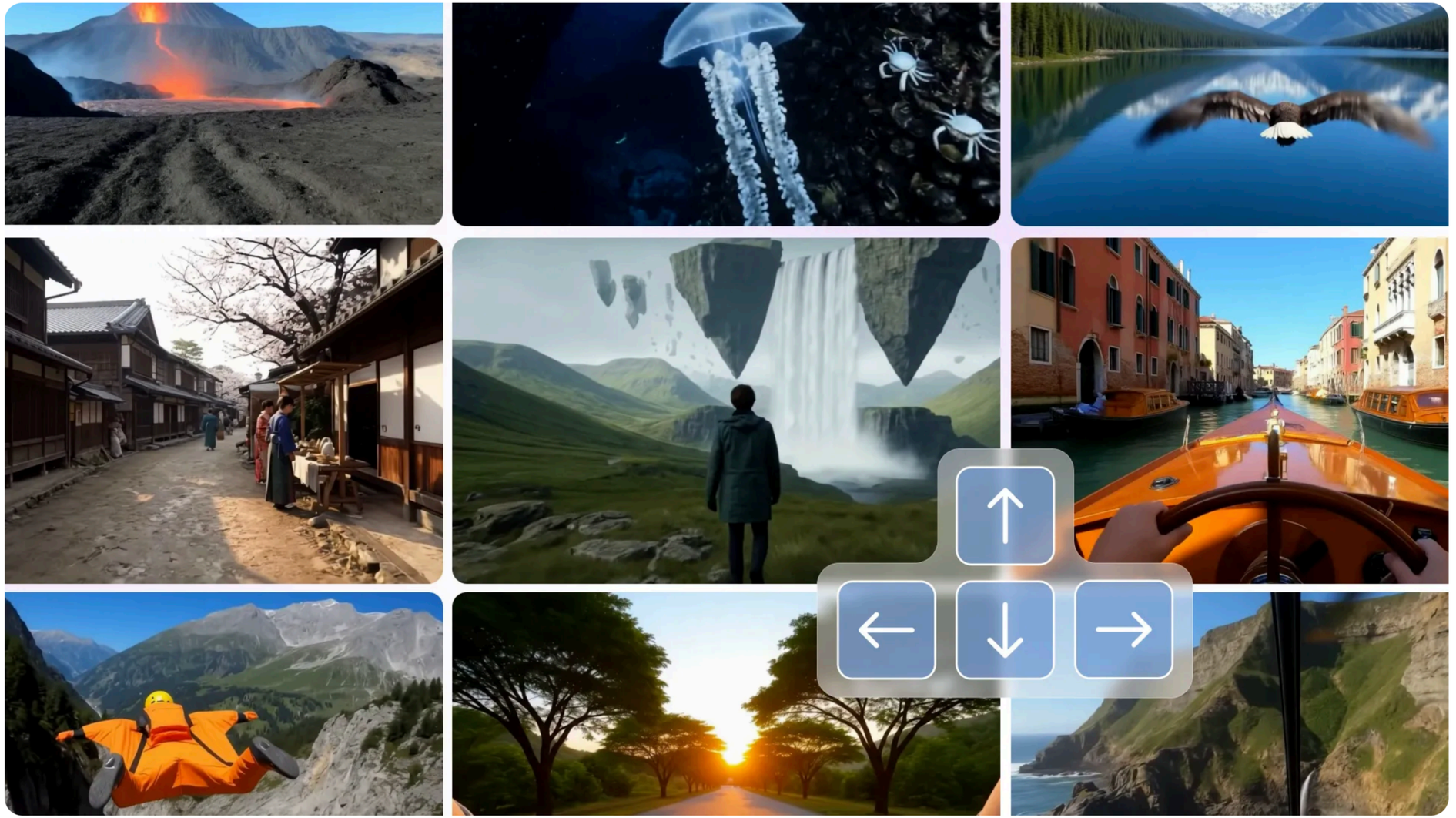
- Blend implicit/explicit neural models with external symbolic knowledge: text rules, code, LLMs.
- World model can be updated or constrained via instructions, manuals, or on-the-fly rule discovery.
- **Components**:
  - Partial implicit/explicit model (latent dynamics and/or decoder).
  - LLM or rule knowledge base that proposes, refines, or checks dynamics / constraints.



LLM / rules KB takes history and provisional prediction, outputs refined prediction or updated rules.

Instruction-Driven World Models: **Genie 3 is Generating Playable Worlds**

Genie 3 can generate an unprecedented diversity of interactive environments.



Instruction-Driven World Models: **The Genie Family**

Given a text prompt, Genie 3 can generate dynamic worlds that you can navigate in real time.

Genie 1: Uses a spatiotemporal video tokenizer, a latent action model inferring discrete actions from unlabeled video, and an **autoregressive dynamics model** to generate 2D interactive environments from single images (including outputs of a text-to-image model, photos, etc.).

Genie 2: An **autoregressive latent diffusion model**: frames pass through an autoencoder, then a large transformer dynamics model with causal masking predicts subsequent latents, using classifier-free guidance for action control, generating 3D environments from single prompt images (via text-to-image model).

