08 Are Video Models World Simulators?

BY NILUSHANAN KULASINGHAM

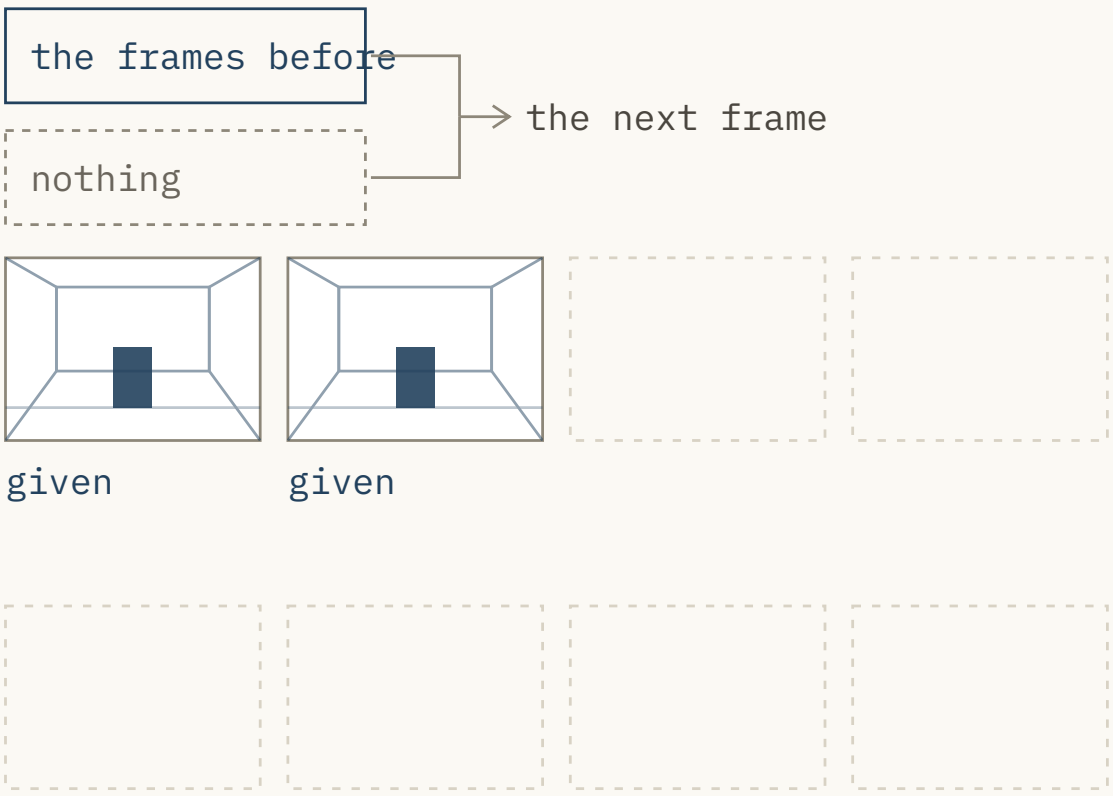
13 MIN READ · INTERACTIVE

> Add an action input to a video model and it is steerable in principle. From Genie to Genie 3: how to tell conditioning from control, what scaling bought, and why fitting physics is not the same as having the rule.

The figures in this chapter are interactive. Press and drag them online at worldmodels101.com/chapters/video-worlds.

If you have watched any generated video lately, you'll have seen somebody walk through a scene with the arrow keys. Until 2024 a video model was something you watched. It chose its own future and you had no say. One extra input changed that.

Take a model that predicts the next frame. Now tell it, along with the frames, which key is being held. From the same opening it can give you one continuation for left and another for right. Have a go with the figure below, and watch who picks each frame.



FRAMES SO FAR	CHOSEN BY YOU	THE EXTRA INPUT
2 of 8	0 of 0	none

FIG. 8.1 Two ways of getting the next frame. Leave it on watch and press Next frame: the model picks, and starting again gives you the same clip back. Switch to steer and the arrow you press goes in alongside the frames, so the frame you get is the one you asked for. Watch the tally underneath for who chose each frame. The clip is illustrative.

