Section 9: Agent World Model

2026 Spring

LLM Agents Foundation & Applications

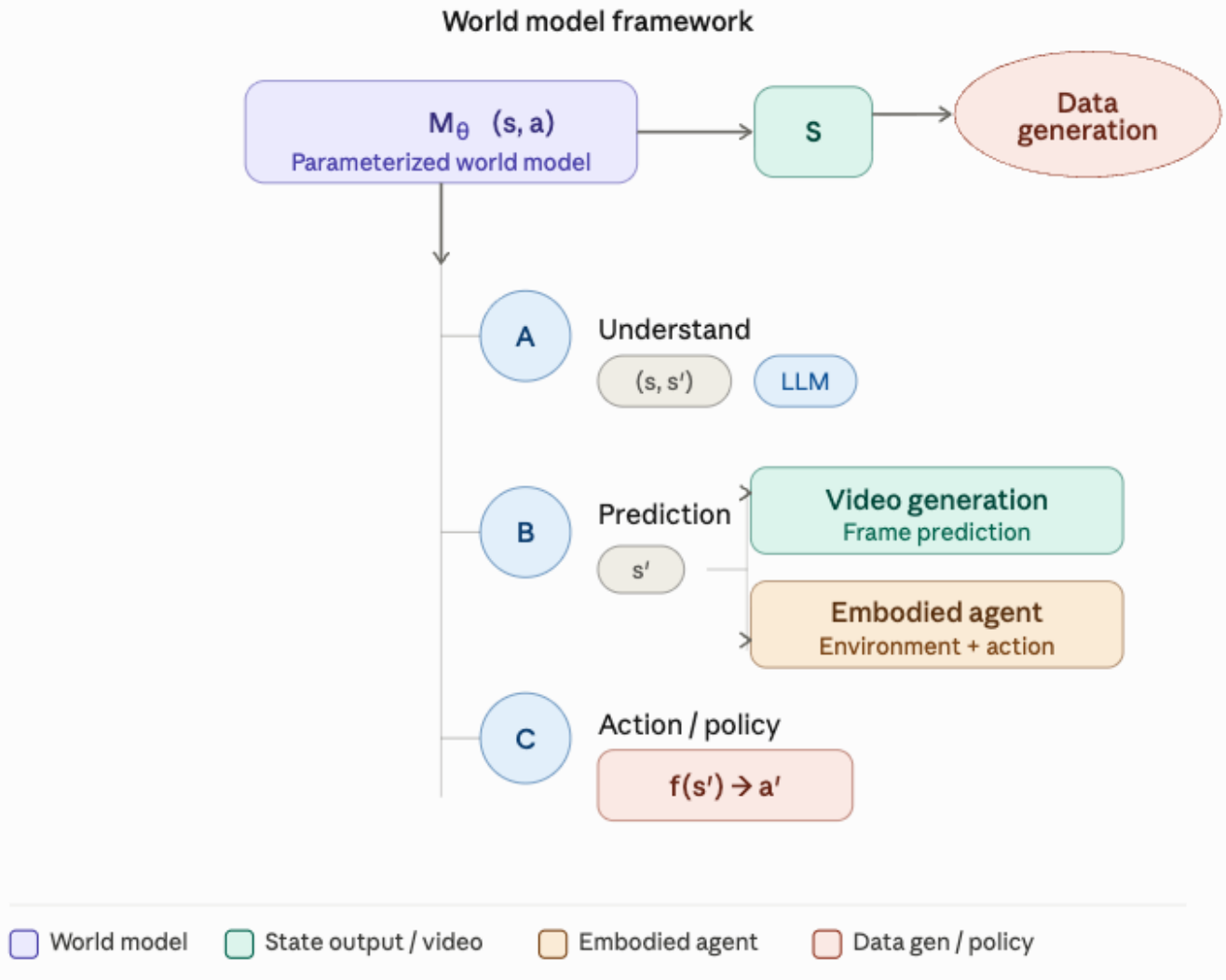
Student Team / 20260414

Roadmap

- S1: LLM Basic Alignment
- S2: LLM Alignments for Reasoning
- S3: Agent applications
- S4: LLM Data synthesis
- S5: Agent Memory
- S6: LLM model serving
- S7: Agent Evaluation and Attack/Defense Landscape
- S8: Agent Planning / Testing Time Scaling
- S9: World Modeling for GenAI Agents ➔ This lecture!
- S10: Multi-Agents

Three papers:

- 1. **Understanding World or Predicting Future?**
- **2.** World Models for Physical AI
- **3. Mastering Diverse Domains through World Models**


World model framework
M_θ (s, a)
Parameterized world model
S
Data generation
A
Understand
(s, s′)
LLM
B
Prediction
s′
Video generation
Frame prediction
Embodied agent
Environment + action
C
Action / policy
f(s′) → a′
World model
State output / video
Embodied agent
Data gen / policy

Understanding World or Predicting Future?

A Comprehensive Survey of World Models

Guangyi (Mark) Xu

Why do we need World Model?

AGI goal → need models that:

- understand environment
- predict future

Examples:

- GPT-4 (knowledge)
- Seedance (video prediction)

What is a World Model?

Core Capabilities

A model that:

- Learns **world dynamics**
- Predicts **future states**
- Supports **decision making**

“understanding the world vs predicting the future”

Key Clarifications

What it is not:

- **Not a single architecture**
- It is a **concept/framework**

Eample:

Resnet / GPT trained only for text completion / BERT for QA

History

1. Pre-Deep Learning

Before ~2015

- Symbolic AI (frames, logic)
- Model-based RL
- Explicit environment dynamics

2. 2018 Neural Models

World Models

- Ha & Schmidhuber
- Latent representation + RNN dynamics
- “Imagination” for planning

3. 2022 Representation

JEPA

- LeCun’s Architecture
- Predict **latent space**, not pixels
- Self-supervised understanding
- Understanding the world not just reconstruction

4. 2023-Now Generative

Modern Era

- LLMs → world knowledge
- Video models (e.g., Sora)
- Interactive environments

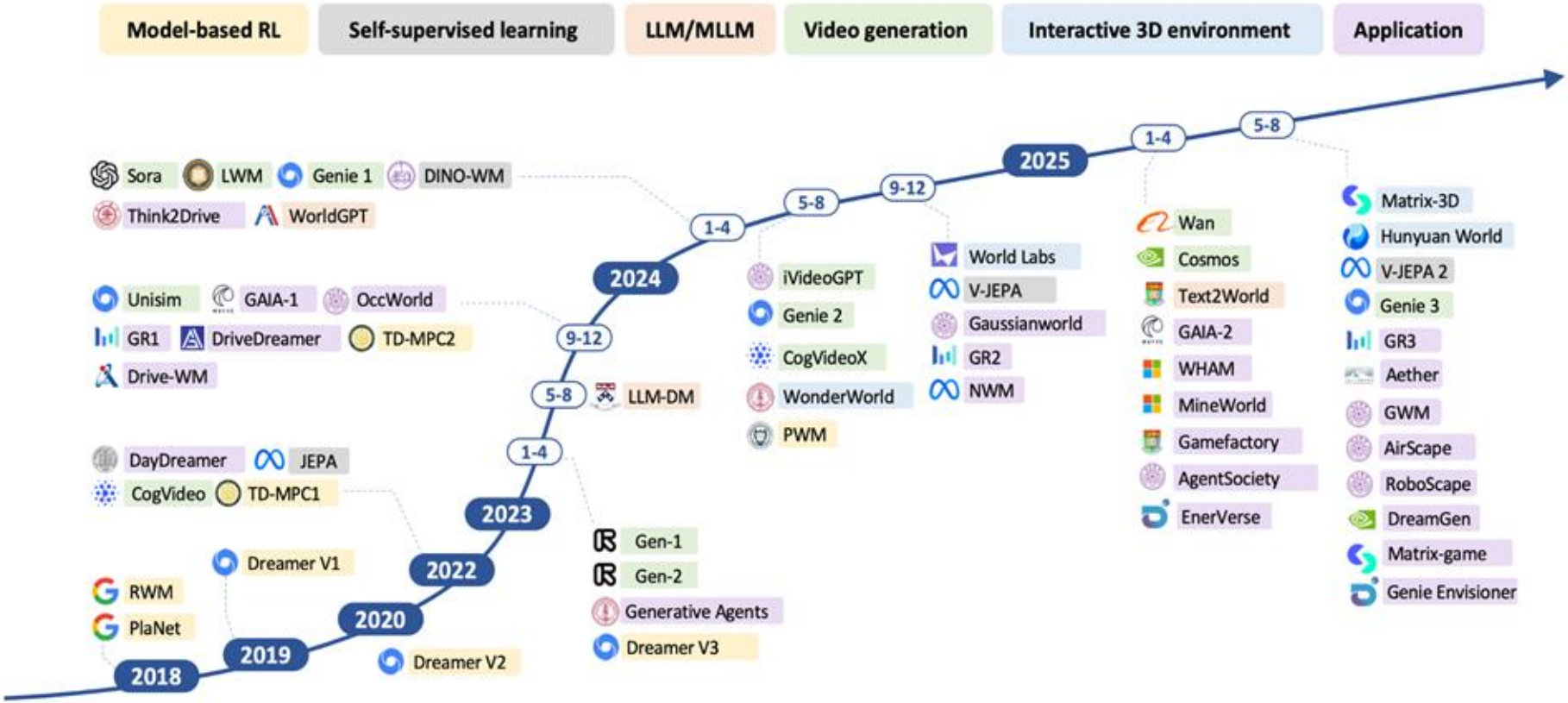
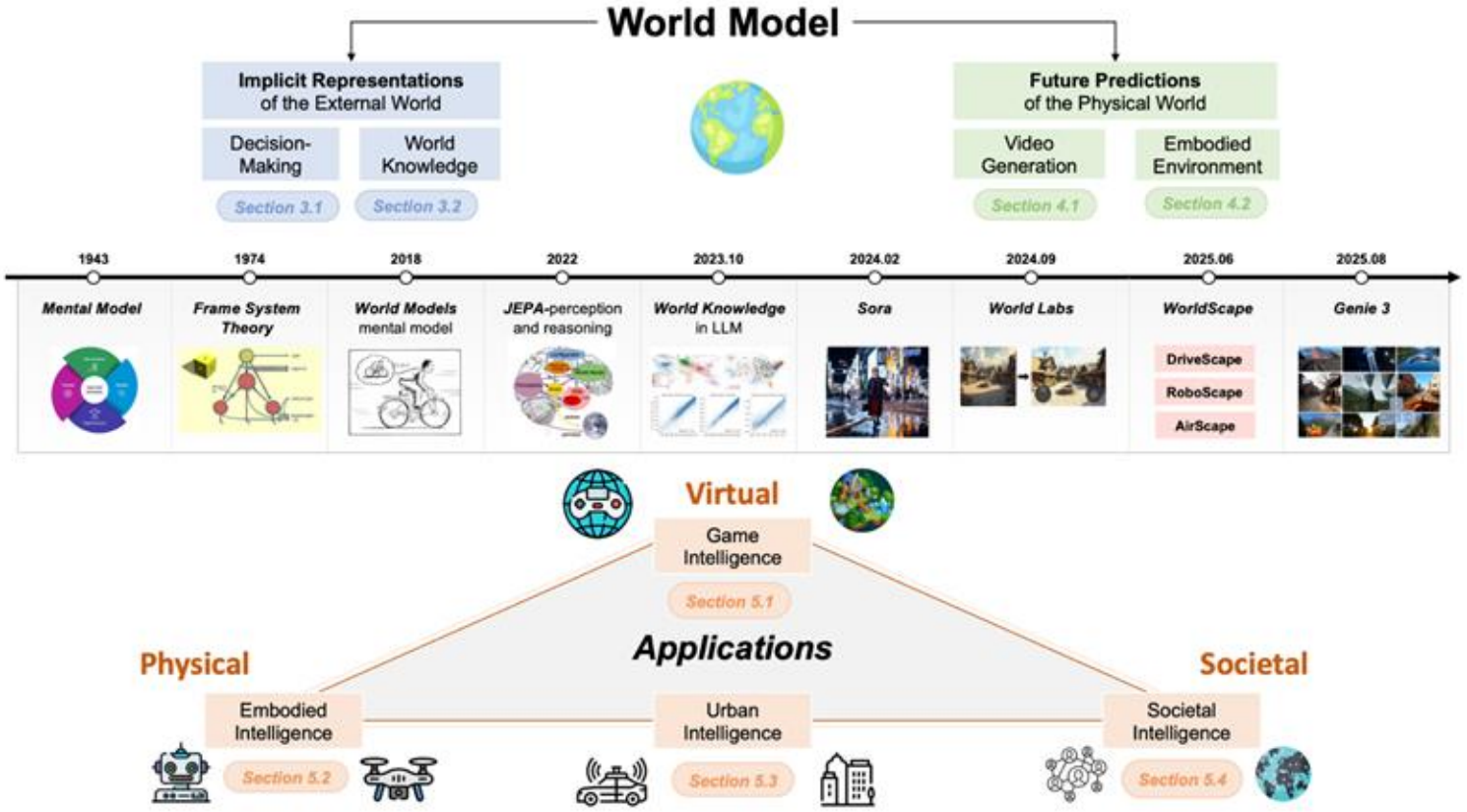


Fig. 1. The roadmap of world models in deep learning era.

