Genie: Generative Interactive Environments

From: Google Deepmind

Presented by: Sathvik Reddy

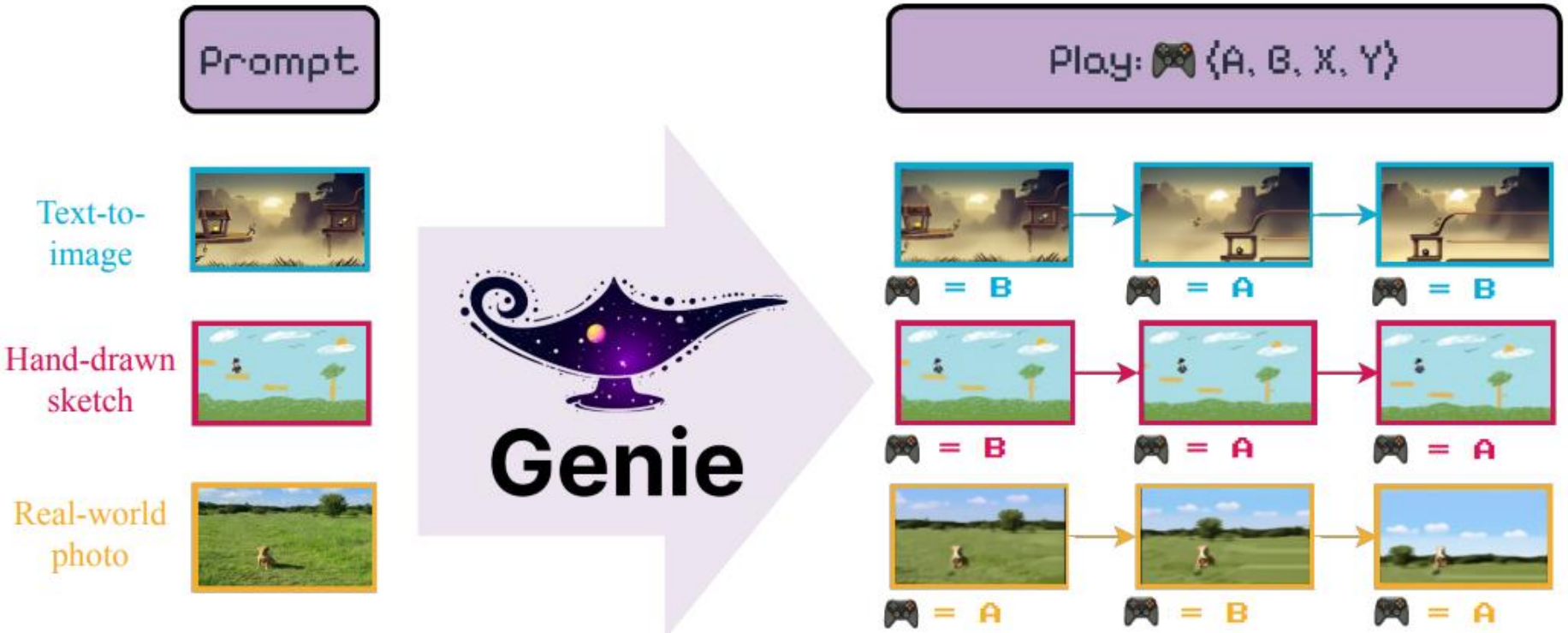
Levels of AGI

Morris et al. 2024

Access to a simulator for the task

Performance (rows) x **Generality** (columns)	**Narrow** *clearly scoped task or set of tasks*	**General** *wide range of non-physical tasks, including metacognitive tasks like learning new skills*
Level 0: No AI	**Narrow Non-AI** calculator software; compiler	**General Non-AI** human-in-the-loop computing, e.g., Amazon Mechanical Turk
Level 1: Emerging *equal to or somewhat better than an unskilled human*	**Emerging Narrow AI** GOFAI (Boden, 2014); simple rule-based systems, e.g., SHRDLU (Winograd, 1971)	**Emerging AGI** ChatGPT (OpenAI, 2023), Bard (Anil et al., 2023), Llama 2 (Touvron et al., 2023), Gemini (Pichai & Hassabis, 2023)
Level 2: Competent *at least 50th percentile of skilled adults*	**Competent Narrow AI** toxicity detectors such as Jigsaw (Das et al., 2022); Smart Speakers such as Siri (Apple), Alexa (Amazon), or Google Assistant (Google); VQA systems such as PaLI (Chen et al., 2023); Watson (IBM); SOTA LLMs for a subset of tasks (e.g., short essay writing, simple coding)	**Competent AGI** not yet achieved
Level 3: Expert *at least 90th percentile of skilled adults*	**Expert Narrow AI** spelling & grammar checkers such as Grammarly (Grammarly, 2023); generative image models such as Imagen (Saharia et al., 2022) or Dall-E 2 (Ramesh et al., 2022)	**Expert AGI** not yet achieved
Level 4: Exceptional *at least 99th percentile of skilled adults*	**Exceptional Narrow AI** Deep Blue (Campbell et al., 2002), AlphaGo (Silver et al., 2016; 2017)	**Exceptional AGI** not yet achieved
Level 5: Superhuman *outperforms 100% of humans*	**Superhuman Narrow AI** AlphaFold (Jumper et al., 2021; Varadi et al., 2021), AlphaZero (Silver et al., 2018), StockFish (Stockfish, 2023)	**Artificial Superintelligence (ASI)** not yet achieved

Genie

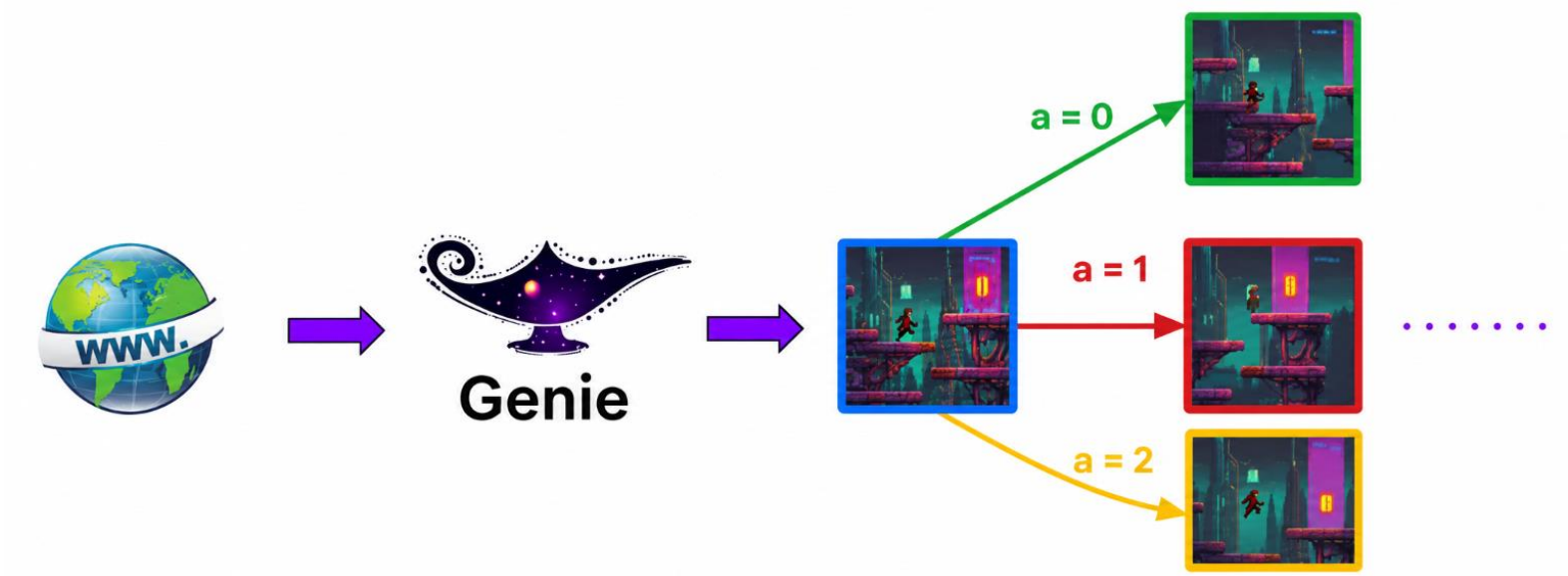

Prompt
Play: {A, B, X, Y}
Text-to-image
Hand-drawn sketch
Real-world photo
Genie
= B
= A
= B
= B
= A
= A
= A
= B
= A