Jake Bruce and colleagues at Google DeepMind made that move with Genie in 2024. Their paper was called *Genie: Generative Interactive Environments*. Genie was trained on footage of 2D platform games, Mario and its descendants, and it turned them into worlds you could steer. Nobody had labelled the keypresses.

Instead, the model had to explain each change between frames with one of a handful of codes. The codes it settled on behaved like a controller. Genie also married two families that had grown up apart: world models built for agents (chapters 5 and 6)

and video models built for audiences.

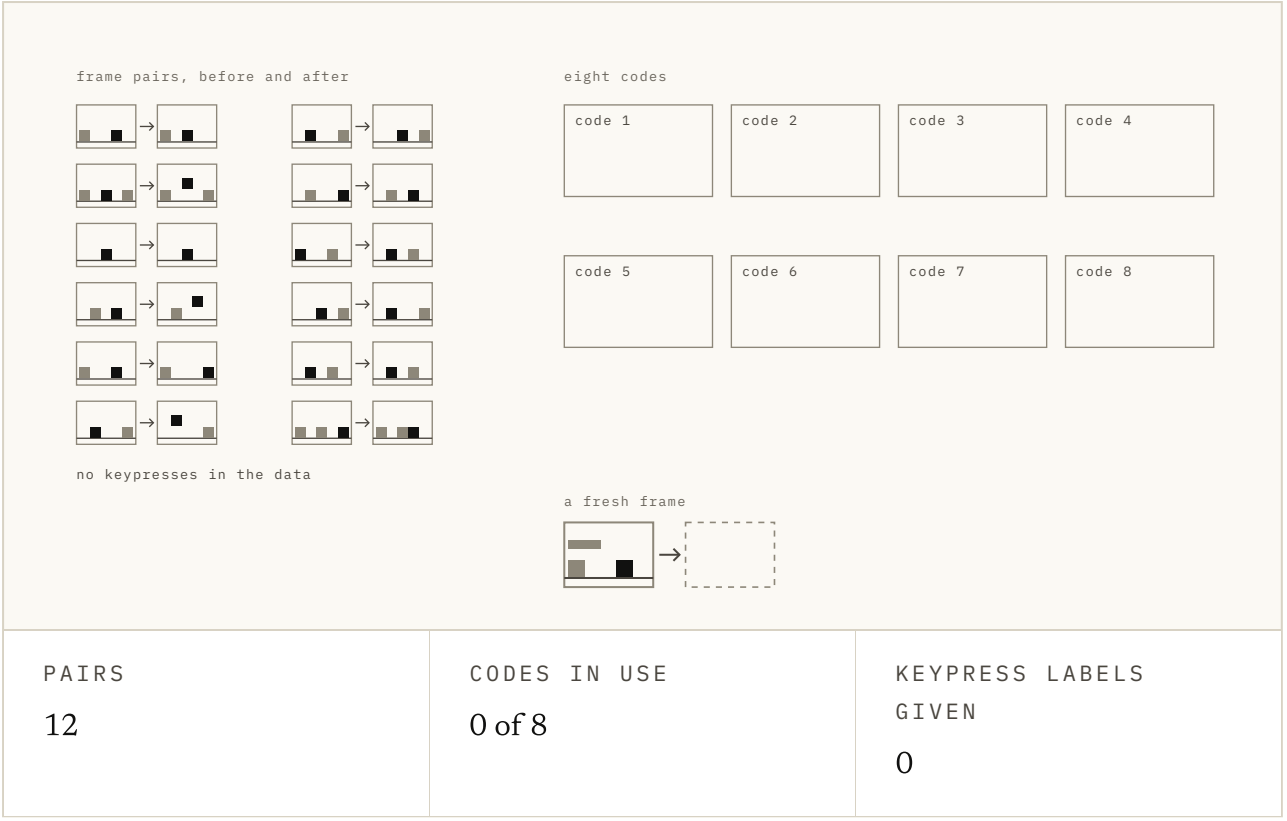


FIG. 8.2 This is that trick in miniature. The model sees pairs of frames and no keypresses, and has to file each change under one of eight codes. Press Sort the changes and watch the piles: one code collects every step left, another every jump, and nobody told it which was which. Then press a code and see what it does to a fresh frame. The frames and the codes are illustrative.