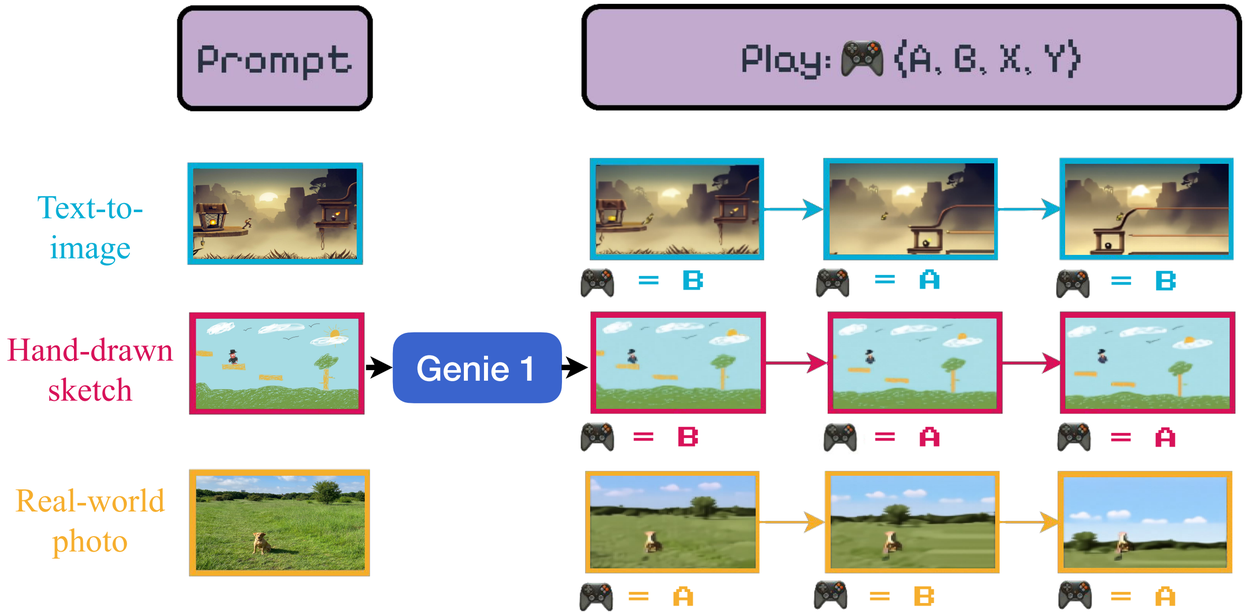
Genie 3: A **frame-by-frame autoregressive world model** attending to the full prior trajectory, optimized for real-time interaction with visual memory, accepting direct text prompts and promptable world events; *detailed architecture undisclosed*.

Instruction-Driven World Models: **Multimodality from Genie 1 to 3**

- **Multimodality** in **Genie 1**

  Uses an *external* **text-to-image model** to turn text into an initial frame, then tokenizes video frames and maps user controls into latent action tokens for an autoregressive video+action token model.

  Multimodality is handled before the core:
  - text → image via T2I,
  - sketches/photos → image,
  - all converted into a single video+action token stream that the Transformer models.



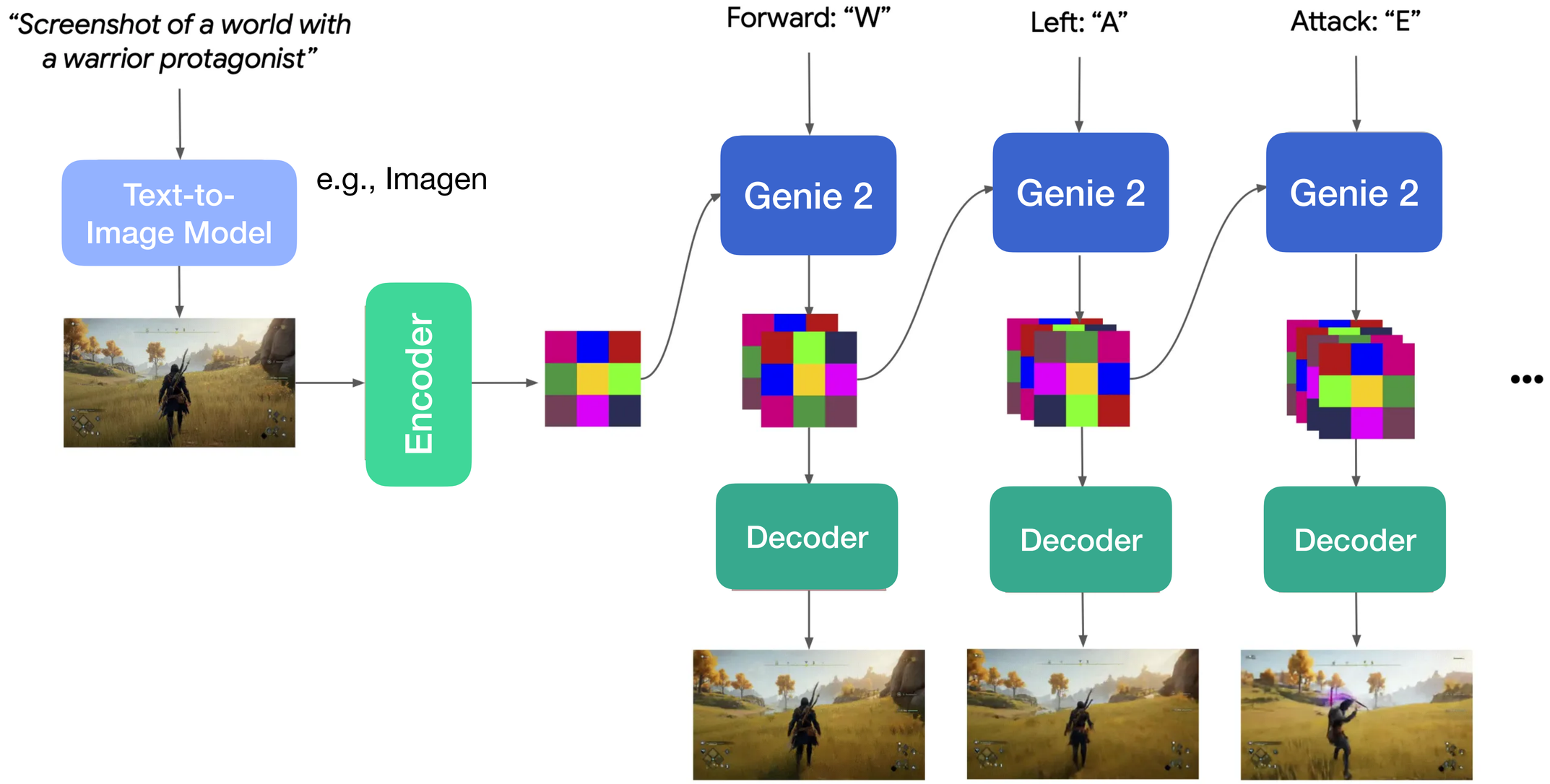
VS.

- **Multimodality** in **Genie 3**

  A single autoregressive world-model Transformer *natively* conditions on text prompts, user actions, and video history to generate video (and sometimes audio).