A Demo

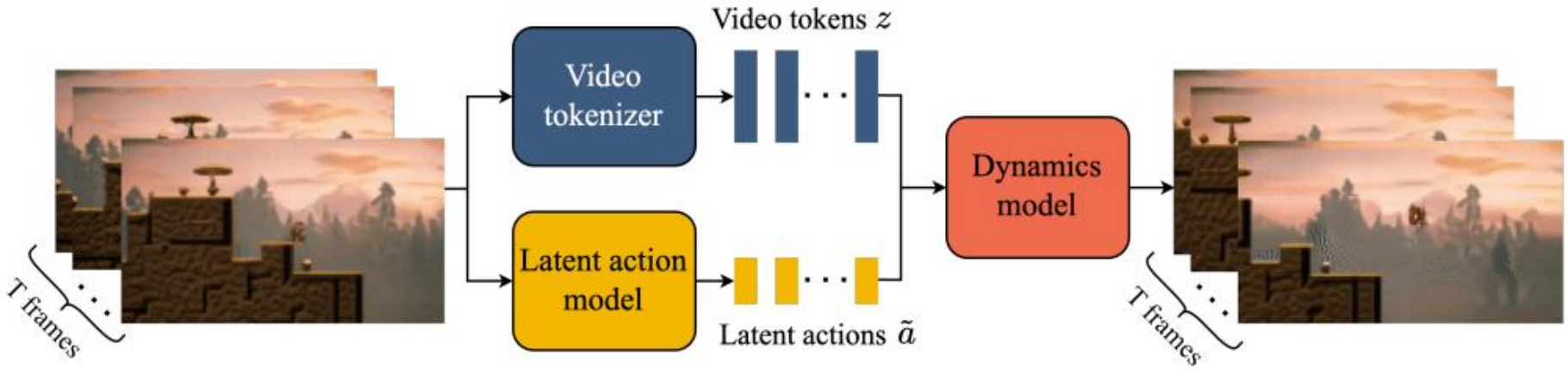


Genie's Mission

Learn a Generative Interactive Environment purely from unlabeled videos that is playable by both humans and AI agents.