A video model with an action input is somewhere you can be, and that is new. Whether it is a simulator, something that obeys your actions and holds the rules of the world, has to be measured, and the announcements tend to skip that.

An action input is not control

A model can receive an action and still ignore it, or respond only weakly. It can also let the action change how things look without changing the path underneath. Think of a steering wheel that turns without being connected to anything.

So control has to be measured. Hold the start fixed, vary only the command, and check whether the futures split apart as asked. I call it the steering-wheel test, and the figure below is that test as a picture.

ACTION INPUT	LEFT-RIGHT SEPARATION	CONTROL TEST
present in every row	107 px	The command changes the video, but weakly.

FIG. 8.3 Every row gets a different command from the same start. Conditioning is the name for giving the model the action as an input, and action fidelity is how far the model obeys it. Drag fidelity towards zero and you should see the paths fold together while the input stays present.

As you can see, the input can be there the whole time and mean very little. That failure is the one I'd expect. A model trained on footage is rewarded for frames that look right. If the footage mostly goes straight, going straight is a good bet whatever the key says.

Good video hides weak control. If three beautiful continuations all go straight, you have been shown video. A planner that asked for three counterfactuals, three what-ifs from one start, has been handed one of them three times.

From watching to walking in

Scaling did more than most people expected, including most of the people building it. Let's take the two systems that mark the change.

GameNGen came from Dani Valevski and colleagues at Google Research in 2024. Their paper was called *Diffusion Models Are Real-Time Game Engines*. A diffusion model makes a picture by starting from noise and cleaning it up step by step. GameNGen used one to run DOOM, the first-person shooter from the early nineties.