  Multimodality is handled inside the core:
  text, actions, and visual context are jointly embedded to maintain a persistent world state and produce the next frame.

Instruction-Driven World Models: **Architecture of Genie 2**



Frameworks: **Cosmos World Foundation Models**

The **Cosmos World Foundation Model Platform** is NVIDIA's open platform designed to help developers build customized world models for physical AI systems, e.g., autonomous vehicles, robots, industrial systems.

Platform Components:

Pre-trained World Foundation Models: Generalist models trained on 9,000 trillion tokens, i.e., 20M hours of video

Cosmos Tokenizer: State-of-the-art visual tokenizer
→ 8× more compression, 12× faster than alternatives

Cosmos Curator: Accelerated video processing and data curation pipeline

Cosmos Guardrails: Safety filtering for inputs and outputs

NeMo Framework: For efficient fine-tuning and customization

Frameworks: **Cosmos World Foundation Models**

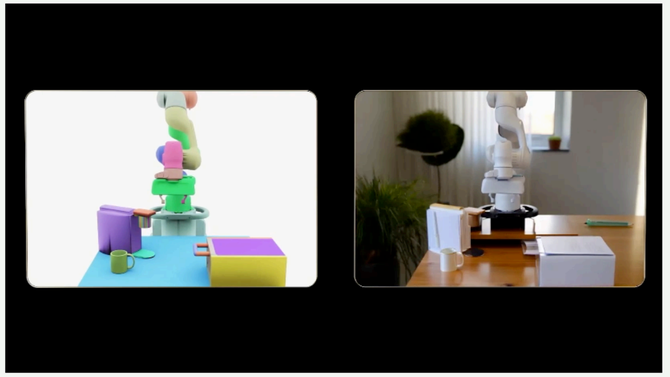
Cosmos World Foundation Models come in **three model types** which can all be **customized in post-training**.

Cosmos Models for Physical AI

Pretrained multimodal generative models that developers can use out-of-the-box for world generation or reasoning, or post-train to develop physical AI models.

Cosmos Predict

A state-of-the-art world state prediction model that can generate up to 30 seconds of continuous video from multimodal inputs with superior speed, fidelity, and prompt adherence. Unlock advanced forecasting and scenario planning for robotics and AI agents by predicting future states of dynamic environments.



Cosmos Transfer

Multicontrol model scales a single simulation or spatial video quickly across various environments and lighting conditions. Accelerate 3D inputs from physical AI simulation frameworks, like CARLA or NVIDIA Isaac Sim™, to enable fully controllable data augmentation and synthetic data generation pipelines.

Cosmos Reason

Open, customizable, reasoning vision language model (VLM) for physical AI lets robots and vision AI agents reason like humans. It can utilize prior knowledge, physics understanding, and common sense to comprehend the real world and how to interact with it.

What is the Next Leap in World Foundation Models?

We are only at the beginning …

… so far mostly video world models that we can control with an agent.

What could come next?

… **multimodal & 3D**: joint models over language, audio, vision, and explicit 3D structure.

… **causal & compositional**: objects, physics, and goals that can be recombined and used for genuine reasoning and zero-shot generalization.

… **agent-integrated**: world FMs that co-train with agents, support planning and tool use, and can be adapted safely to real robots and scientific simulators.

… and more!

World Models for Biology

… other article on Biological World Models

AI & Advanced Computing

Schmidt Sciences awards $18M to researchers working to ensure AI benefits society

Nov 5, 2025

AFFILIATION

Assistant Professor, École Polytechnique Fédérale de Lausanne

HARD PROBLEM

Great Opportunities

Charlotte Bunne

2025 EARLY CAREER FELLOW

Charlotte Bunne is an Assistant Professor at EPFL, jointly appointed in the School of Computer and Communication Sciences (IC) and the School of Life Sciences (SV). She is a member of the EPFL AI Center and the Swiss Institute for Experimental Cancer Research (ISREC), and is affiliated with the Precision Oncology Unit at the University Hospital in Geneva. She previously held postdoctoral positions at Genentech and Stanford, working with Aviv Regev and Jure Leskovec, after completing her PhD in Computer Science at ETH Zurich under the supervision of Andreas Krause and Marco Cuturi. During her doctoral studies, she was a visiting researcher at the Broad Institute of MIT and Harvard and a visiting student at MIT. Her honors include the AI2050 Early Career Fellowship, a Fellowship of the German National Academic Foundation, and two ETH Zurich Medals.