Two Core Perspectives



PART 1: UNDERSTANDING

World Model as Understanding (Implicit Representation)

Core idea:

- Learn latent structure of world
- Used for reasoning / planning

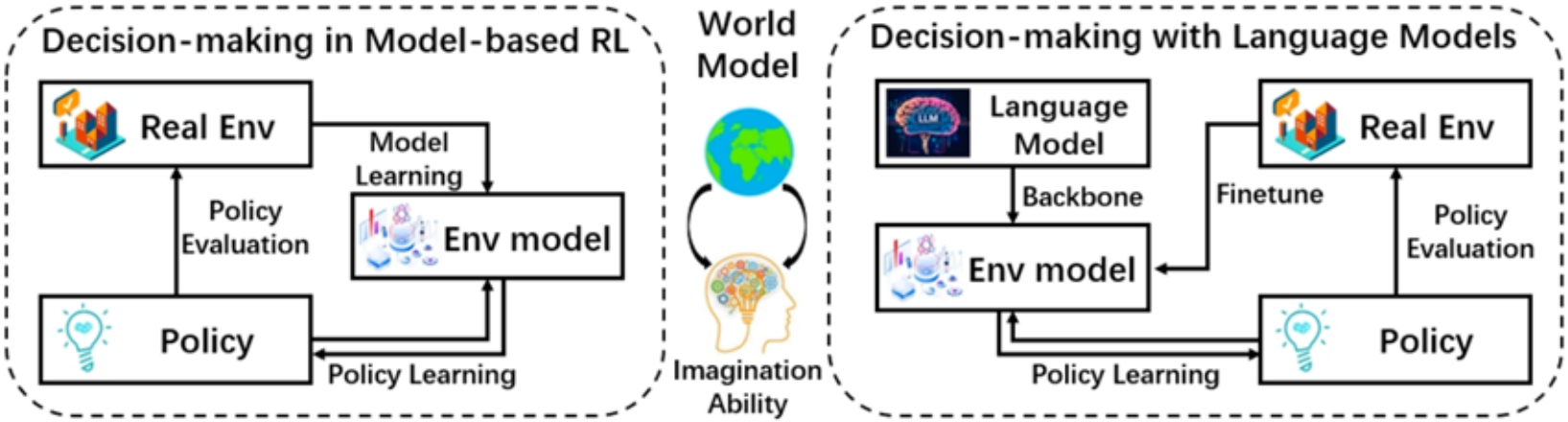


Fig. 3. Two schemes of utilizing world model in decision-making.

Model-Based RL World Models

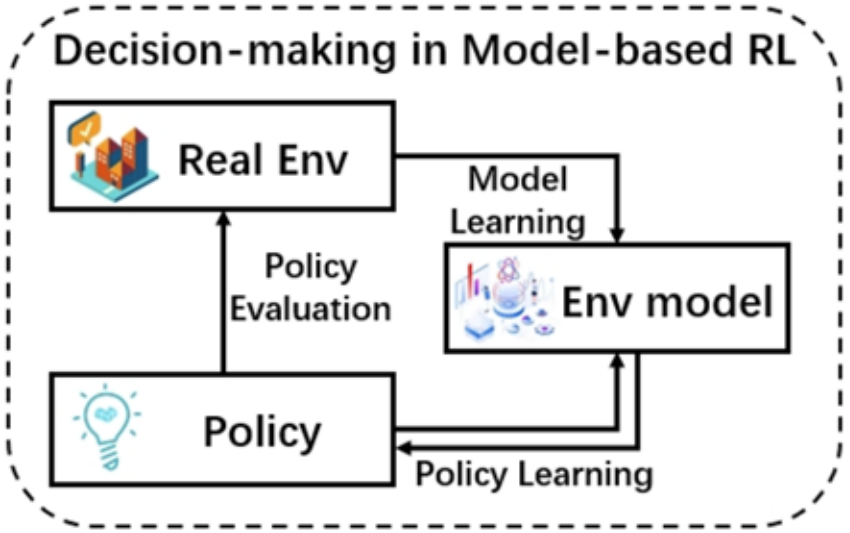
MDP(Markov Decision Process) Components

- State (S)
- Action (A)
- Transition (M)
- Reward (R)

Key Idea

Learn **environment dynamics** to predict transitions.

`state → action → next state`



The interaction loop in Model-Based RL systems

Learning the World Model

Concept:

- predict next state
- minimize prediction error

Key insight:

- world model = supervised learning on transitions
- leverage the mean squared prediction error on each one-step transitions

$$\min_{\theta} \mathbb{E}_{s' \sim M^*(\cdot|s,a)} [||s' - M_\theta(s,a)||_2^2],$$

Planning with World Models

Key Idea

simulate future → choose best action

Model Predictive Control (MPC)

An iterative process of planning and optimization over a finite time horizon.

Monte Carlo Tree Search (MCTS)

A probabilistic search algorithm for finding the most promising moves in a tree.

Techniques for decision-making using learned or internal models

LLM-based World Models

Core Concept: Large Language Models (LLMs) contain rich world knowledge.

Capabilities

- Plan complex sequences of actions
- Simulate dynamic environments

Examples

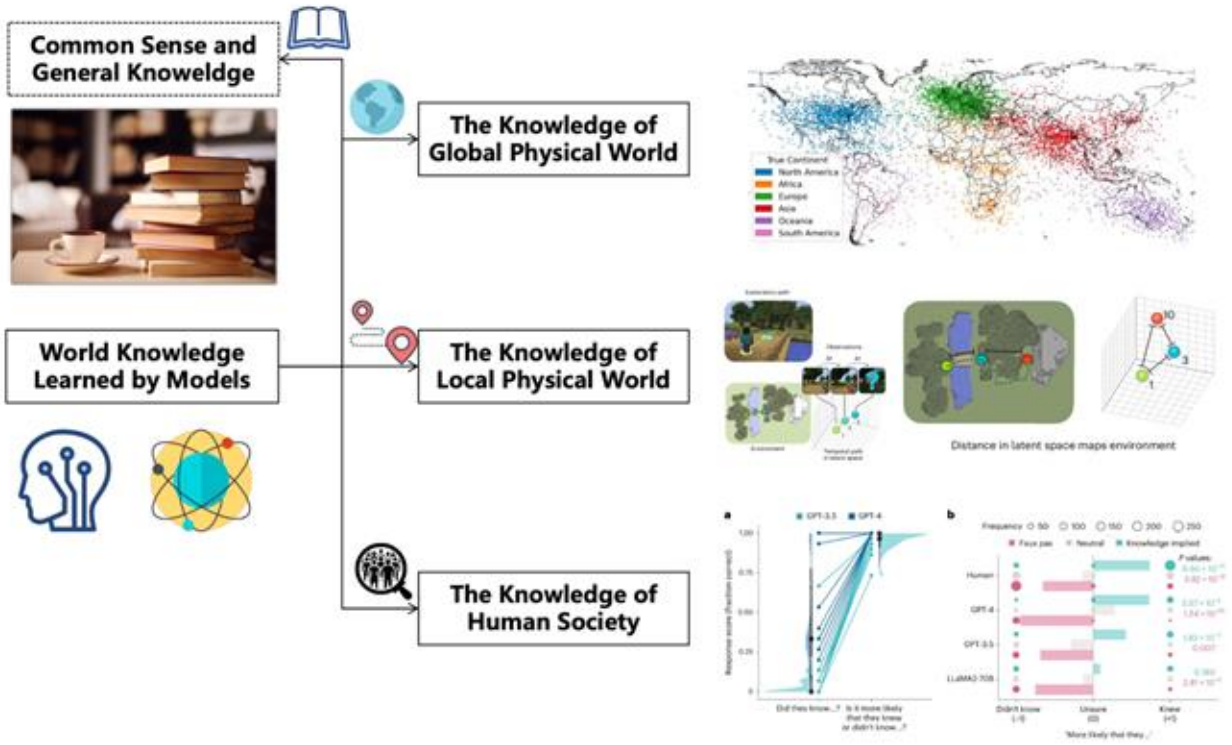
- GPT-based planning systems
- Autonomous web agents

Leveraging pre-trained knowledge for agentic reasoning and environmental interaction

World Knowledge in LLMs

Three types:

1. Global physical world
2. Local physical world
3. Human society



Conclusion: Key points

World model = latent representation

Enables:

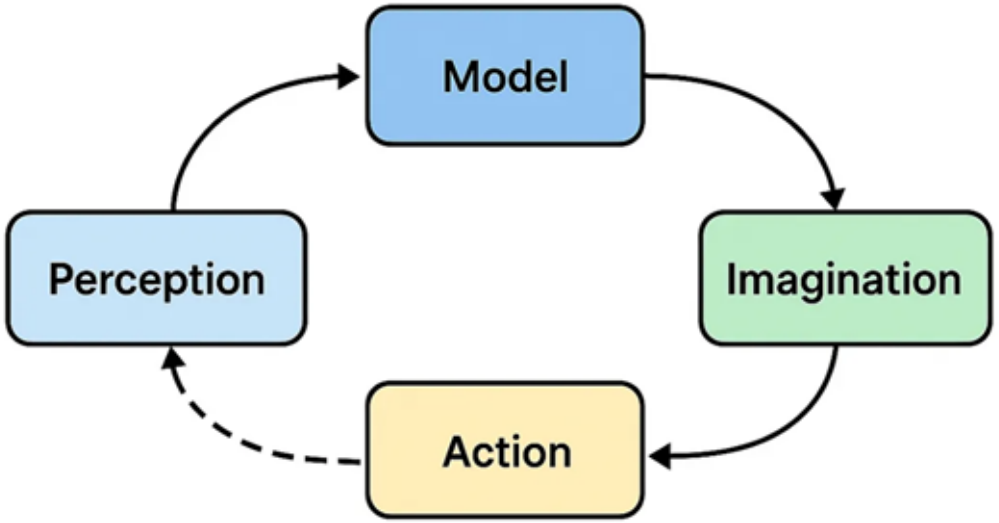
- reasoning
- planning
- decision making

Part 2: PREDICTION

“World Model as Prediction (Simulation)”

Core:

- simulate future states
- generate environments



Video World Models

EXPLAIN

Systems that learn to **generate future frames** based on current visual data.

EXAMPLES

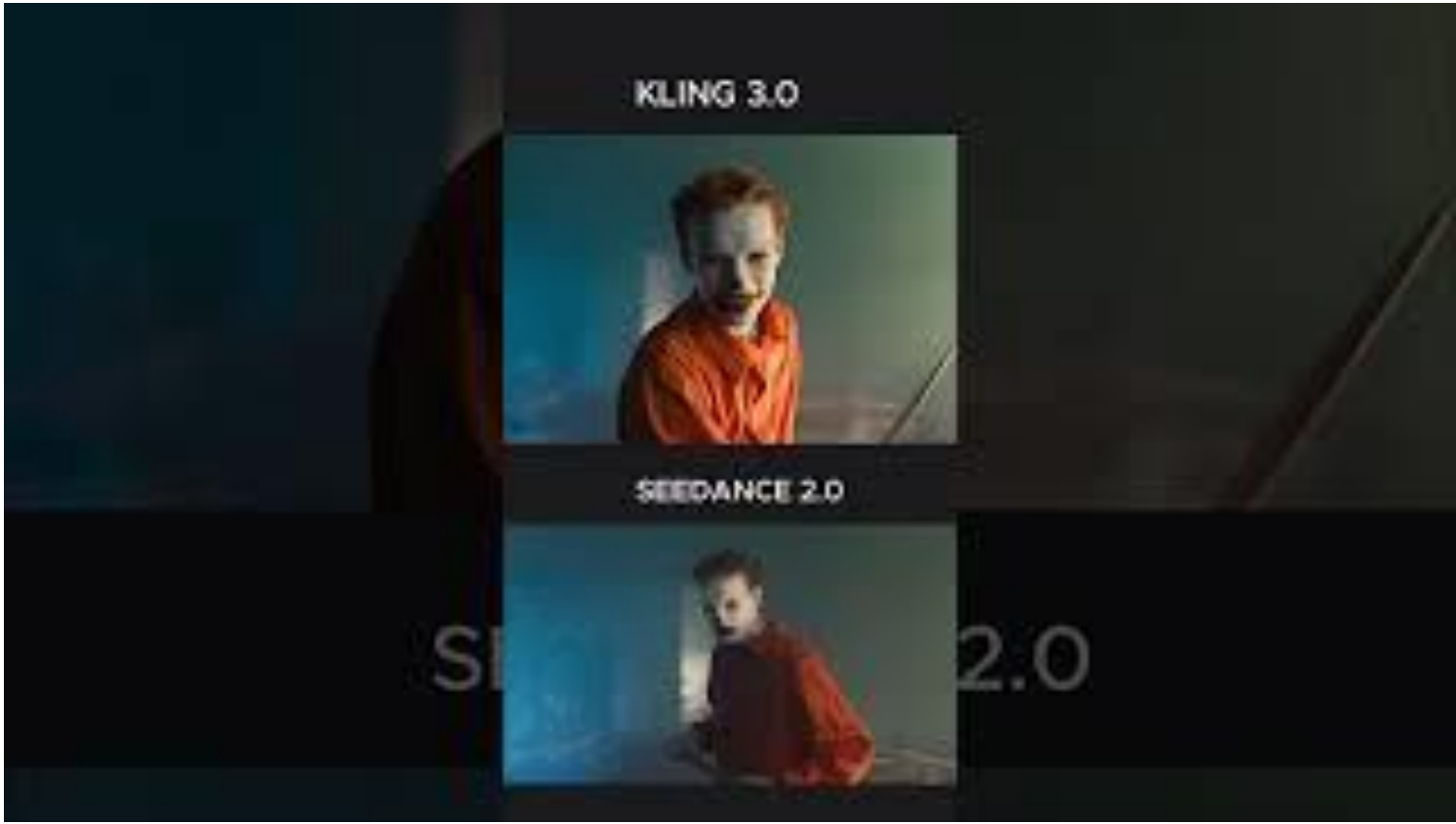
- Sora
- CogVideo
- Seedance

KEY IDEA

world =
video dynamics

Limitations of Video Models

- weak causal reasoning
- physics inconsistency



What Makes a True World Model?