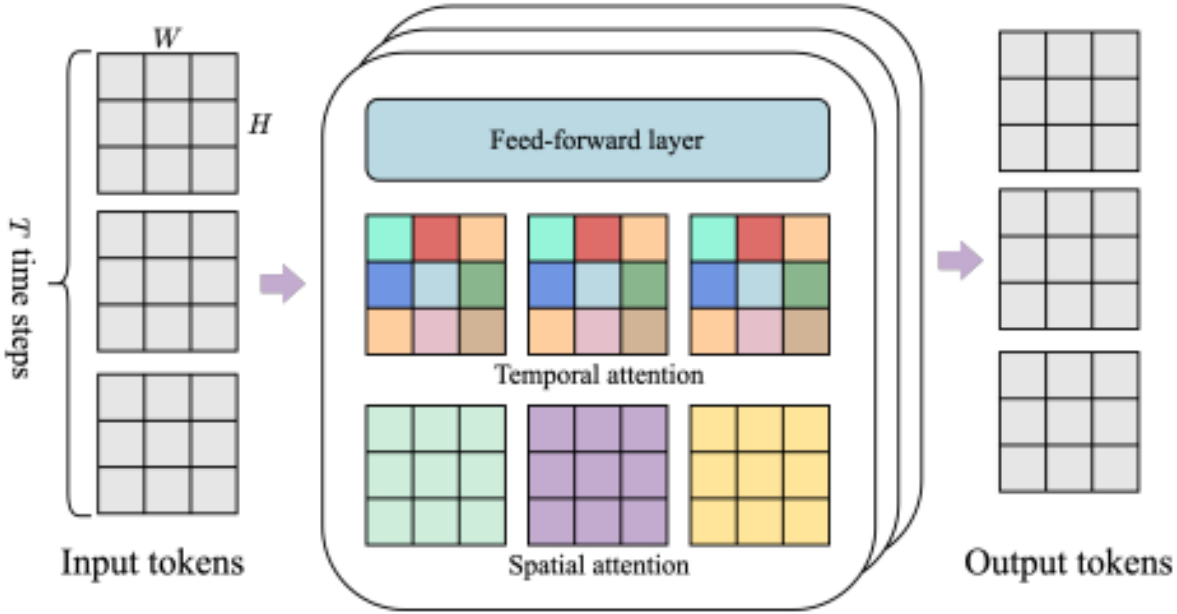


Model Training

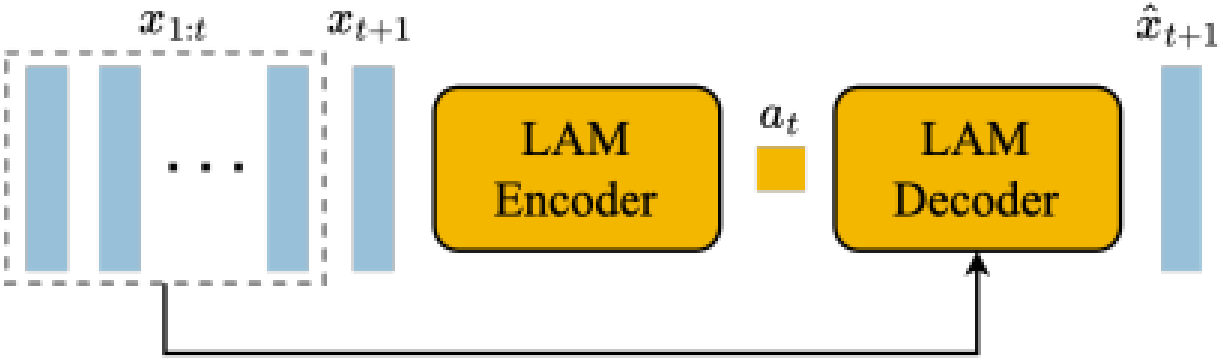


Spatio-Temporal Transformer Architecture

(ST-transformer)



Latent Action Model (LAM)



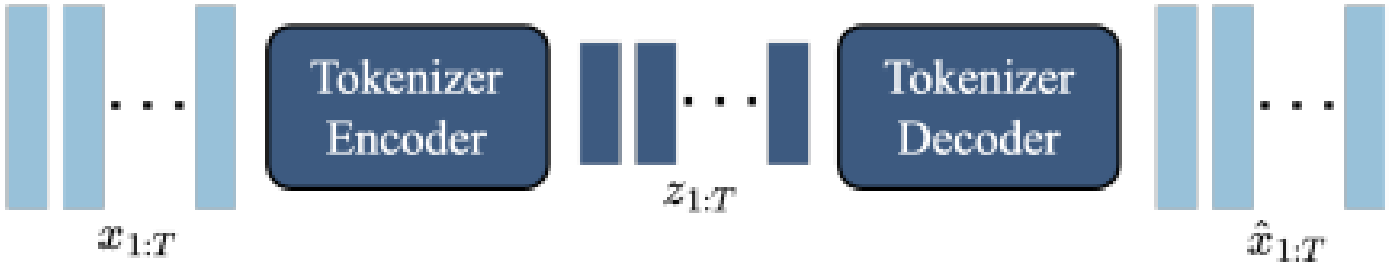
Frames: $\boldsymbol{x}_{1:t} = (x_1, \cdots x_t)$

Latent Actions: $\tilde{\boldsymbol{a}}_{1:t} = (\tilde{a}_1, \cdots \tilde{a}_t)$