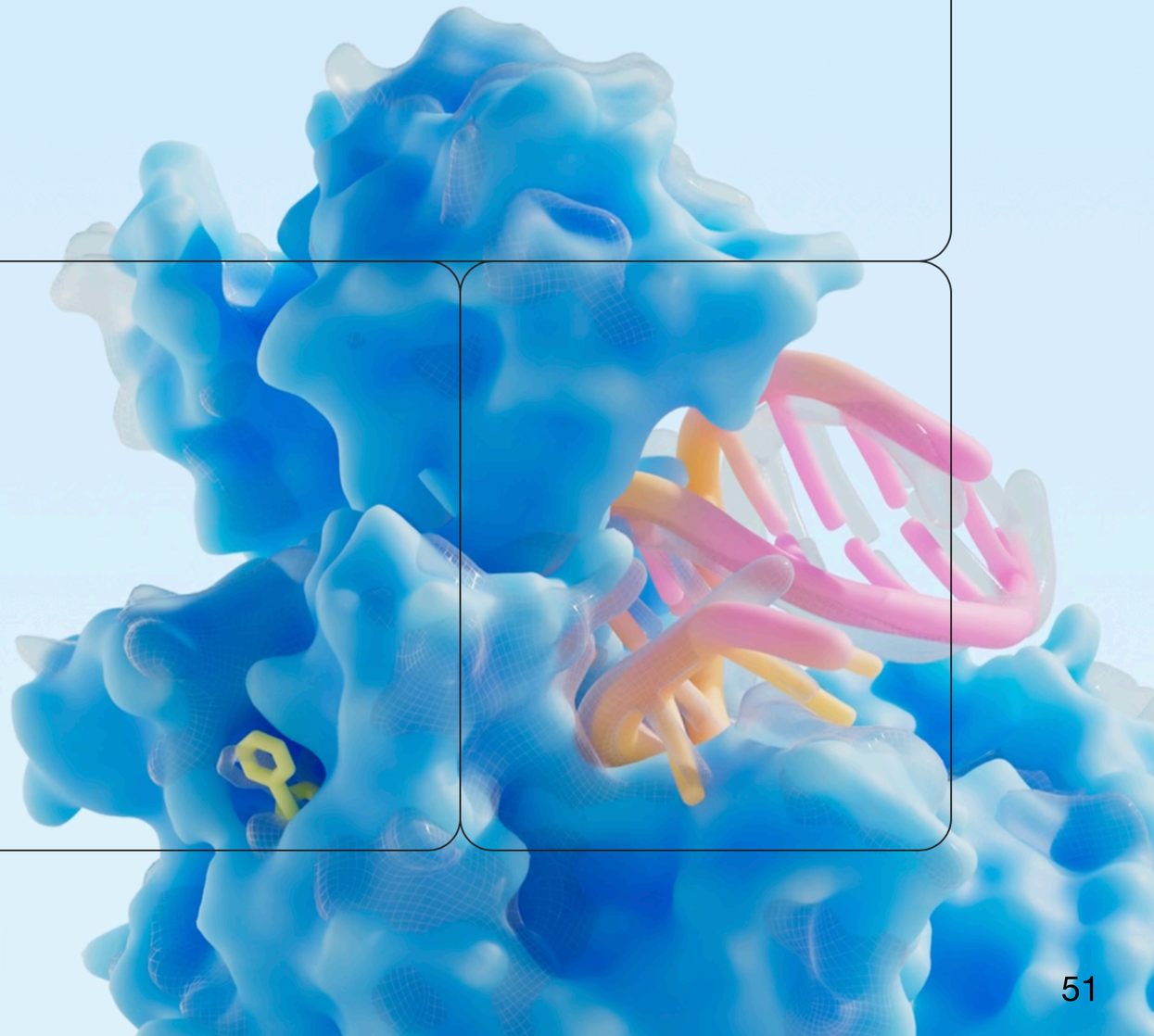
AI2050 Project

Biology needs its own transformative leap in artificial intelligence: moving beyond static snapshots to systems that simulate, understand, and reason about living tissues. Bunne's project introduces biological world models: computational frameworks that integrate multimodal data into structured, spatially and molecularly grounded representations of cellular and tissue organization. Equipped with generative simulators and intelligent reasoning agents in a closed feedback loop, these models forecast system dynamics, test hypotheses, and optimize therapeutic strategies. Validated through collaborations with experimental and clinical partners, this approach lays the foundation for simulation-based discovery and decision support, with initial applications in cancer treatment prediction.

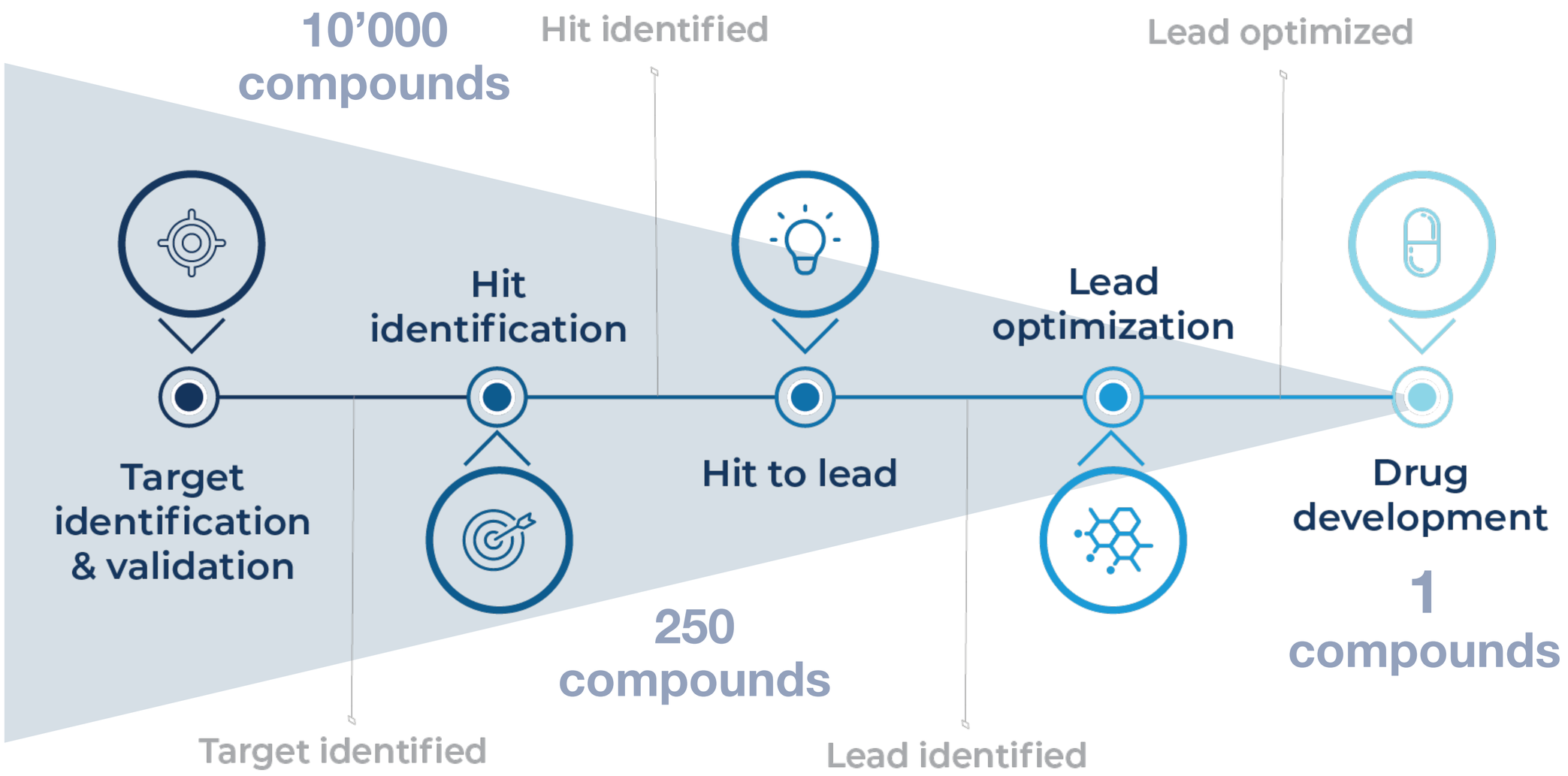
World Models for Biology e.g., **drug design**