A true world model requires four core capabilities:

timeline

Long-term prediction

Projecting future states accurately over extended horizons.

layers

Multi-modal understanding

Integrating text, visual, and sensory data.

touch_ap p

Interactivity

Responding to actions and feedback in real-time.

public

Diverse environments

Generalizing across various physical scenarios.

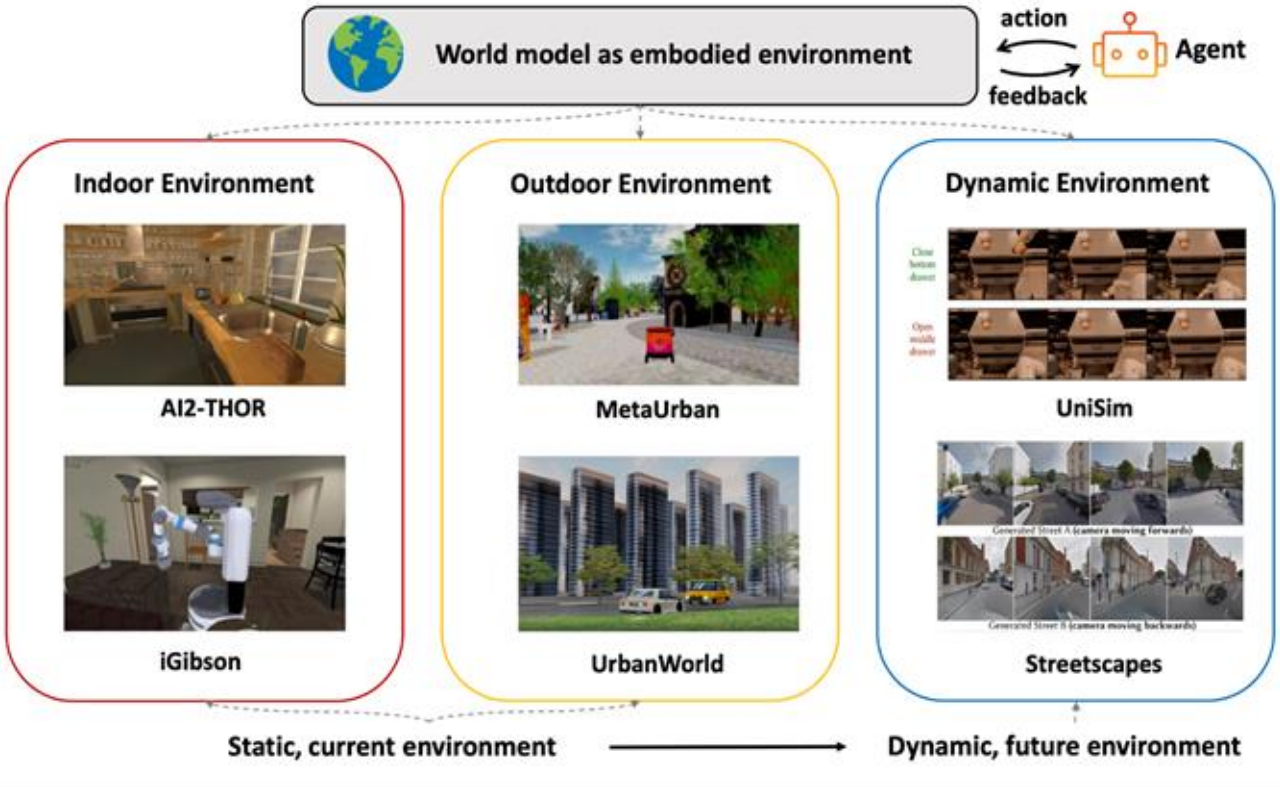
Beyond Video: Embodied World Models

Explain:

- interactive environments
- 3D simulation
- robotics / agents

Examples:

- Habitat
- VirtualHome



Examples - Dji ROMO clean robot

Part 2 Conclusions

Shift from:

Passive Generation

trending_flat

Interactive Simulation

Closer to real-world decision making

Part 3: APPLICATIONS

Applications Overview

Four domains:

1. Games
2. Robotics
3. Autonomous driving
4. Social simulation

Example Applications

- Autonomous driving → prediction
- Robotics → planning
- Games → simulation

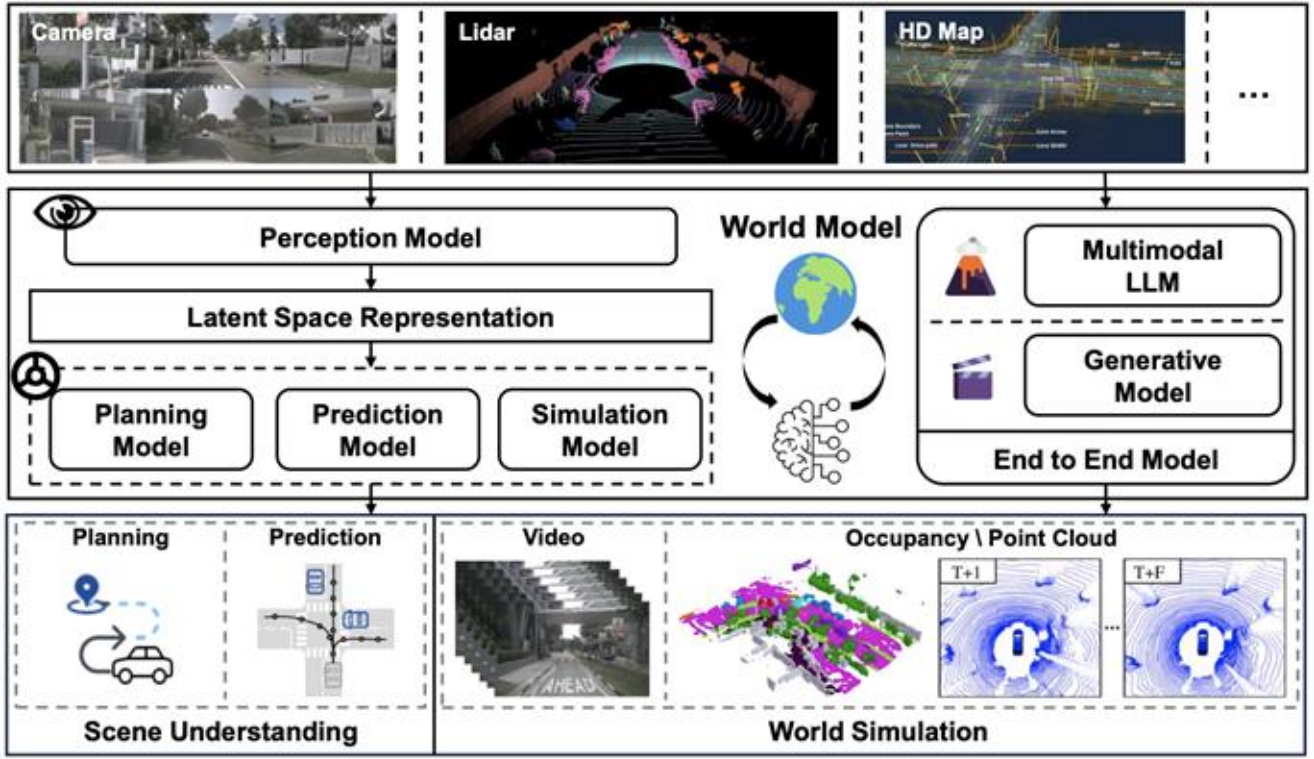


Fig. 6. Application of world model in autonomous driving.

Remaining Problems & Future Directions

Key Challenges

- long-term consistency
- causal reasoning
- data efficiency
- evaluation

Future Directions

- unified world model
- multimodal integration
- interactive agents
- AGI

*What's missing for AGI?

World Models for Physical AI

Wentao Zhou

Two real-world examples

- Video Generation with Cosmos
- One model for both world representation and action generation with Lingbot

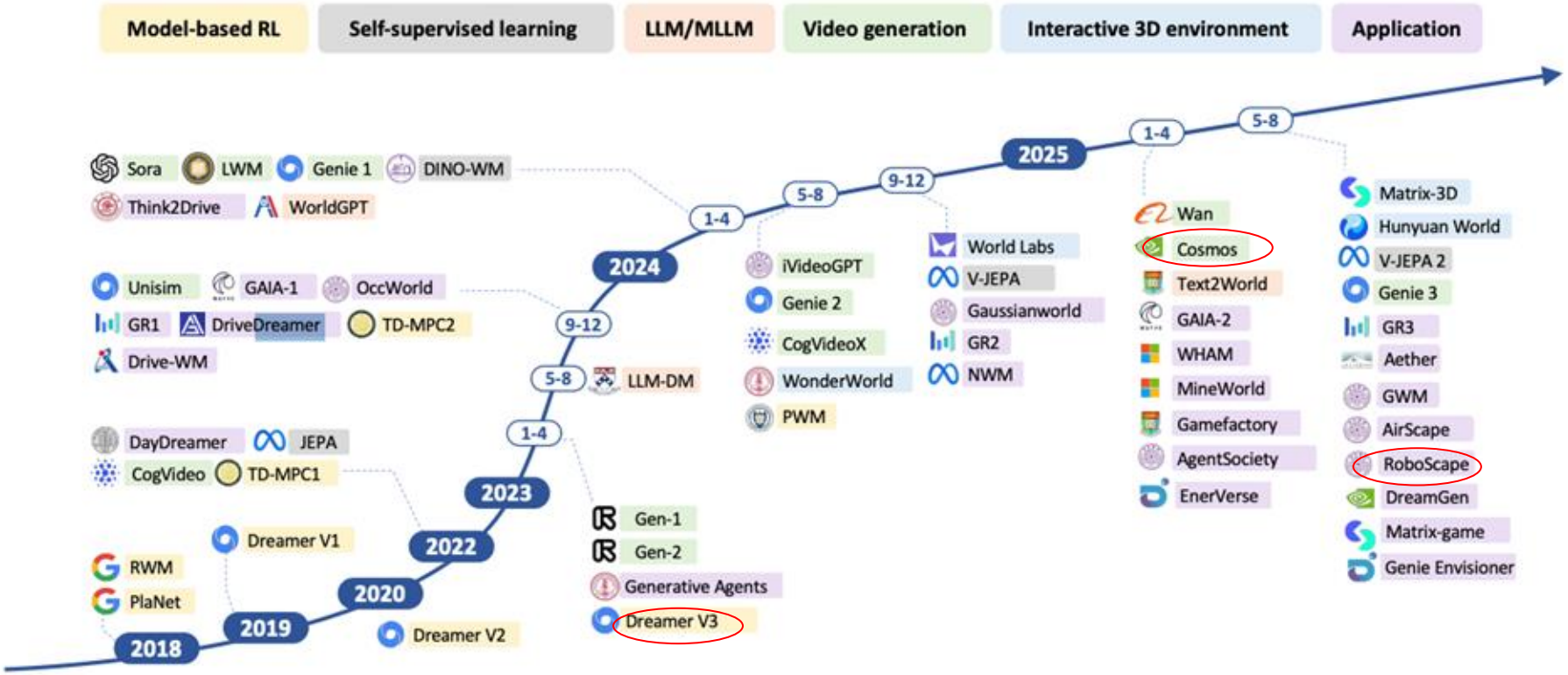


Fig. 1. The roadmap of world models in deep learning era.

Cosmos World Foundation Model Platform for Physical AI

NVIDIA[1]

Abstract

Physical AI needs to be trained digitally first. It needs a digital twin of itself, the policy model, and a digital twin of the world, the world model. In this paper, we present the Cosmos World Foundation Model Platform to help developers build customized world models for their Physical AI setups. We position a world foundation model as a general-purpose world model that can be fine-tuned into customized world models for downstream applications. Our platform covers a video curation pipeline, pre-trained world foundation models, examples of post-training of pre-trained world foundation models, and video tokenizers. To help Physical AI builders solve the most critical problems of our society, we make Cosmos open-source and our models open-weight with permissive licenses available via NVIDIA Cosmos-Predict1.

What does Cosmos do?