VQ-VAE based training

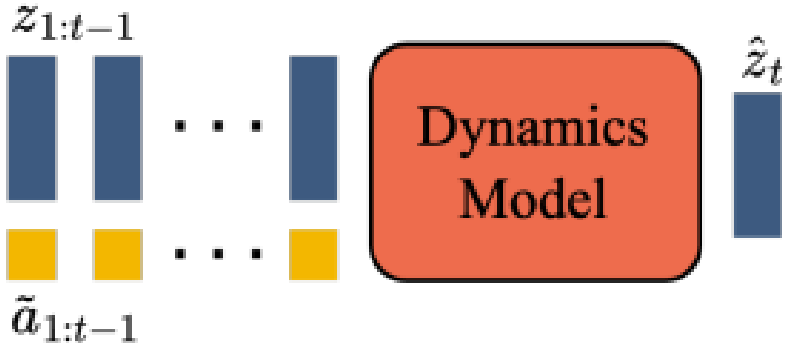
VQ codebook: **A**

Video Tokenizer



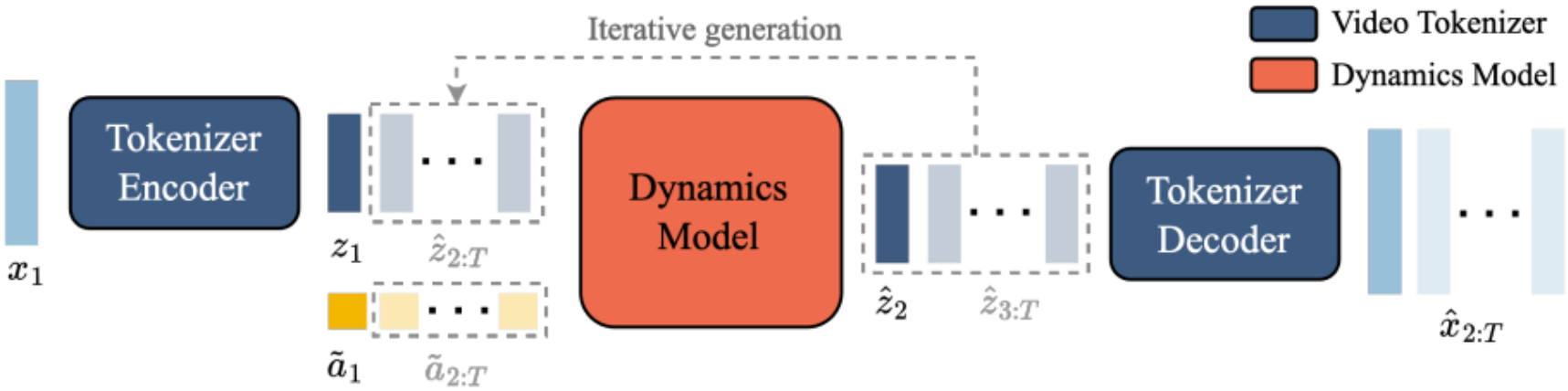
Also based on a **VQ-VAE** style architecture

Dynamics Model



A decoder only **MaskGIT** transformer

At inference

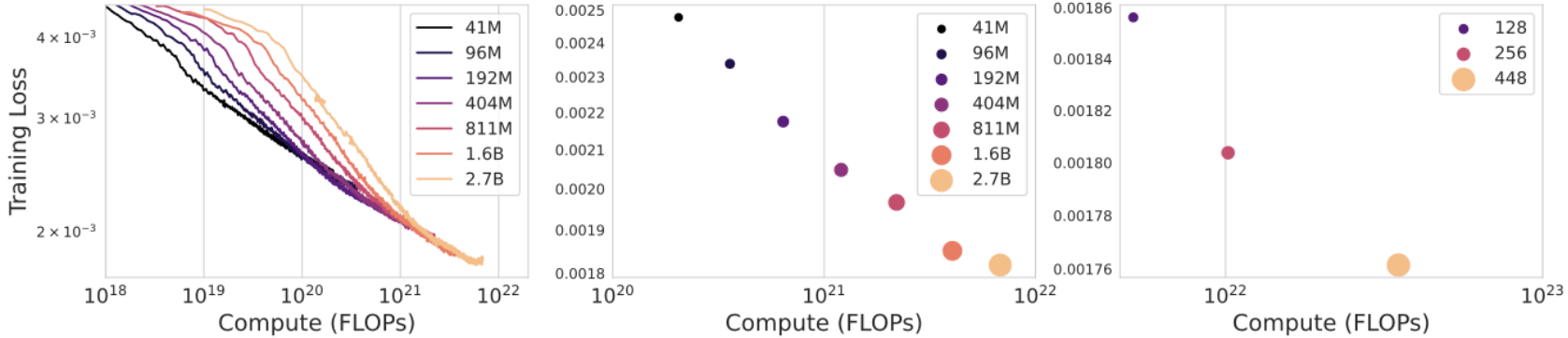


Experimental setup

- Datasets
  - Platformers: 6.8M curated gameplay clips (~30K hours), 160 × 90 resolution, 10 FPS
  - Robotics: RT-1 and additional real/simulated robot demonstrations
  - No ground-truth action labels, video-only training
- Metrics
  - Video fidelity: FVD (↓)
  - Controllability: ΔtPSNR (↑) $\Delta_t\text{PSNR} = \text{PSNR}(x_t, \hat{x}_t) - \text{PSNR}(x_t, \hat{x}'_t),$
- Training configuration
  - Video tokenizer: 200M parameters, 1024 × 32 codebook
  - LAM: 300M parameters, 8 × 32 codebook
  - Dynamics model: 10.1B parameters
  - 125k steps, batch size 512, 942B training tokens, 256 TPUv5p