Solve all disease

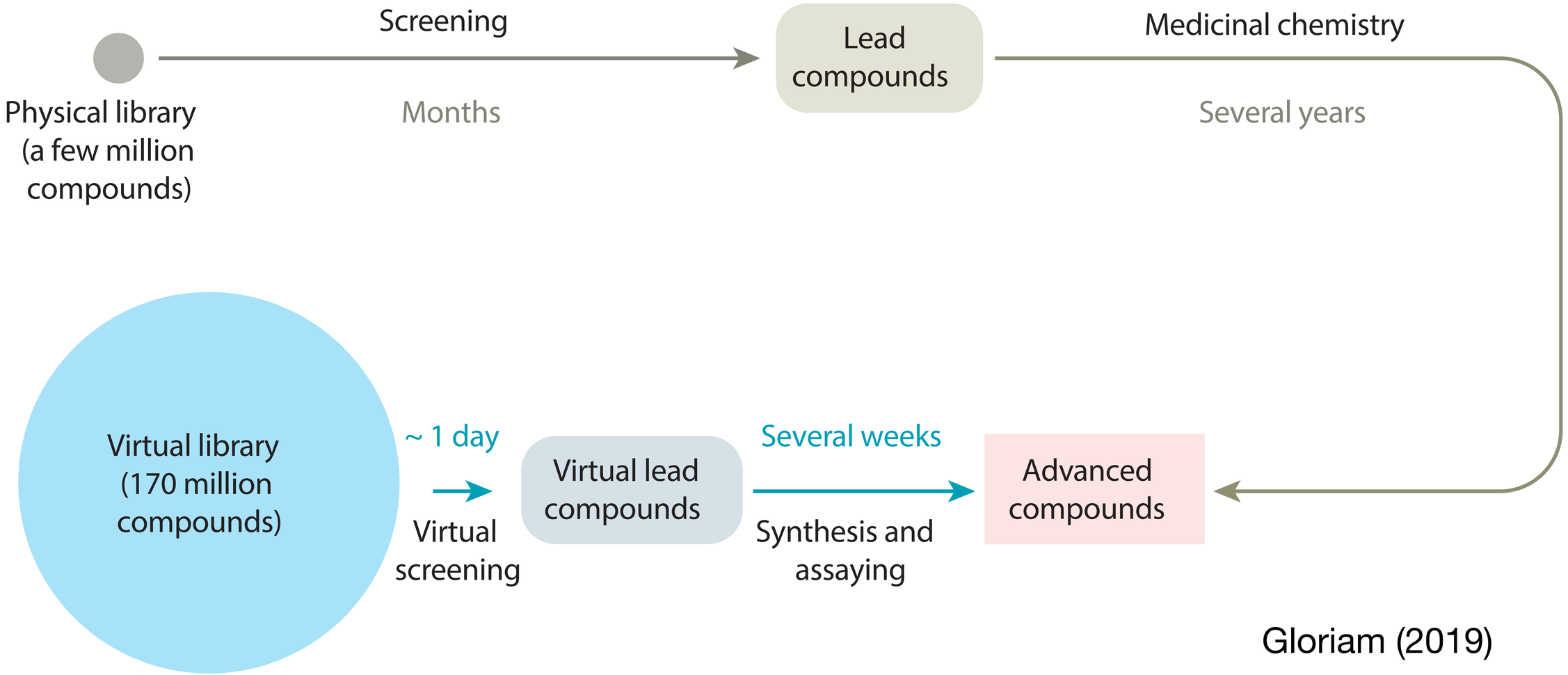
We’re entering a new era of drug discovery — one where frontier AI can unlock deeper scientific insights, faster breakthroughs, and life-changing medicines. At Isomorphic Labs, we’re building that future.