- Cosmos generates videos, especially for robotics tasks
- Videos by Cosmos are more physically correct.
- Cosmos accept input from different modalities, like camera control

Traditional Way of Training Robots: Expensive

Reactive Policy:

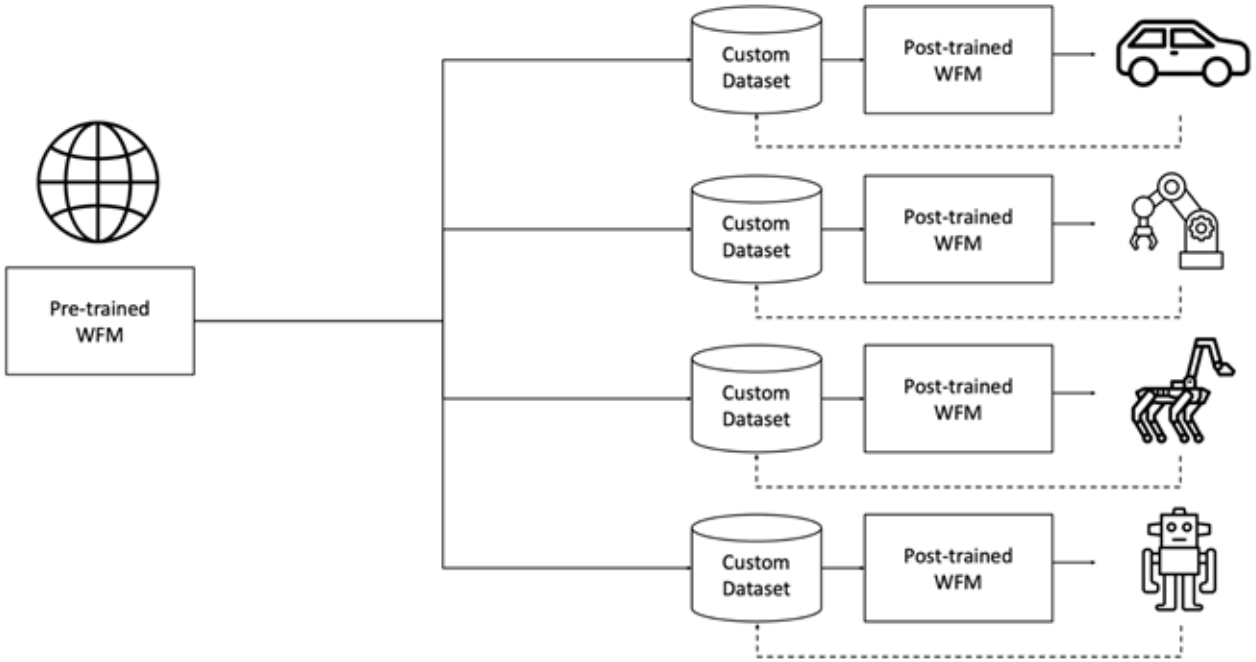
- Observe -> Act: Need image-action data pairs
- Expensive real robot data

Motivation: Using Cosmo as Simulate to Generate Data?

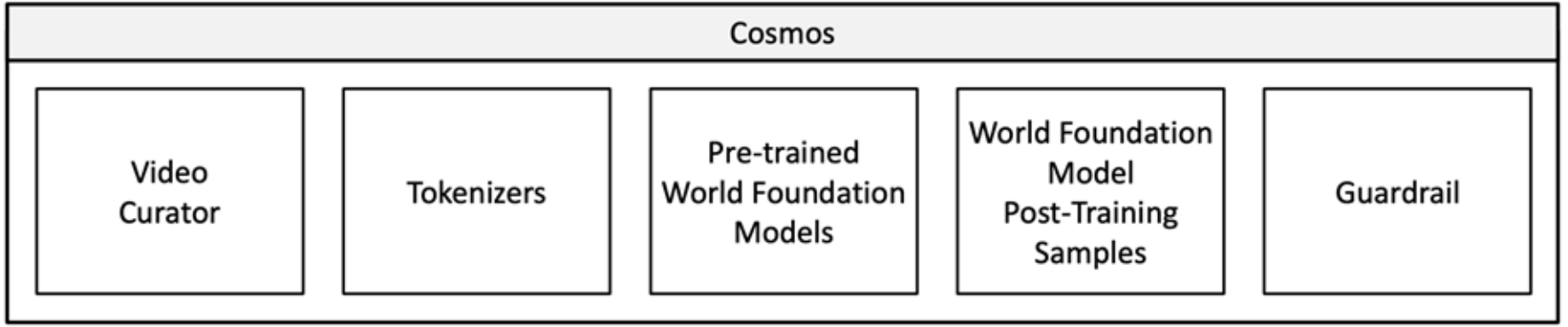
World Model for Action Policy

- Cheaper generated data
- Adaptive to more modalities

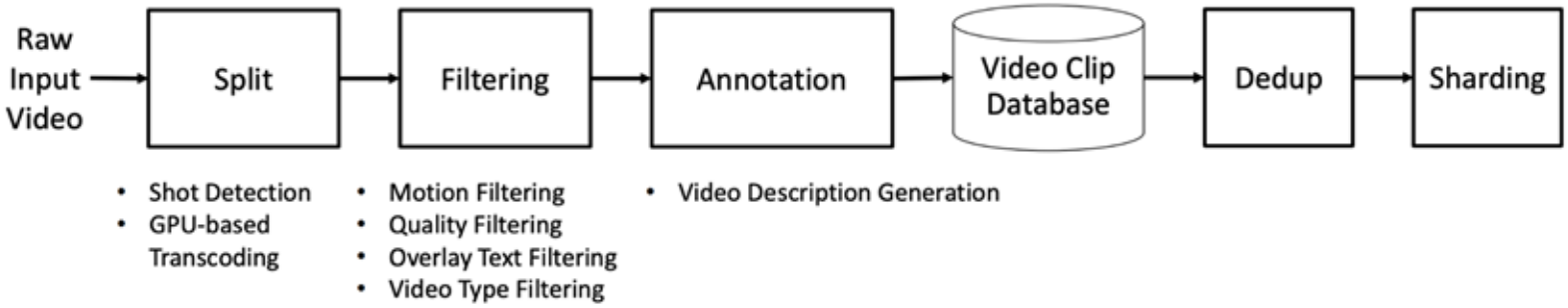
Cosmo Platform can Generate Data for Multi-tasks

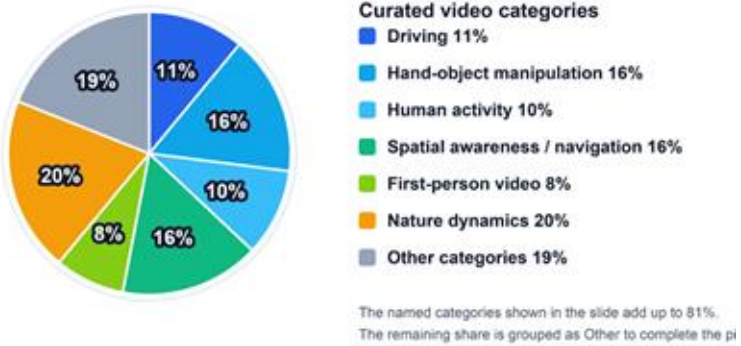


Components of Cosmo

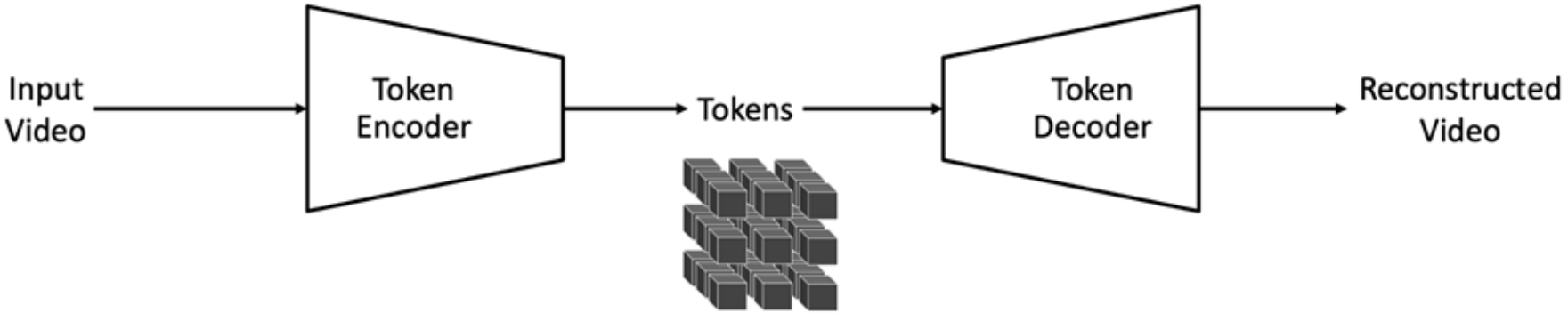


Step 1: Video Curation





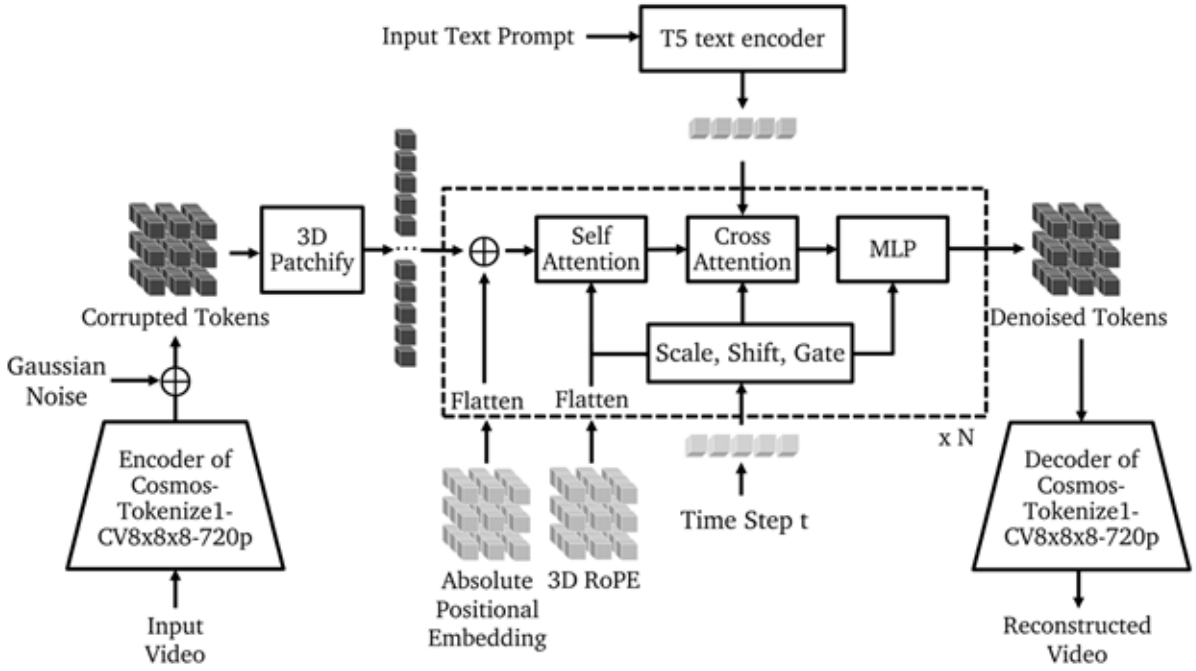
Step 2: Tokenizer



- Compact videos into tokens
- Tokenizer is also causal

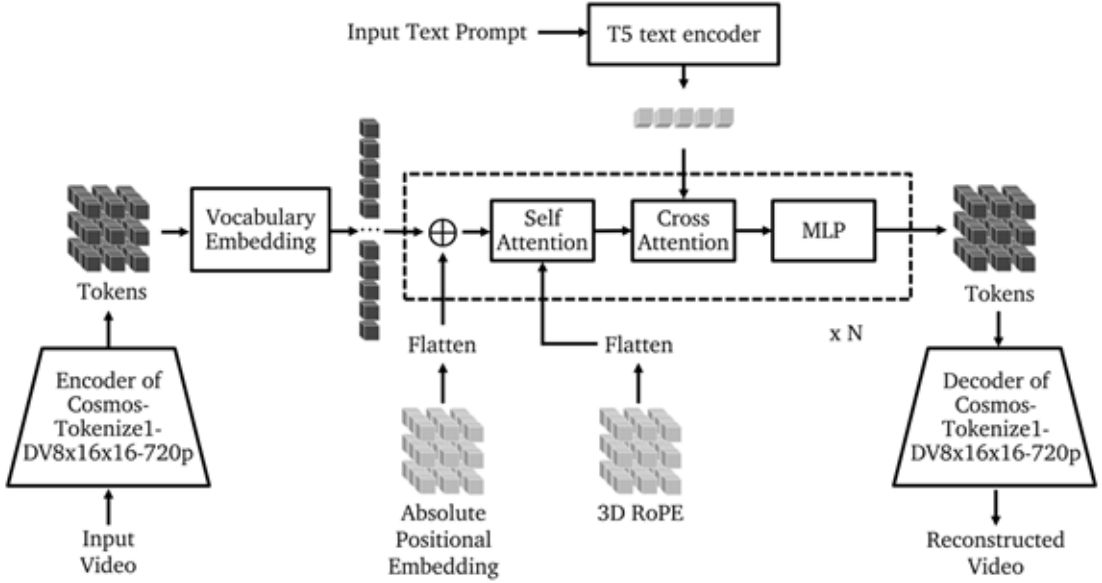
Step 3: Pretraining - Diffusion WFM

- Stage 1: Text2World
  - Generate videos from text prompts
- Stage 2: Video2World
  - Predict future videos from current video + text condition



Step 3: Pretraining - Autoregressive WFM

- Stage 1: Video Prediction
  - Predict future videos *only* from current video
- Stage 2: Text Incorporation
  - Add cross-attention layers for text embeddings



Choices for World Foundation Models (WFM):

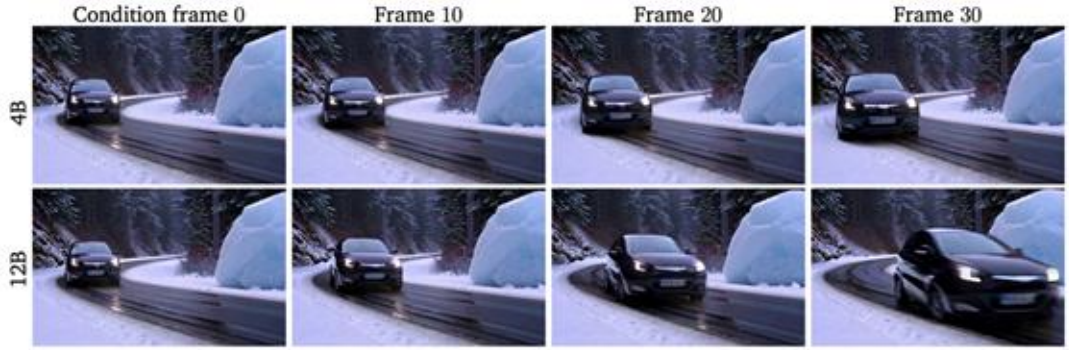
Diffusion WFM

- Continuous tokens
- Better visual quality
- Easier post-training with control

Autoregressive WFM

- Discrete tokens
- Better sequential reasoning
- Easier LLM inference optimization