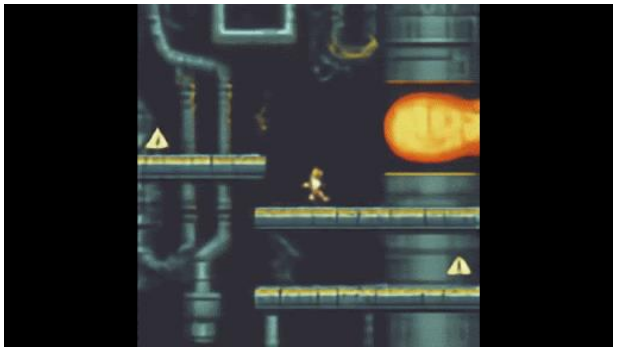
Scaling

- Scaling model size
- Scaling batch size
- Increasing model size and batch size reduces training loss



Platformer-trained model

- 11B parameter model
- Same sequence of latent actions yields similar behaviors across different prompt images



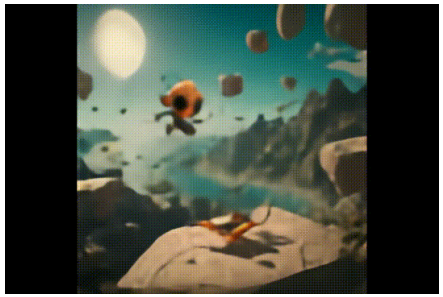


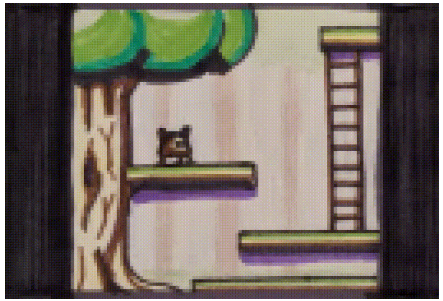
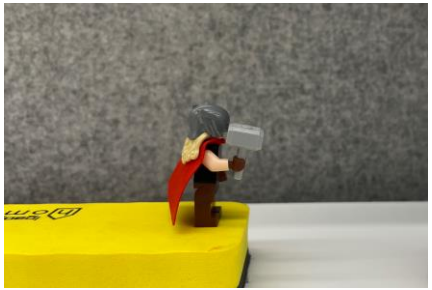
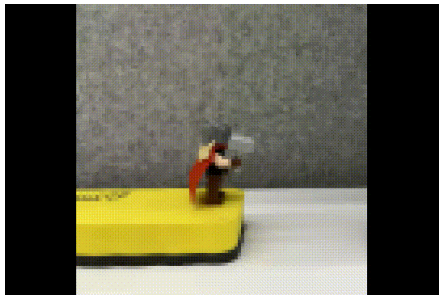
`latent actions: 5, 6, 2, 2, 6, 2, 5, 7, 7, 7`

Platformer-trained model

OOD generalization

- Images generated from text-to-image models
- Hand-drawn sketches
- Realistic photos

Platformer-trained model

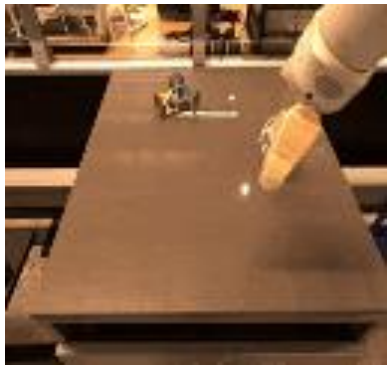
- Genie can emulate motion parallax in generated environments
- This behavior emerges from video-only training

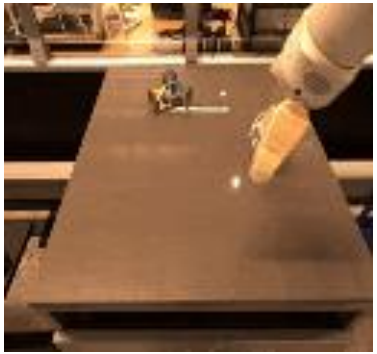


Figure 12 | **Emulating parallax**, a common feature in platformer games. From this initial text-generated image, the **foreground** moves more than the **near** and **far** middle ground, while the **background** moves only slightly.