World Models for Biology e.g., **drug design**



World Models for Biology e.g., **drug design**

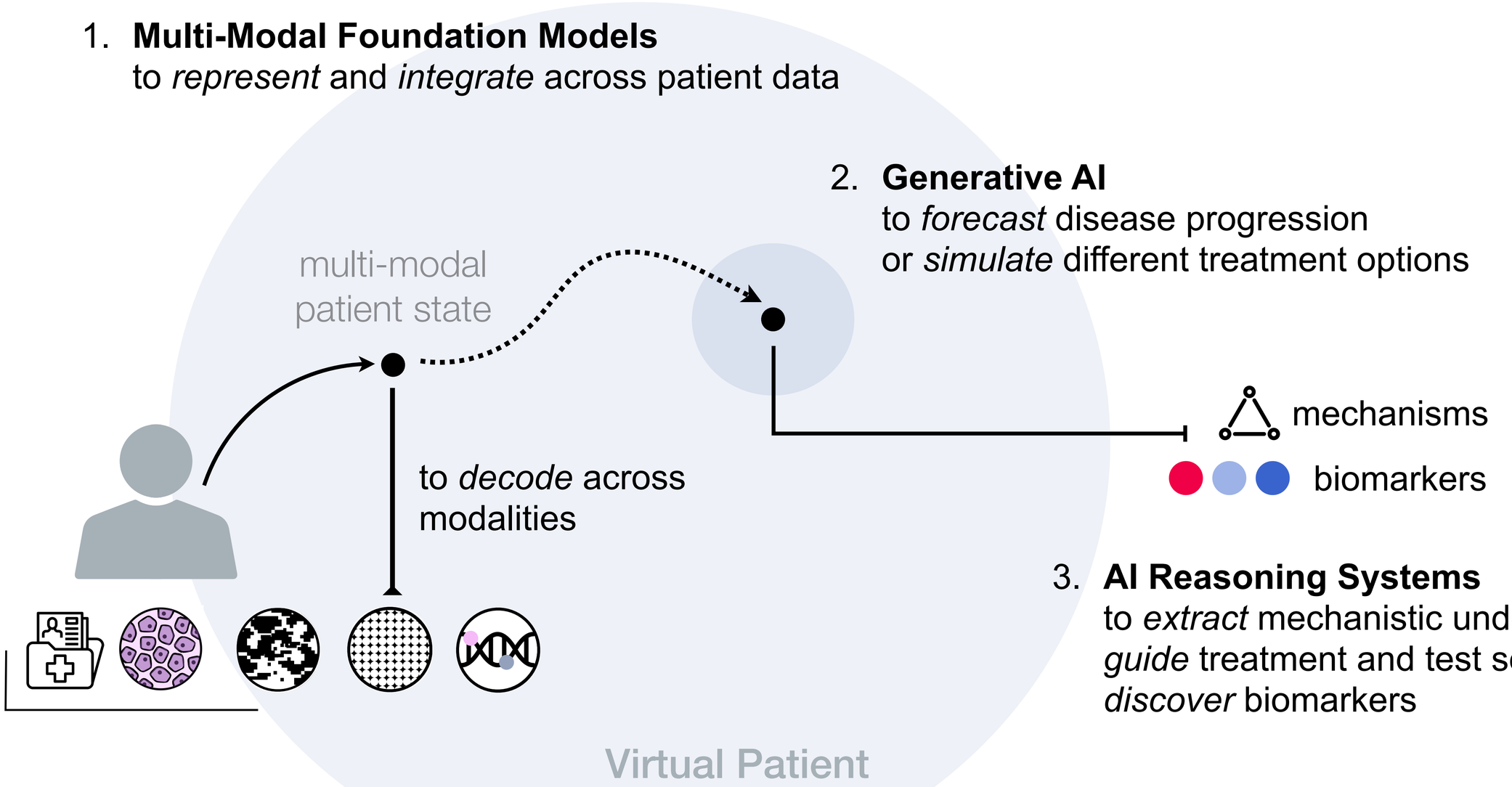


Gloriam (2019)

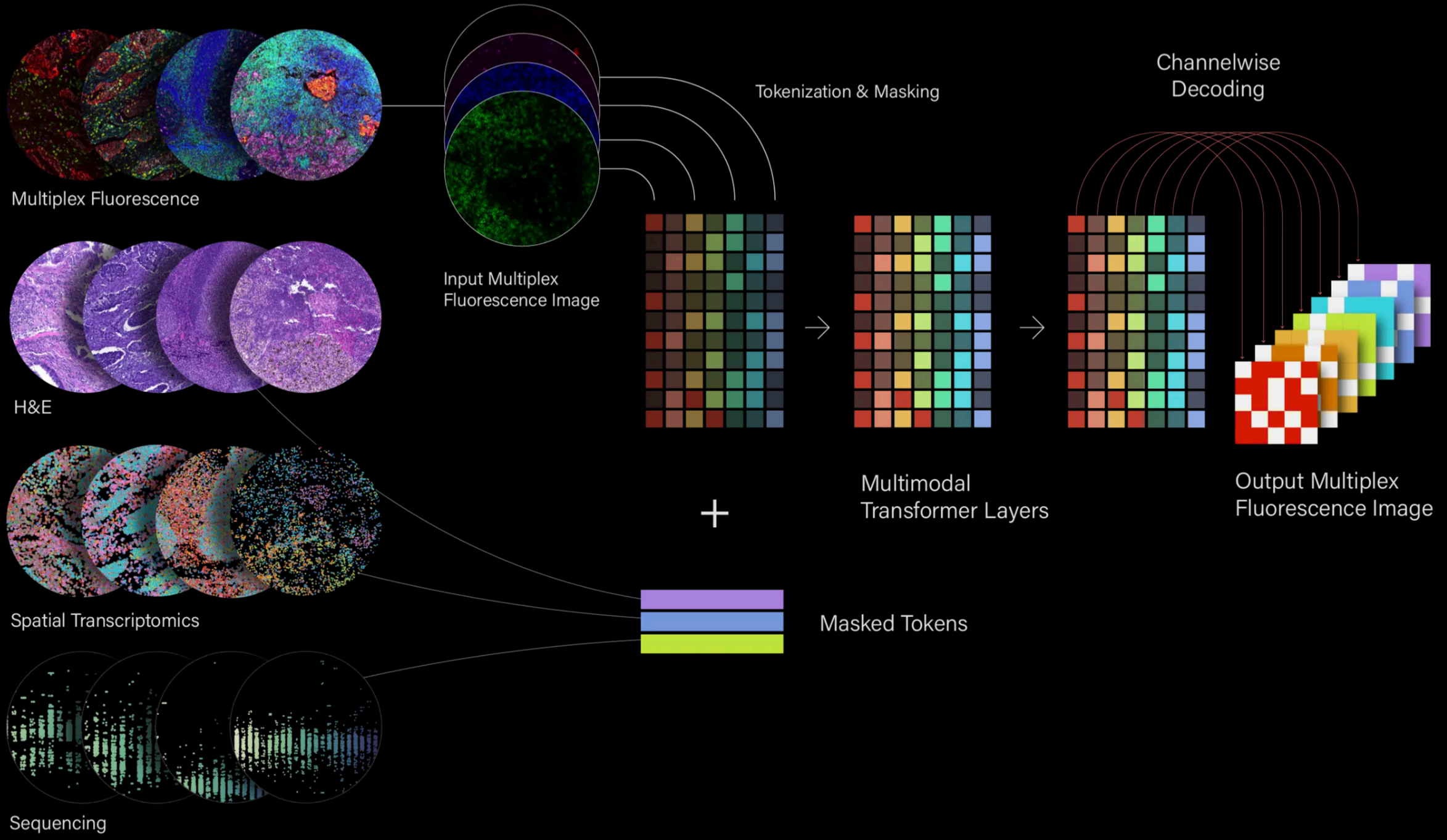
Next Generation AI Systems for Precision Medicine