Effect of Scaling-Up

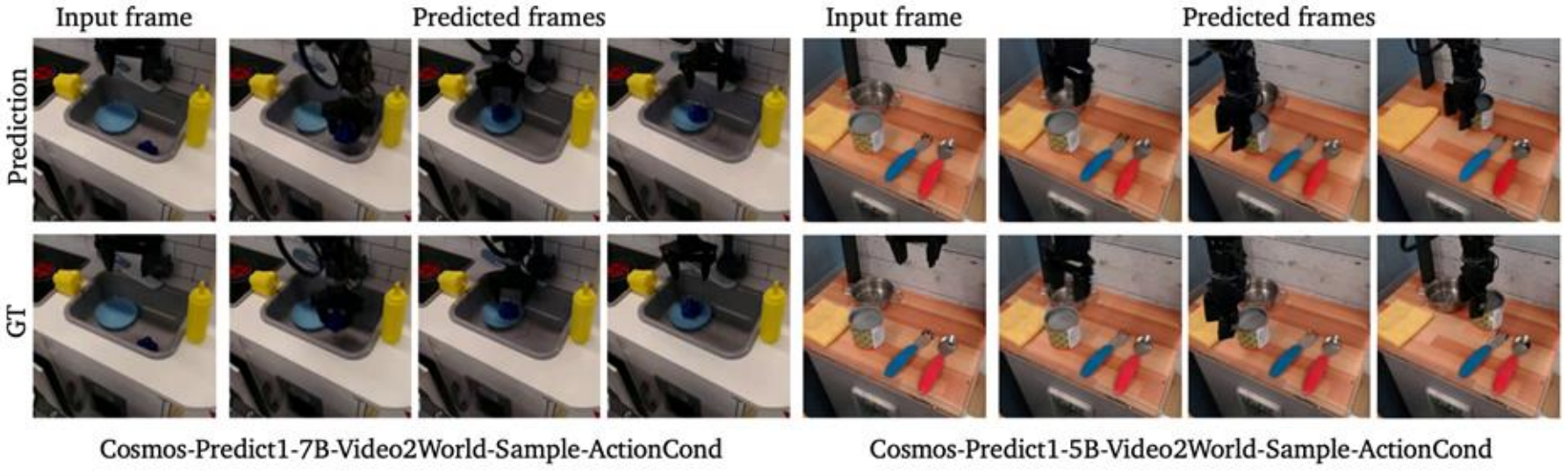


Prompt: None.

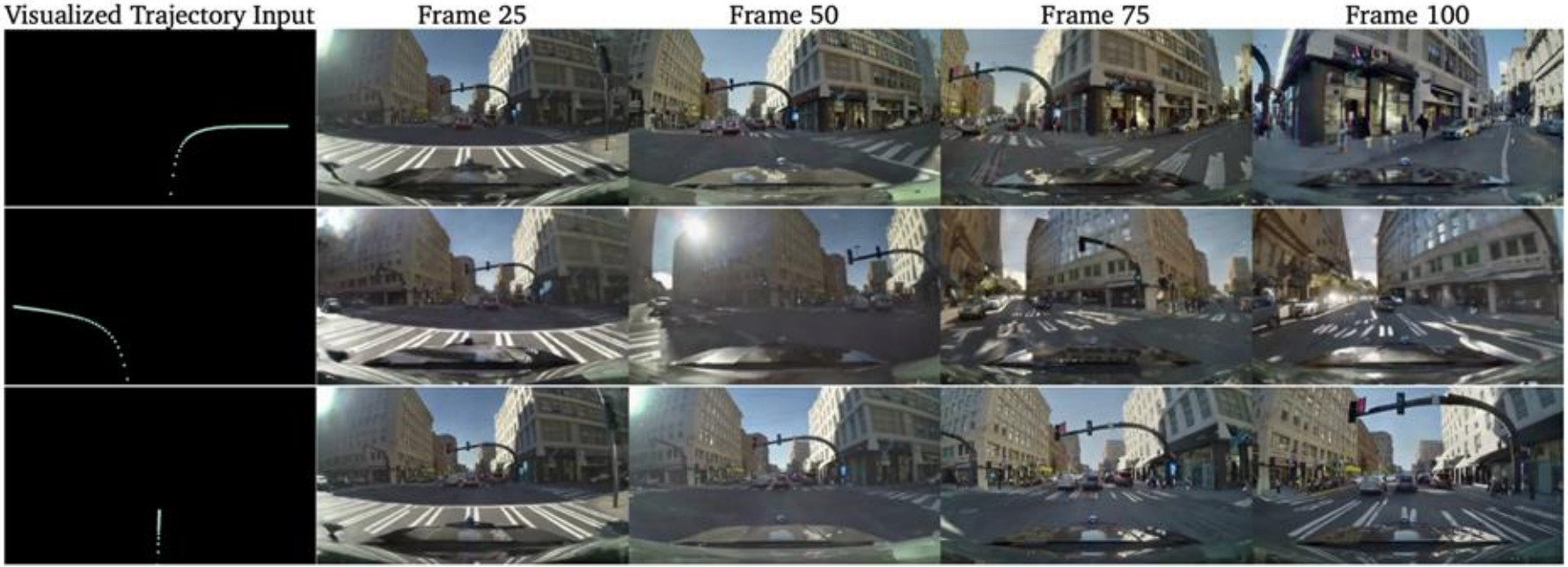
Prompt: The video of a car moving forward, passing under a large overpass. The road is clear, and there are a few other cars visible in the distance. The weather appears to be sunny, and the time of day is daytime. The scene is set on a busy highway with concrete structures and greenery on the sides.

Posttraining - Camera Control

Posttraining - Action Conditioning



Posttraining - Trajectory Conditioning



Causal World Modeling for Robot Control

Lin Li* Qihang Zhang*† Yiming Luo* Shuai Yang Ruilin Wang Fei Han
Mingrui Yu Zelin Gao Nan Xue Xing Zhu Yujun Shen Yinghao Xu‡

*Equal Contribution †Project Lead ‡Corresponding Author

This work highlights that video world modeling, alongside vision-language pre-training, establishes a fresh and independent foundation for robot learning. Intuitively, video world models provide the ability to "imagine" the near future by understanding the causality between actions and visual dynamics. Inspired by this, we introduce LingBot-VA, an autoregressive diffusion framework that learns frame prediction and policy execution simultaneously. Our model features three carefully crafted designs: (1) ***a shared latent space***, integrating vision and action tokens, driven by a Mixture-of-Transformers (MoT) architecture, (2) ***a closed-loop rollout mechanism***, allowing for ongoing acquisition of environmental feedback with ground-truth observations, (3) ***an asynchronous inference pipeline***, parallelizing action prediction and motor execution to support efficient control. We evaluate our model on both simulation benchmarks and real-world scenarios, where it shows significant promise in long-horizon manipulation, data efficiency in post-training, and strong generalizability to novel configurations. The code and model are made publicly available to facilitate the community.

Website: https://technology.robbyant.com/lingbot-va
Github: https://github.com/robbyant/lingbot-va
Checkpoints: https://huggingface.co/robbyant/lingbot-va

Robbyant

Backgrounds

What is VLA:

Takes in visual observations and natural language instructions to make executable robotic control commands.

What is embodied Agents:

AI systems, such as robots or virtual avatars, that perceive and interact with a physical or simulated environment to accomplish complex tasks.

Physical Agent with World Model

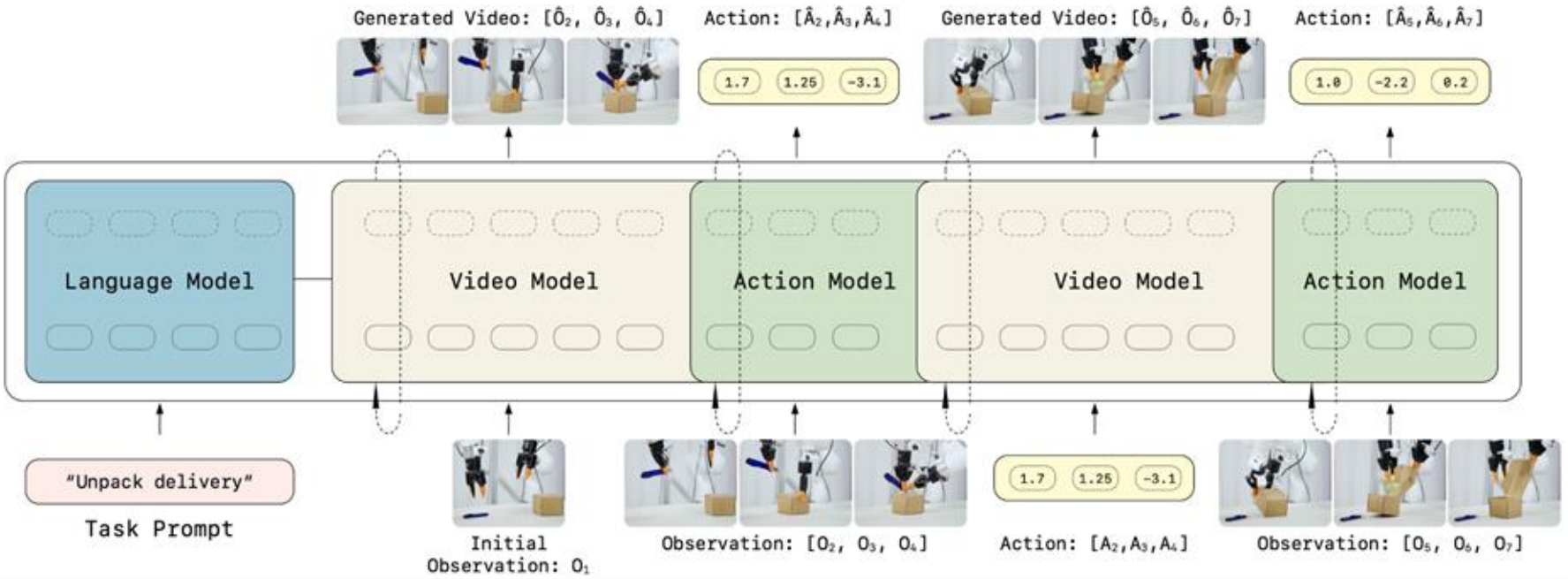
Existing Vision-Language Action policies for robot

- Directly map obeservation to action sequences
- Single neural network

World model for robot control

- Unified video + action prediction with dense supervision
- Disentangled architecture
- Utilizing real-world knowledge in pretrained video generation models

LingBot


Generated Video: [Ô₂, Ô₃, Ô₄]
Action: [Â₂,Â₃,Â₄]
Generated Video: [Ô₅, Ô₆, Ô₇]
Action: [Â₅,Â₆,Â₇]
1.7
1.25
-3.1
1.0
-2.2
0.2
Language Model
Video Model
Action Model
Video Model
Action Model
"Unpack delivery"
Task Prompt
Initial Observation: O₁
Observation: [O₂, O₃, O₄]
1.7
1.25
-3.1
Action: [A₂,A₃,A₄]
Observation: [O₅, O₆, O₇]

Inference

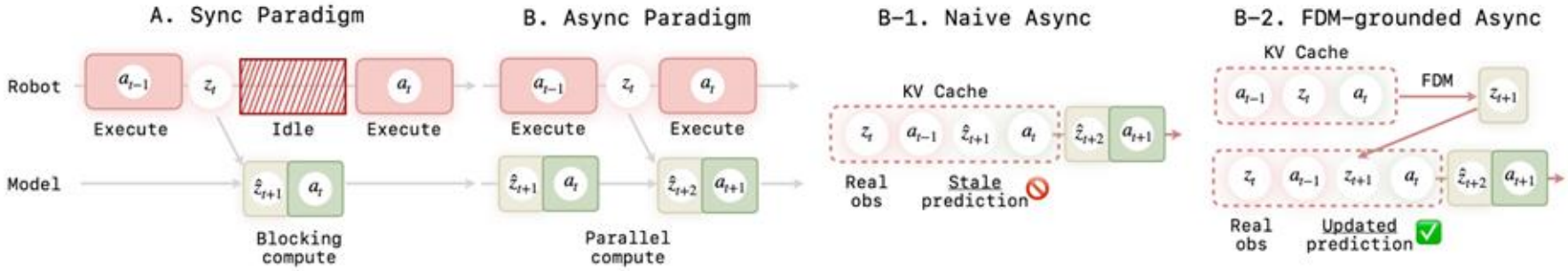
- Given past and current images, predict next image
- Conditioned on current and next image, predict current action
- KV-cache for past images

$$\text{(Stage 1) Visual dynamics prediction:} \quad o_{t+1} \sim p_\theta(\cdot \mid o_{\leq t}),$$

$$\text{(Stage 2) Inverse dynamics:} \quad a_t \sim g_\psi(\cdot \mid o_t, o_{t+1}).$$

Deployment

- A: delay by blocked computation
- B-1: observation not updated
- B-2: imagine grounded observation for future prediciton