Robotics-trained model

- 2.5B parameter model
- Trajectories with the same latent action sequence typically display similar behaviors






`latent actions: 0, 0, 1, 2, 6, 1, 1, 4, 0, 0`

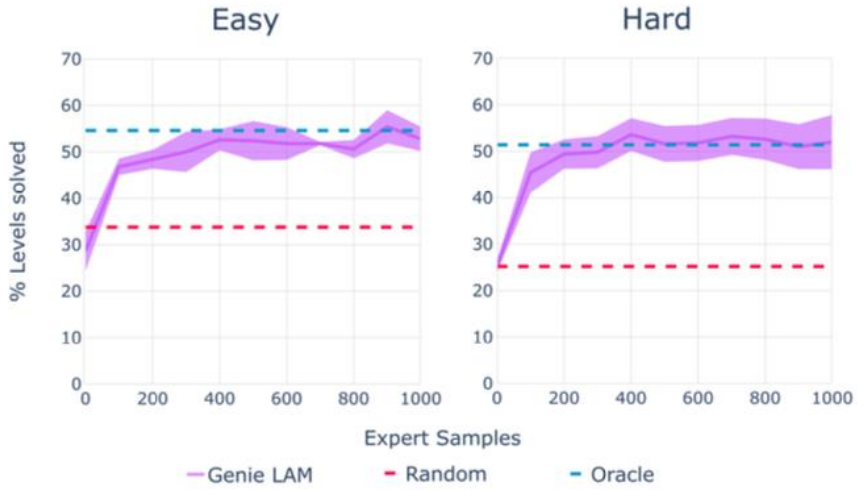
Robotics-trained model

- Genie can also simulate object deformation



Behavior cloning from unlabeled videos

- Pretrain LAM from unlabeled internet videos
- Take expert videos from target domain
- Infer latent action using pretrained LAM
- Behavior clone latent actions
- Roll out latent action policy in target environment, learning latent-real mapping from a small labeled dataset



~200 labeled samples →comparable to Oracle BC

Ablation studies

- Latent action model input ablation

	Dataset	#Params	FVD (↓)	Δ_tPSNR(↑)
Token-input	Platformers	2.3B	**38.8**	1.33
Pixel-input (**Genie**)	Platformers	2.5B	40.1	**1.91**
Token-input	Robotics	1B	257.8	1.65
Pixel-input (**Genie**)	Robotics	1B	**136.4**	**2.07**

- Tokenizer architecture ablation

	#Params	Memory	FVD (↓)	Δ_tPSNR(↑)
ViT	230M	0.3GB	114.5	1.39
C-ViViT (Villegas et al., 2023)	225M	1.6GB	272.7	1.37
ST-ViViT (ours)	205M	0.9GB	**81.4**	**1.66**

Key takeaways

- Genie learns latent actions and environment dynamics without ground-truth action labels
- Genie generates action-controllable environments from diverse image prompts, including OOD inputs
- Genie shows potential as a foundation world model for training generalist agents

Genie 4



Evolution of Genie

- **Genie 1**
  3 distinct models: LAM, Video Tokenizer, Dynamics Model
- **Genie 2**
  Autoregressive latent diffusion model
- **Genie 3**
  Environments at 720p & 24 FPS, consistency for several minutes
- **Genie 4**
  –

The Questions

1. Since the latent action codes are shared across games, do they actually represent the same underlying actions, or could differences in game mechanics cause the same code to mean different things? For example, “move right” and “jump right” might look similar while a character is already in the air. Without button labels, could the model assign both transitions the same code, even though they require different controls?
2. To what extent have the techniques used in this paper been utilized for the further development of world models?
3. One question is since the latent action model infers actions from observed frame differences, how can we ensure that the learned actions represent controllable agent (first person view character/game charactor / robotic arm) behaviors rather than camera motion or other uncontrollable environmental changes?
4. How can Genie keep the world consistent when it only remembers 16 frames?
5. Good video prediction does not always mean that a model understands how the environment works. How can we distinguish a model that learns real, controllable dynamics from one that only learns visual correlations? Are there any promising approaches to address this gap?
6. If several characters move at the same time, how can Genie learn which character should the user be able to control?
7. Why do only 8 latent actions emerge as interpretable controls? Does a larger codebook break the semantics, and how sensitive is controllability to this choice?
8. How does Genie distinguish between actual actions and random visual changes in videos? Also, why did the authors choose only eight latent actions, and would using more actions improve the model's control?

Thank you