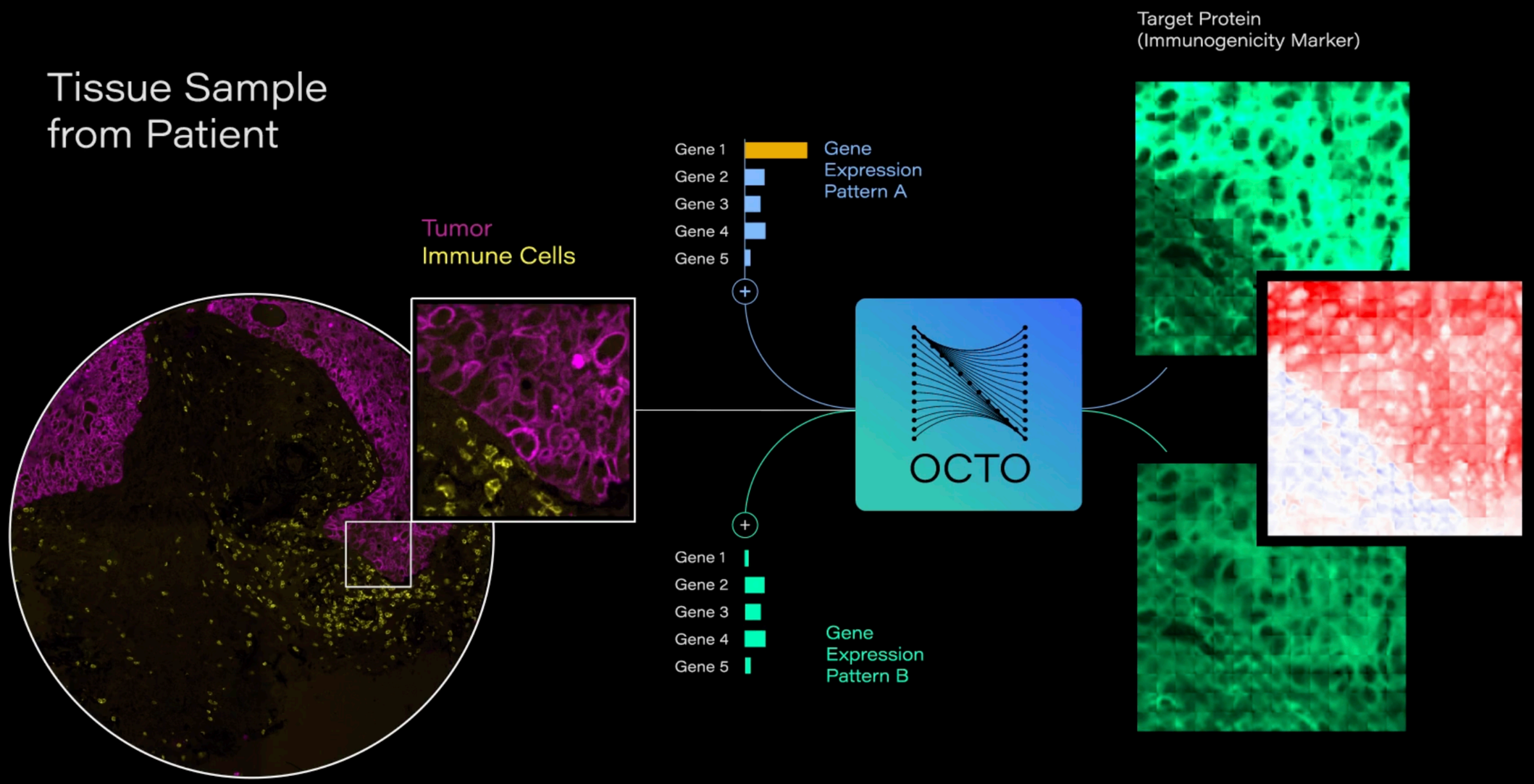
1. **Multi-Modal Foundation Models**
   to *represent* and *integrate* across patient data

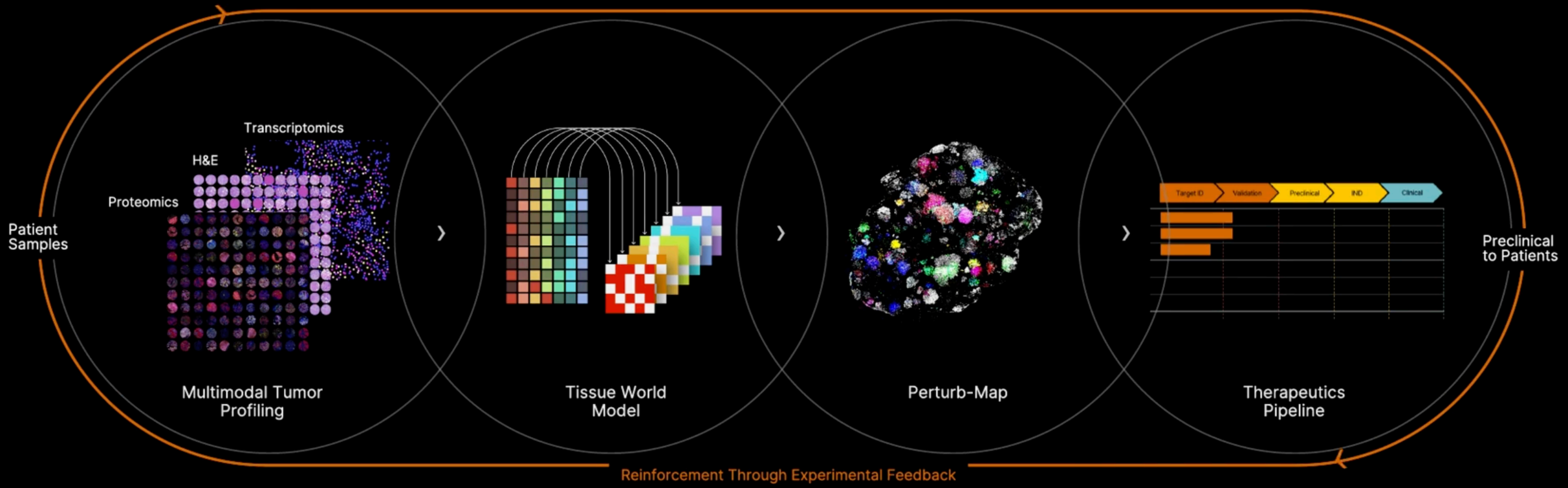
2. **Generative AI**
   to *forecast* disease progression
   or *simulate* different treatment options