Data

Six sources of embodied learning datasets

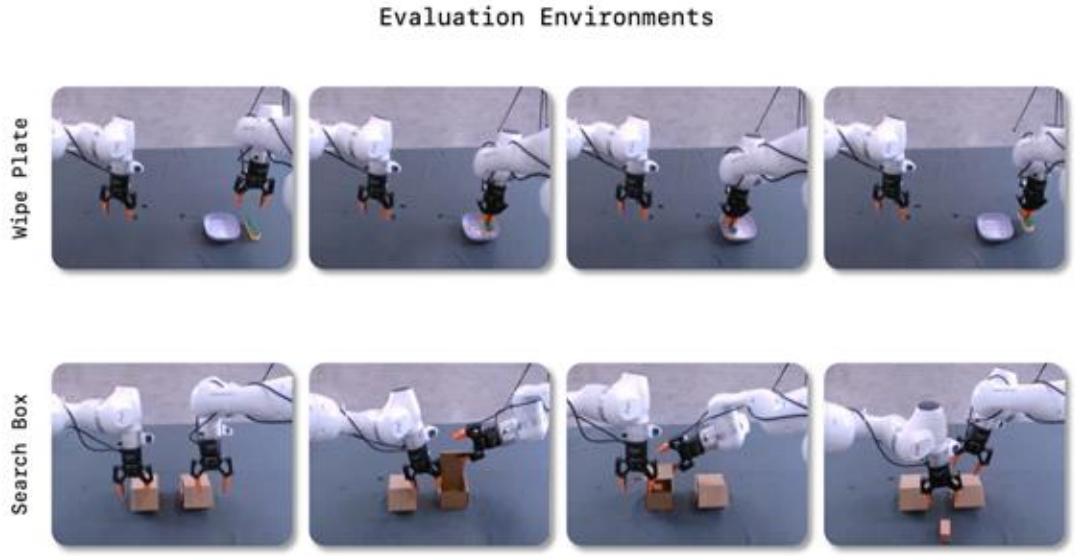
- Manipulation
- Simulation
- Human demonstration
- Bimanual

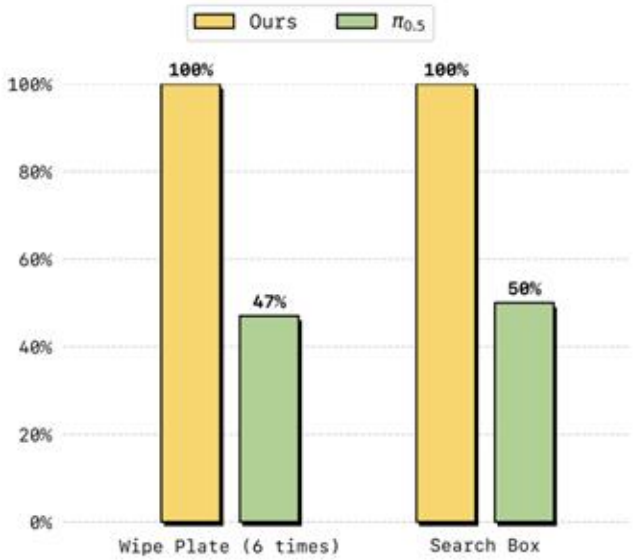
Model Details

- Video model: Pretrained Wan2.2 with causal VAE
- Action encoder & decoder: Single-layer MLP
- Text Encoder: T5

Evaluation - Real-World Tasks

Evaluation - Temporal Memory





Takeaways

- Cosmos makes a reusable, scalable generative platform.
- LingBot integrates a world model inside a robot’s mind

Future directions:

1. Faster inference for robot-related tasks
2. Generalibility for more embodied frameworks
3. More stable physical fidelity in contact-rich settings

Mastering Diverse Domains through World Models

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, Timothy Lillicrap

Article | Open access | Published: 02 April 2025

Mastering diverse control tasks through world models

Danijar Hafner ✉, Jurgis Pasukonis, Jimmy Ba & Timothy Lillicrap

Nature **640**, 647–653 (2025) | Cite this article

Abstract

Developing a general algorithm that learns to solve tasks across a wide range of applications has been a fundamental challenge in artificial intelligence. Although current reinforcement-learning algorithms can be readily applied to tasks similar to what they have been developed for, configuring them for new application domains requires substantial human expertise and experimentation[1,2]. Here we present the third generation of Dreamer, a general algorithm that outperforms specialized methods across over 150 diverse tasks, with a single configuration. Dreamer learns a model of the environment and improves its behaviour by imagining future scenarios. Robustness techniques based on normalization, balancing and transformations enable stable learning across domains. Applied out of the box, Dreamer is, to our knowledge, the first algorithm to collect diamonds in *Minecraft* from scratch without human data or curricula. This achievement has been posed as a substantial challenge in artificial intelligence that requires exploring farsighted strategies from pixels and sparse rewards in an open world[3]. Our work allows solving challenging control problems without extensive experimentation, making reinforcement learning broadly applicable.

Roadmap

- Review basic of world models
- DREAMER V3

What is world model

A world model is a learned internal representation that simulates the dynamics of the real world

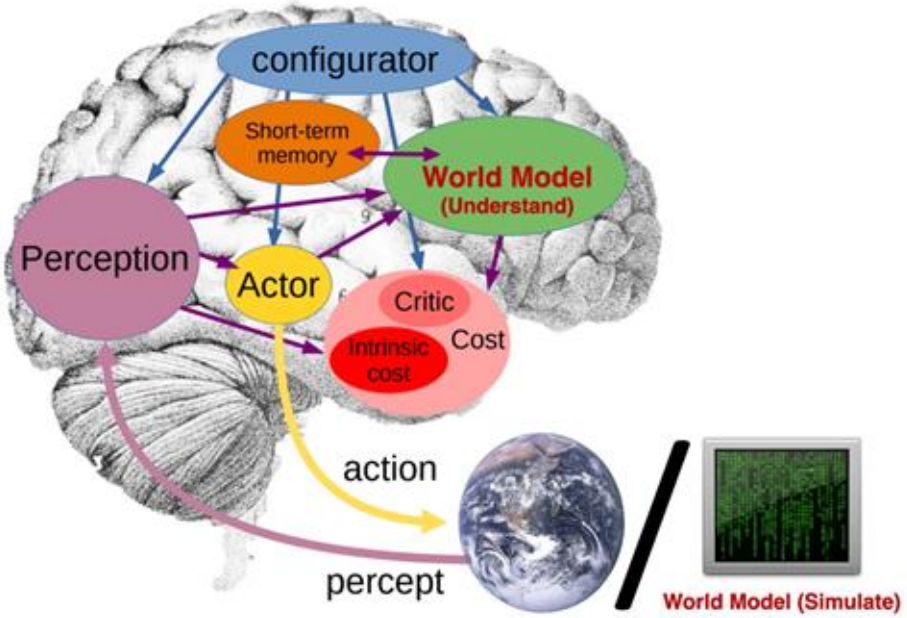
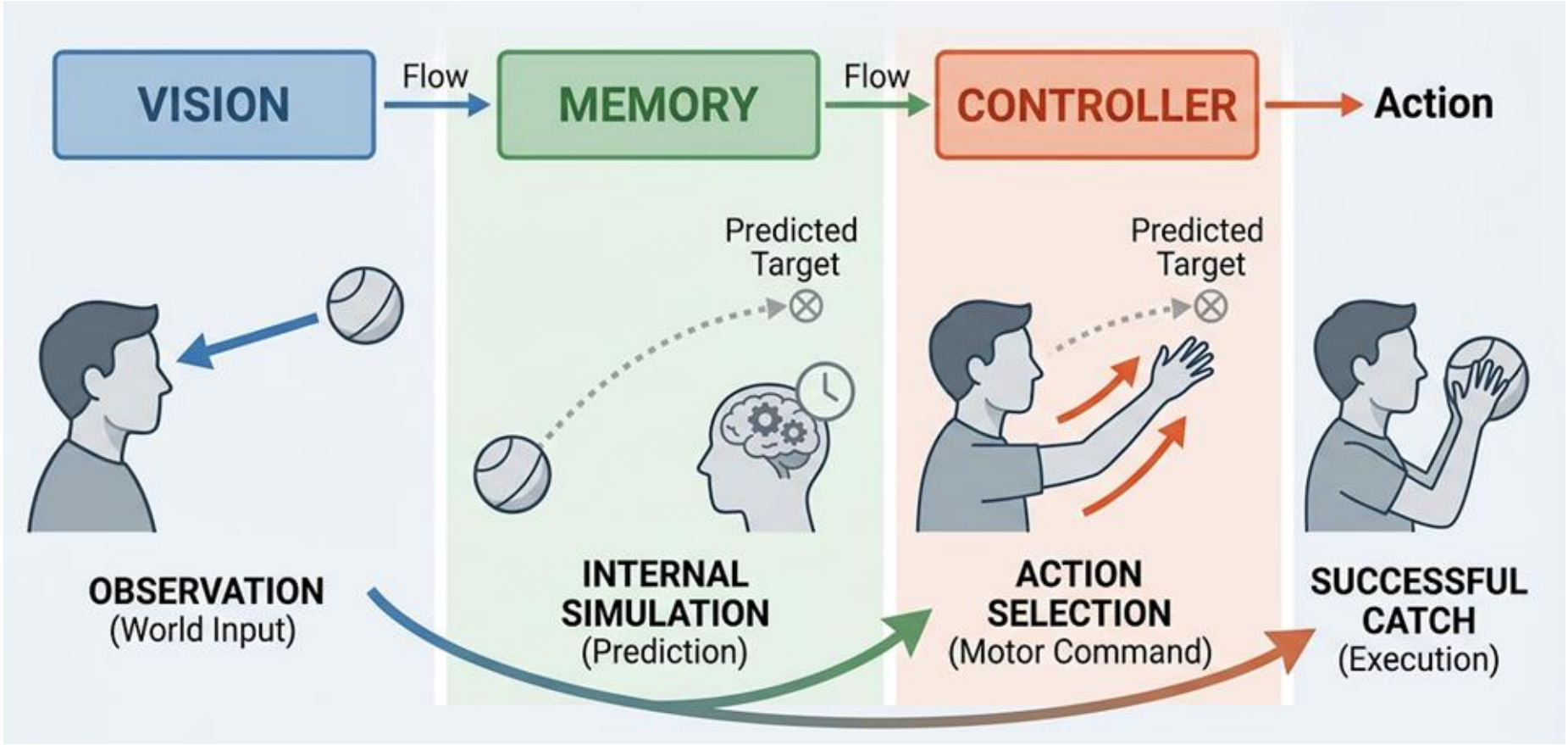


Figure 1: An internal "world model" in the human brain that predicts the coarse future, versus an external "world model" that aims to simulate every detail of the reality. Figure from Yann LeCun's slides.

Basic Process of Using World Model



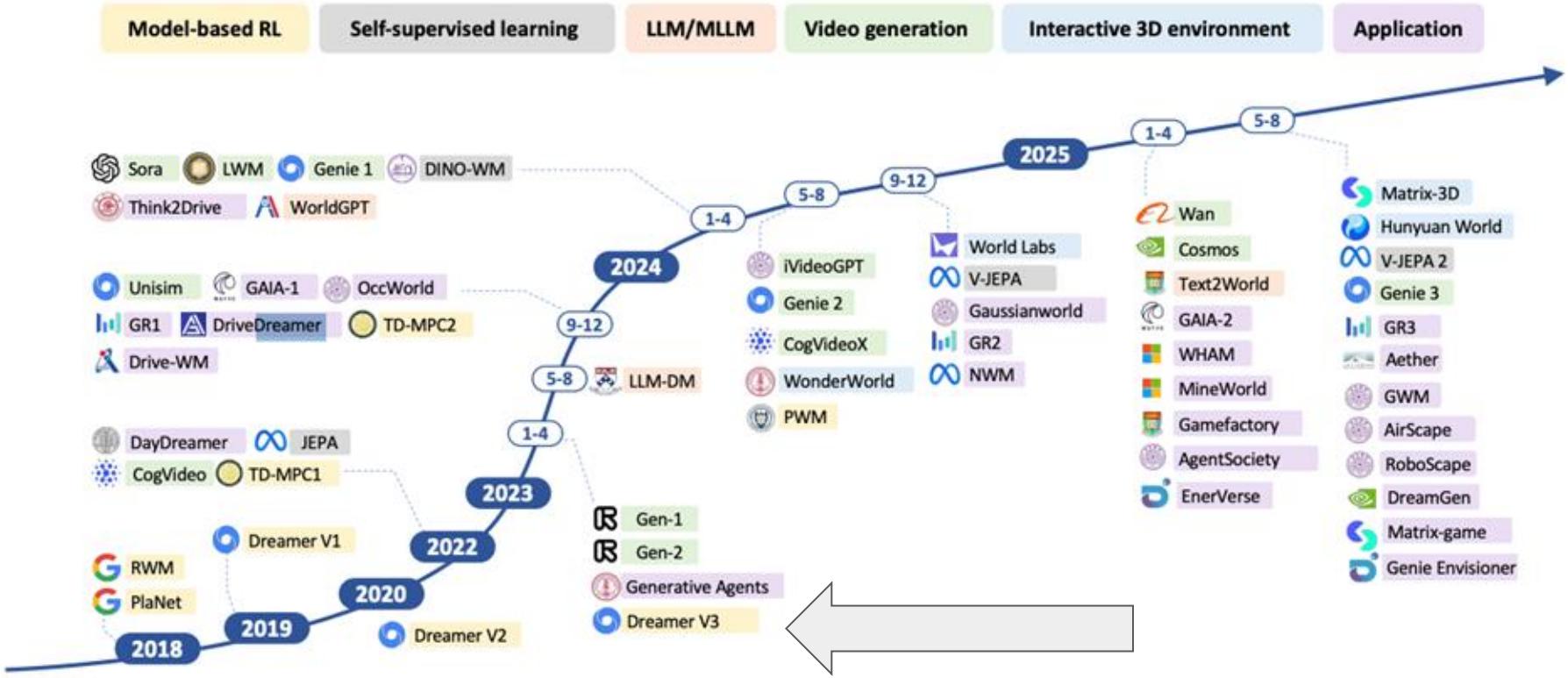


Fig. 1. The roadmap of world models in deep learning era.

World Models

Generative / Simulative World Models

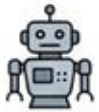
- simulate realistic worlds

- generate videos or environments

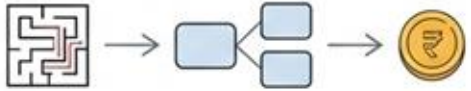
- examples: Sora, Genie 2, Genie 3

Decision / Control-Oriented World Models

- model environment dynamics for agents



- predict future states and rewards



- support planning and action ⟶ plan
- examples: Dreamer, DreamerV3, IRIS, TWM

Genrative world models focus on world simulation.
Control-oriented word models focus on helping agents act.

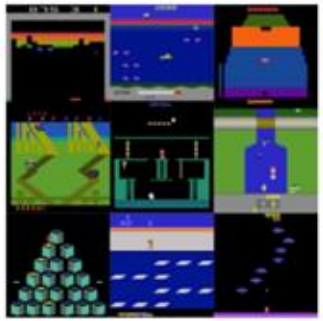
Dreamer V3: Learning to Act by Imagining the World

World Model

- **Input:** observations such as images or video, together with actions
- **Core idea:** predict future states, observations, rewards, or outcomes
- **Focus:** understanding how the environment changes over time
- **Output** Output example: future state, next frame, predicted reward
- **Supports:** planning and decision-making

General Algorithm: 150 tasks

(a) Control Suite



(b) Atari

(c) ProcGen

(d) DMLab

(e) Minecraft

A **single general algorithm** with fixed hyperparameters that beats domain-specific expert algorithms across 8 benchmarks (150+ tasks) all with fixed hyperparameters

DREAMER works well across many visual domains

Benchmark

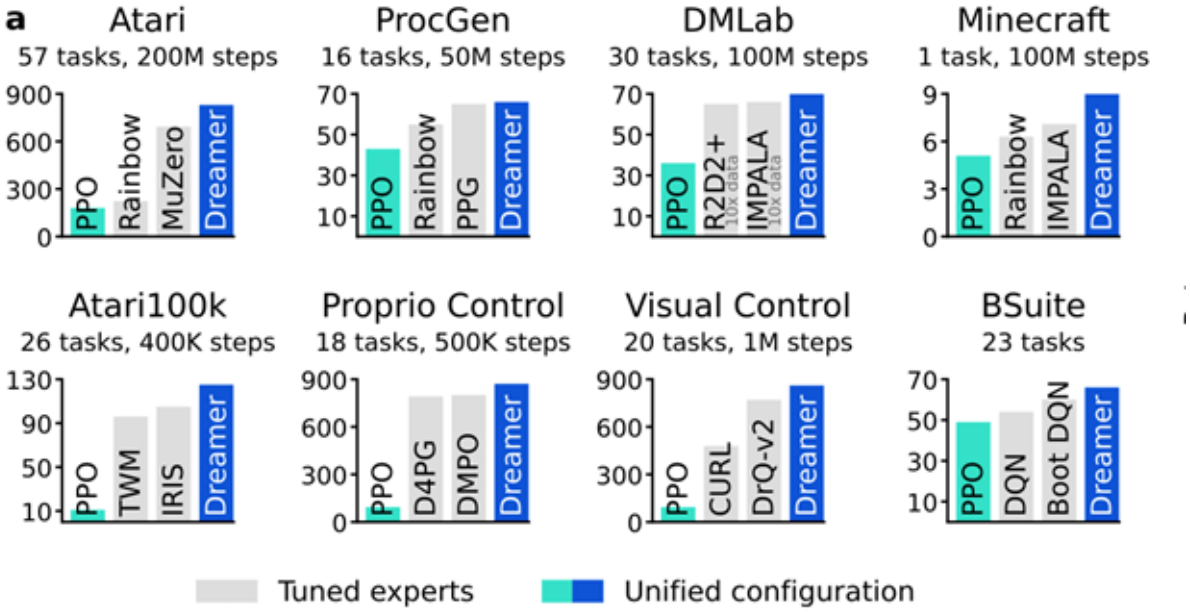
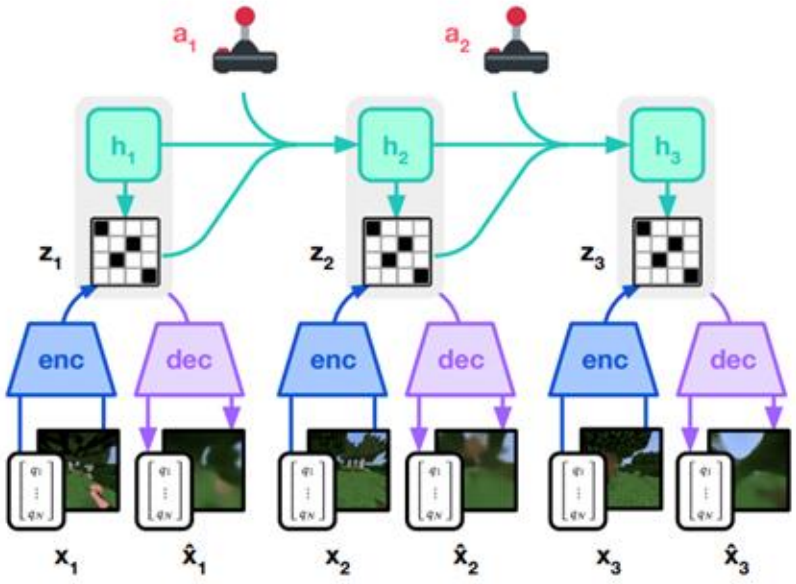


Figure 1: Benchmark summary. **a**, Using fixed hyperparameters acro
outperforms tuned expert algorithms across a wide range of benchmarks a

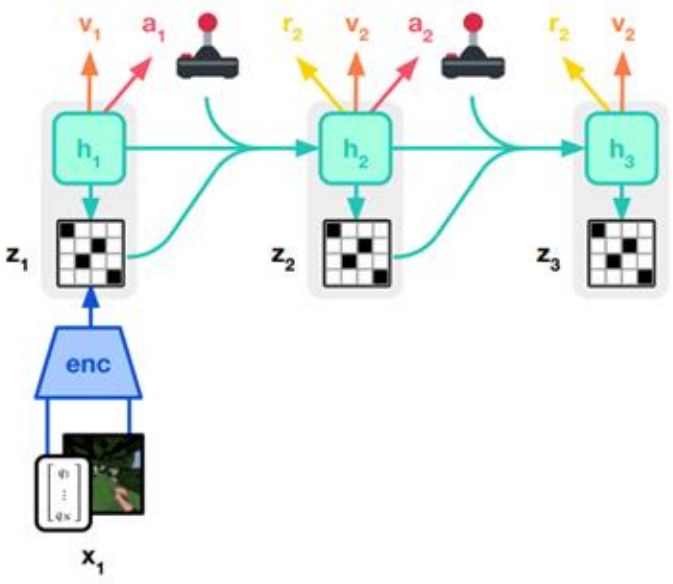
DREAMER is much more capable

What is DREAMER3?

- **The world model** predicts the outcomes of potential actions,
- **a critic neural network** judges the value of each outcome,
- **an actor neural network** chooses actions to reach the best outcomes.

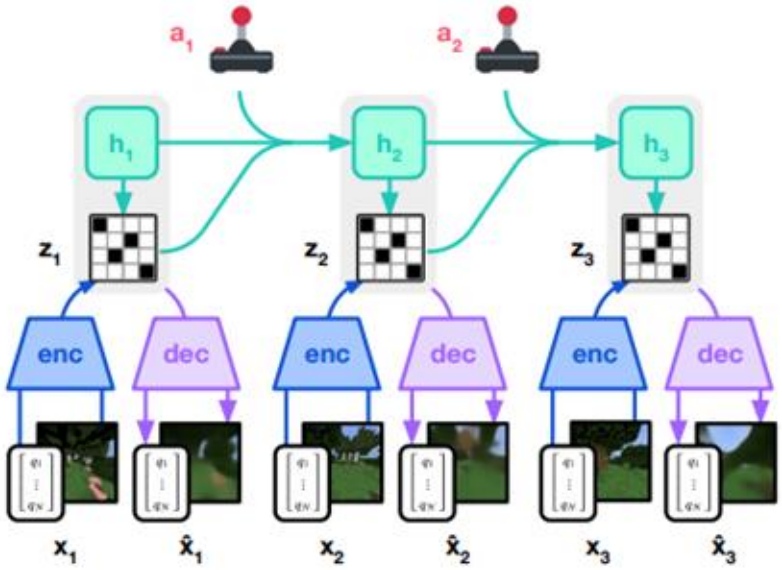


(a) World Model Learning



(b) Actor Critic Learning

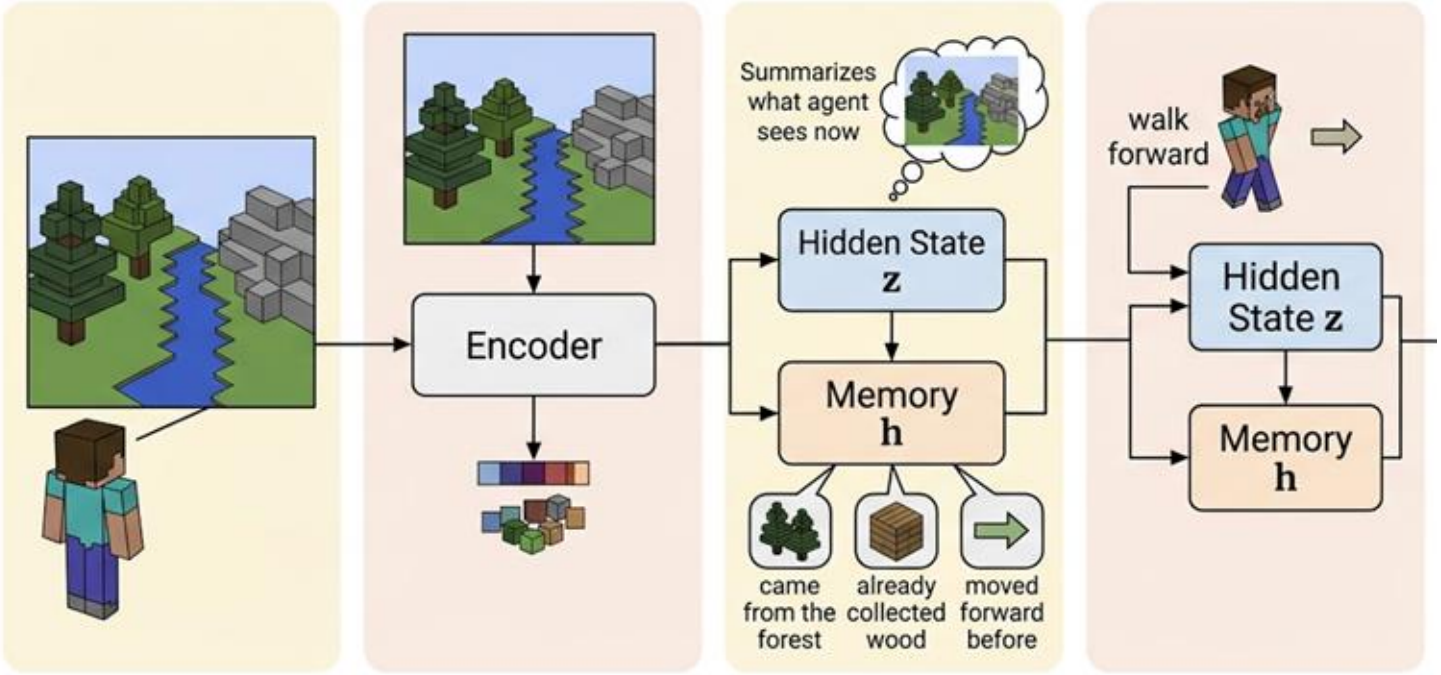
World model learning: learns how the world works from experience



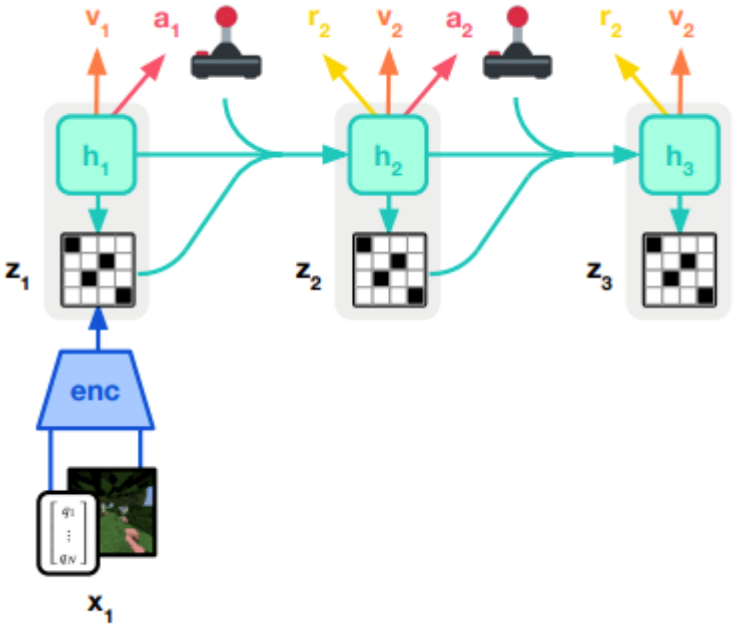
(a) World Model Learning

- x_t = what the agent sees
- z_t = compact representation of the current state
- h_t = memory from previous time steps
- a_t = action taken by the agent
- $\hat{x}_t$ = reconstructed observation from the model

How the world model works in Minecraft?