3. **AI Reasoning Systems**
   to *extract* mechanistic understanding,
   guide treatment and test selection and
   discover biomarkers


Multiplex Fluorescence
H&E
Spatial Transcriptomics
Sequencing
Input Multiplex
Fluorescence Image
Tokenization & Masking
+
Masked Tokens
Multimodal
Transformer Layers
Channelwise
Decoding
Output Multiplex
Fluorescence Image


Tissue Sample
from Patient
Tumor
Immune Cells
Gene 1
Gene 2
Gene 3
Gene 4
Gene 5
Gene
Expression
Pattern A
Gene 1
Gene 2
Gene 3
Gene 4
Gene 5
Gene
Expression
Pattern B
OCTO
Target Protein
(Immunogenicity Marker)
Higher for
Expression
Pattern A
Higher for
Expression
Pattern B



Human Platform
Multimodal spatial data purpose-built for self supervised learning

Multimodal AI
Self supervised foundation models of tumor biology

Mouse Platform
Scaled In Vivo validation of dozens of targets in the same mouse

Pipeline
Multiple patient-specific immunotherapy programs

This Week’s Exercise Sheet

Multiple Choice Questions from the Lecture Slides.

Exercise 11 · Task 1

Some parts of the exam are MCQ that test conceptual knowledge.

This Week's Code Demonstration

Code Notebook 11 · Task 1

Diving into the Implementation of JEPA

1. Starting from the image-based JEPA (I-JEPA). Implementing and understanding each core component (context encoder, target encoder, predictor network, masking-strategy, and overall training objective).

2. Extending from I-JEPA to the video-based version (V-JEPA). Understanding the difference from I-JEPA and why learning representations from video is crucial for many forms of world modeling.

More in the exercise session: Discussion on current world modeling schemes.

This Week's Papers

Papers are linked in Moodle.

ADVANCES AND CHALLENGES IN FOUNDATION AGENTS

FROM BRAIN-INSPIRED INTELLIGENCE TO EVOLUTIONARY, COLLABORATIVE, AND SAFE SYSTEMS

Bang Liu[1,2,20*†], Xinfeng Li[3*], Jiayi Zhang[4,10*], Jinlin Wang[5*], Tanjin He[6*], Sirui Hong[5*], Hongzhang Liu[4*], Shaokun Zhang[7*], Kaitao Song[8*], Kunlun Zhu[9*], Yuheng Cheng[3*], Suyuchen Wang[1,2*], Xiaoqiang Wang[20,21], Yuyu Luo[10], Haibo Jin[11], Peiyan Zhang[12], Ollie Liu[13], Jiaqi Chen[14], Huan Zhang[15], Zhaoyang Yu[5], Haochen Shi[2], Boyan Li[16], Dekun Wu[2,3], Fengwei Teng[3], Xiaojun Jia[3], Jiawei Xu[1], Jinyu Xiang[5], Yizhang Lin[5], Tianming Liu[14], Tenglong Liu[5], Yu Su[15], Huan Sun[15], Glen Berseth[2,3,20], Jianyun Nie[2], Ian Foster[8], Logan Ward[8], Qingyun Wu[7], Yu Gu[15], Mingchen Zhuge[16], Xiangru Tang[17], Haohan Wang[18], Jiaxuan You[9], Chi Wang[19], Jian Pei[21], Qiang Yang[3,10,11], Xiaoliang Qi[3,11], Chenglin Wu[1,2†]

[1]MetaGPT, [2]Université de Montréal, [3]Mila - Quebec AI Institute, [4]Penn State University, [5]Nanyang Technological University, [6]Argonne National Laboratory, [7]Microsoft Research Asia, [8]University of Illinois at Urbana-Champaign, [9]University of Sydney, [10]The Hong Kong University of Science and Technology (Guangzhou), [11]University of Southern California, [12]Yale University, [13]Stanford University, [14]University of Georgia, [15]The Ohio State University, [16]King Abdullah University of Science and Technology, [17]Yale University, [18]Duke University, [19]The Hong Kong Polytechnic University, [20]Google DeepMind, [21]Canada CIFAR AI Chair

Abstract

The advent of large language models (LLMs) has catalyzed a transformative shift in artificial intelligence, paving the way for advanced intelligent agents capable of sophisticated reasoning, robust perception, and versatile action across diverse domains. As these agents increasingly drive AI research and practical applications, their design, evaluation, and continuous improvement present intricate, multifaceted challenges. This book provides a comprehensive overview, framing intelligent agents within modular, brain-inspired architectures that integrate principles from cognitive science, neuroscience, and computational research. We structure our exploration into four interconnected parts. First, we systematically investigate the **modular foundation of intelligent agents**, systematically mapping their cognitive, perceptual, and operational modules onto analogous human brain functionalities and elucidating core components such as memory, world modeling, reward processing, goal, and emotion. Second, we discuss **self-enhancement and adaptive evolution mechanisms**, exploring how agents autonomously refine their capabilities, adapt to dynamic environments, and achieve continual learning through automated optimization paradigms. Third, we examine **collaborative and evolutionary multi-agent systems**, investigating the collective intelligence emerging from agent interactions, cooperation, and societal structures, highlighting parallels to human social dynamics. Finally, we address the critical imperative of **building safe and beneficial AI systems**, emphasizing intrinsic and extrinsic security threats, ethical alignment, robustness, and practical mitigation strategies necessary for trustworthy real-world deployment.

*Major Contribution.
†Corresponding authors: Bang Liu (bang.liu@umontreal.ca), Jian Pei (j.pei@duke.edu), Qiang Yang (qyang@cse.ust.hk), Xiaoliang Qi (xlqi@stanford.edu), Chenglin Wu (alexanderwu@deepwisdom.ai)

Liu, Bang, et al. "Advances and Challenges in Foundation Agents: From Brain-Inspired Intelligence to Evolutionary, Collaborative, and Safe Systems." *arXiv preprint arXiv:2504.01990* (2025).

→ **Chapter 1 and 4**

CS-461

Foundation Models and Generative AI

Have a great week!