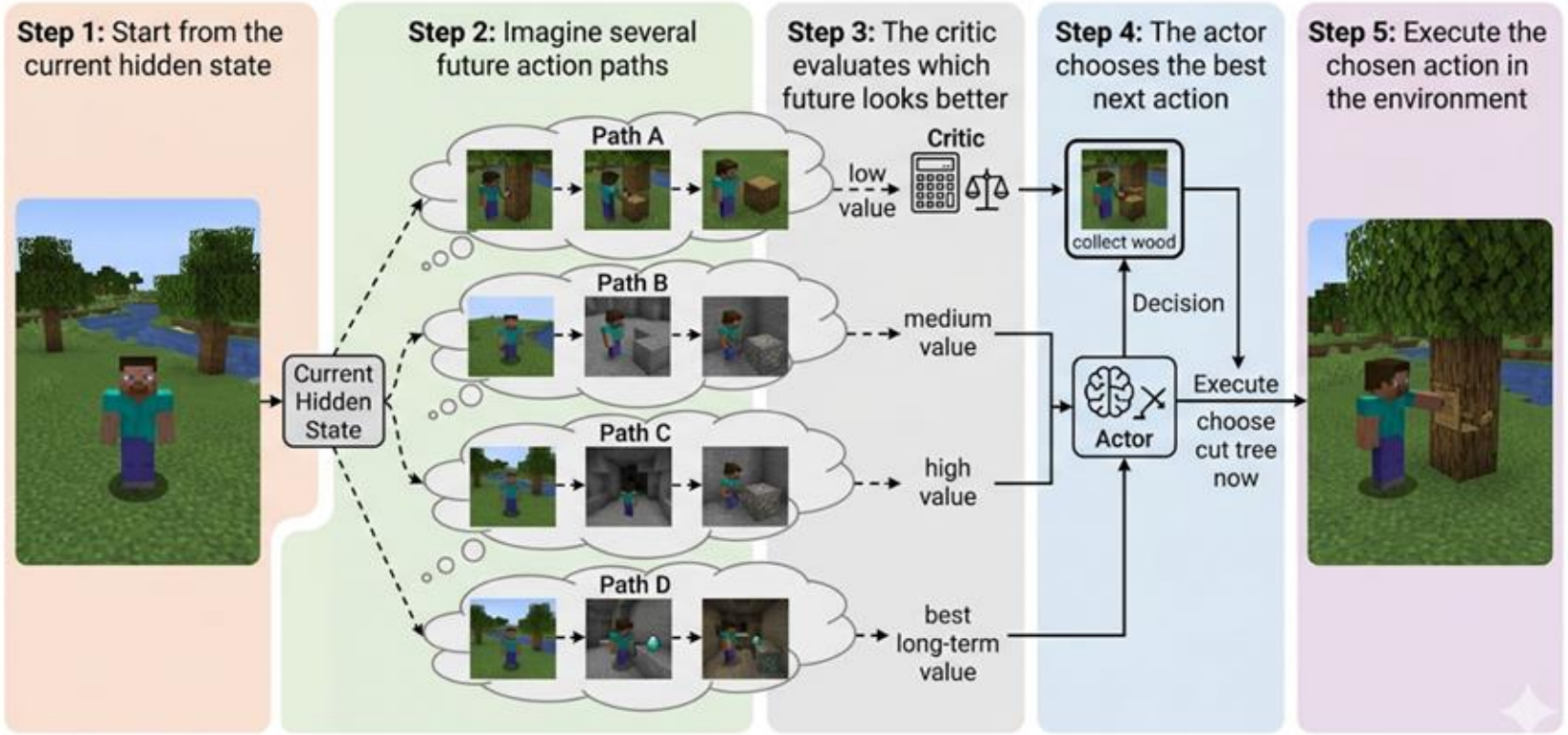
Actor Critic Learning: dream the future

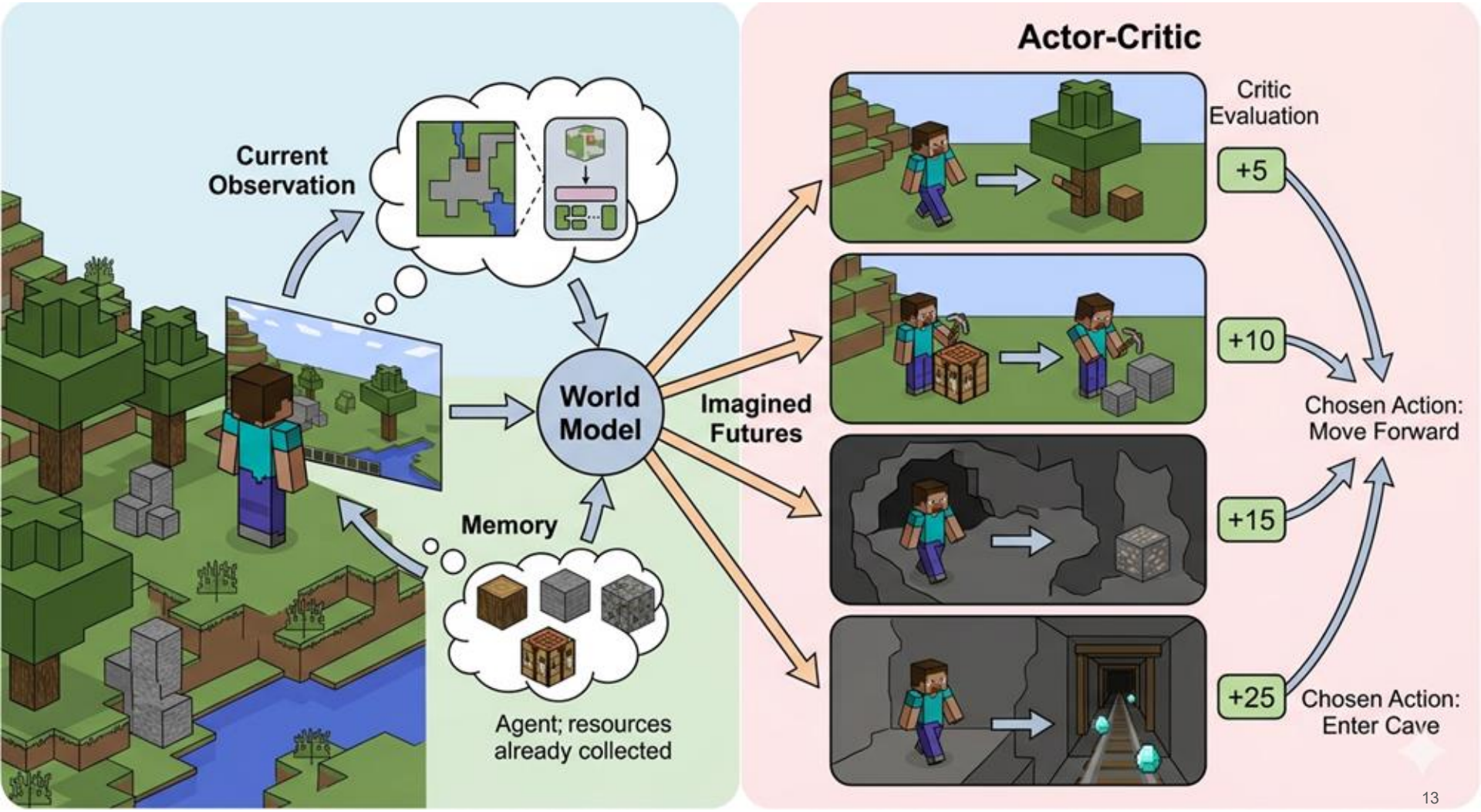


(b) Actor Critic Learning

- x_1: initial observation
- **enc**: encoder
- z_t: latent state
- h_t: recurrent hidden state / memory
- a_t: action
- r_t: reward
- v_t: value estimate

How the Critic- actor Learning works in Mincraft?




Actor-Critic
Current Observation
World Model
Imagined Futures
Memory
Agent; resources already collected
Critic Evaluation
+5
+10
+15
+25
Chosen Action: Move Forward
Chosen Action: Enter Cave

Algorithm Overview

```
for step in range(total_steps):
    latent = world_model.observe(image, latent, action)
    action = actor_critic.act(latent)
    image, reward, done = env.step(action)
    replay_buffer.add(action, image, reward, done)
    world_model.train(replay_buffer)
    actor_critic.train(world_model)
```

Application: Minecraft

Mission: Collect diamonds in Minecraft from scratch

A long-standing challenge in AI because of:

- **Sparse rewards**
- **Difficult exploration**
- **Long time horizons**
- **High procedural diversity in an open-world environment**

Application: Minecraft

- First algorithm to collect diamonds in Minecraft from scratch without using human data
- Discover diamonds in 100M environment steps.

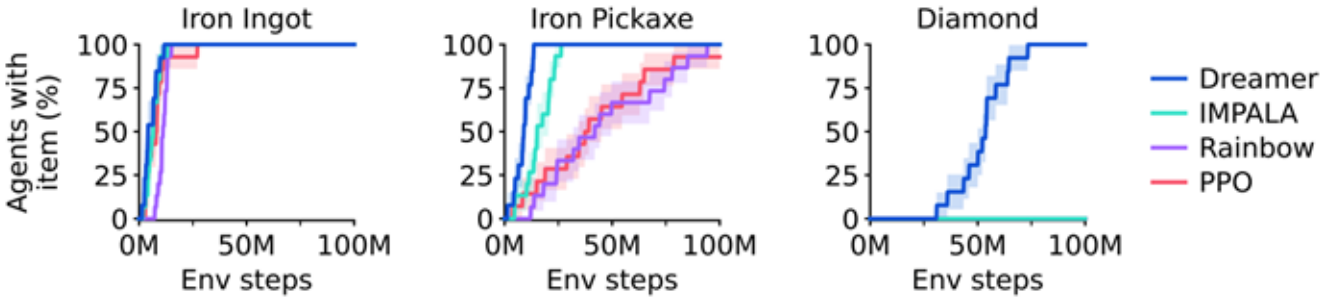
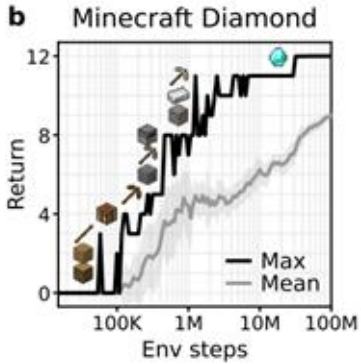
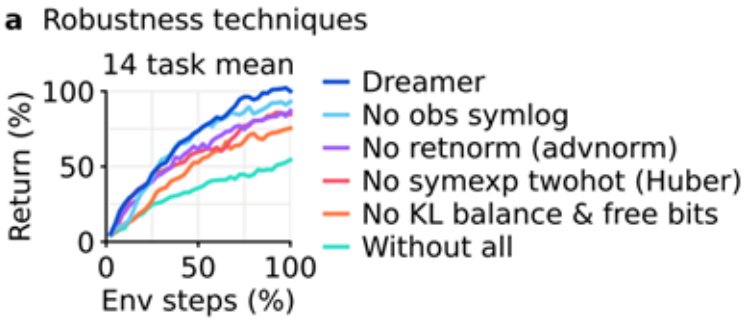


Figure 5: Fraction of trained agents that discover each of the three latest items in the Minecraft Diamond task. Although previous algorithms progress up to the iron pickaxe, Dreamer is the only compared algorithm that manages to discover a diamond, and does so reliably.

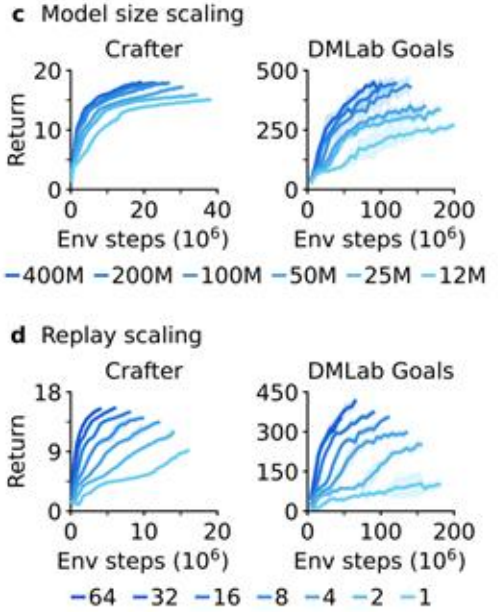


Robustness techniques



- Dreamer performs best when all robustness techniques are used together
- Removing any single technique leads to a noticeable drop in performance
- This suggests Dreamer V3's stability comes from the combination of multiple design choices

Scaling



- Performance improves as the model size increases
- More gradient updates per interaction also reduce the data needed to learn
- This suggests Dreamer V3 scales robustly with both model size and replay ratio

Thank you!