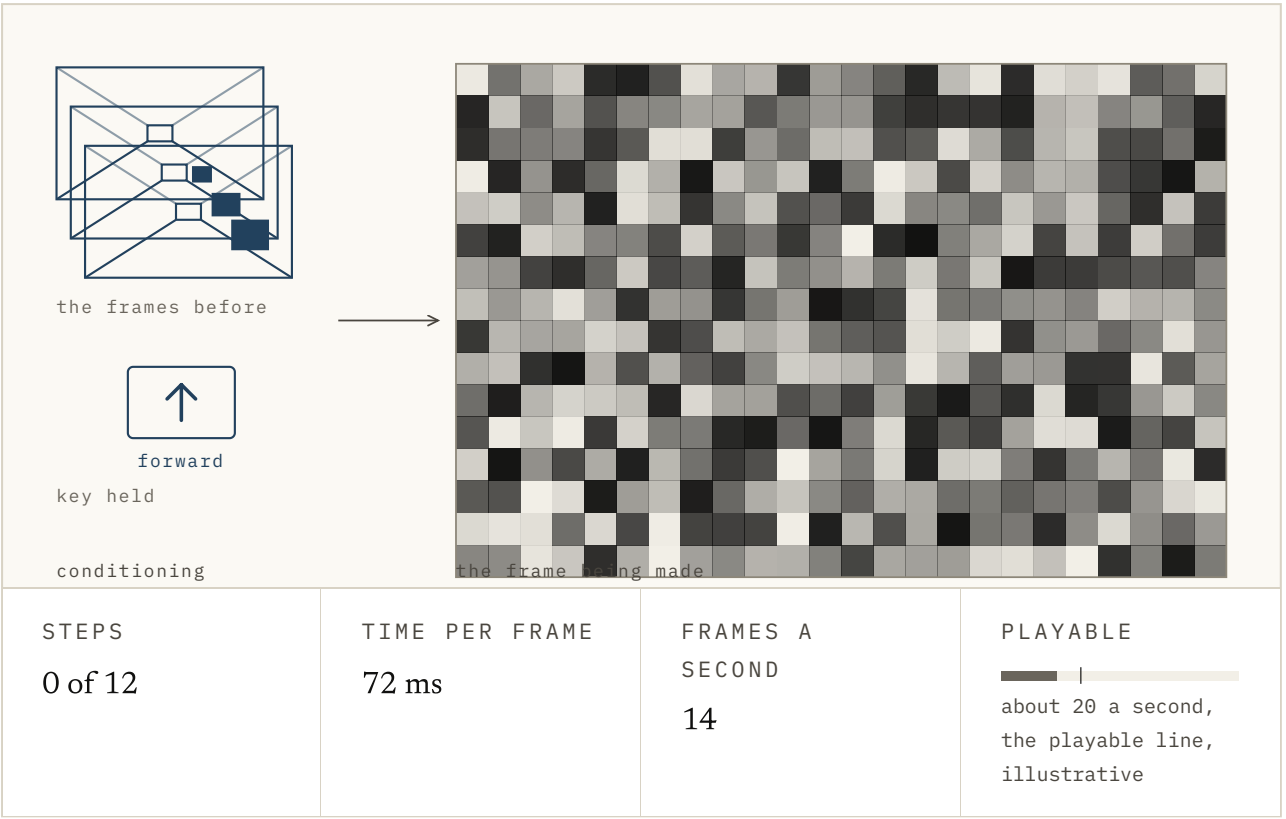


FIG. 8.4 This is how a diffusion model makes one frame of a game. The frames before it and the key being held sit on the left as the conditioning. Press Clean up and watch the noise turn into a frame one step at a time. Then drag the number of steps down and read the frames a second: fewer steps is faster and rougher, and somewhere along that slider watching turns into playing. The picture and the timings are illustrative.

It made each frame from the earlier frames and the player's inputs, fast enough to play. People shown short clips struggled to tell it from the real game, its authors reported.

Genie 3, from Google DeepMind a year later, is where the numbers stopped looking like a demo. It is reported to make interactive video at 720p and 24 frames a second.

> Those numbers are DeepMind's report of its own system. Nobody outside the lab has repeated them. I'd say that is normal here rather than a slight.

It is said to hold a scene together for several minutes and to take instructions part-way through: change the weather, add a flock of birds. Walk away from something, walk back, and it is roughly as you left it. Have a look at the clip below and watch for exactly that.

GOOGLE DEEPMIND Genie 3: Creating dynamic worlds that you can navigate in real-time

FIG. 8.5 DeepMind's own clip of Genie 3. Watch for the moments the camera turns away and comes back, and ask yourself what you were shown. The scene is roughly as it was, and nothing in the interface lets you open up whatever held it there. Everything in the clip is the lab's report, including the numbers I quoted above.

The public interface offers no stored 3D layout of the scene, and no list of its objects, that you could inspect. Yet DeepMind says Genie 3 can look back over its own path and keep visual information for about a minute. Whatever state supports that stays hidden inside the generator.

In under two years the field went from steering a blurry platformer to walking around a photoreal scene that remembered where you had been. Scaling did not give you anything you could open up and read.

Emergent physics, and where the footage runs out

You will see the claim made about all of them. OpenAI's 2024 report, *Video generation models as world simulators*, put emergent physics into common use. It was the write-up for Sora, OpenAI's own video generator. The title is the part that spread, and the claim in it bundles an observation with a conclusion.

The observation is real. Train on enough video and the outputs start to respect regularities nobody put in. Dropped objects speed up as they fall, water pools rather than piling up, and hidden objects come back into view when the thing in front moves. Nobody supplied an equation; those regularities are in the training footage, and predicting it well means copying them.

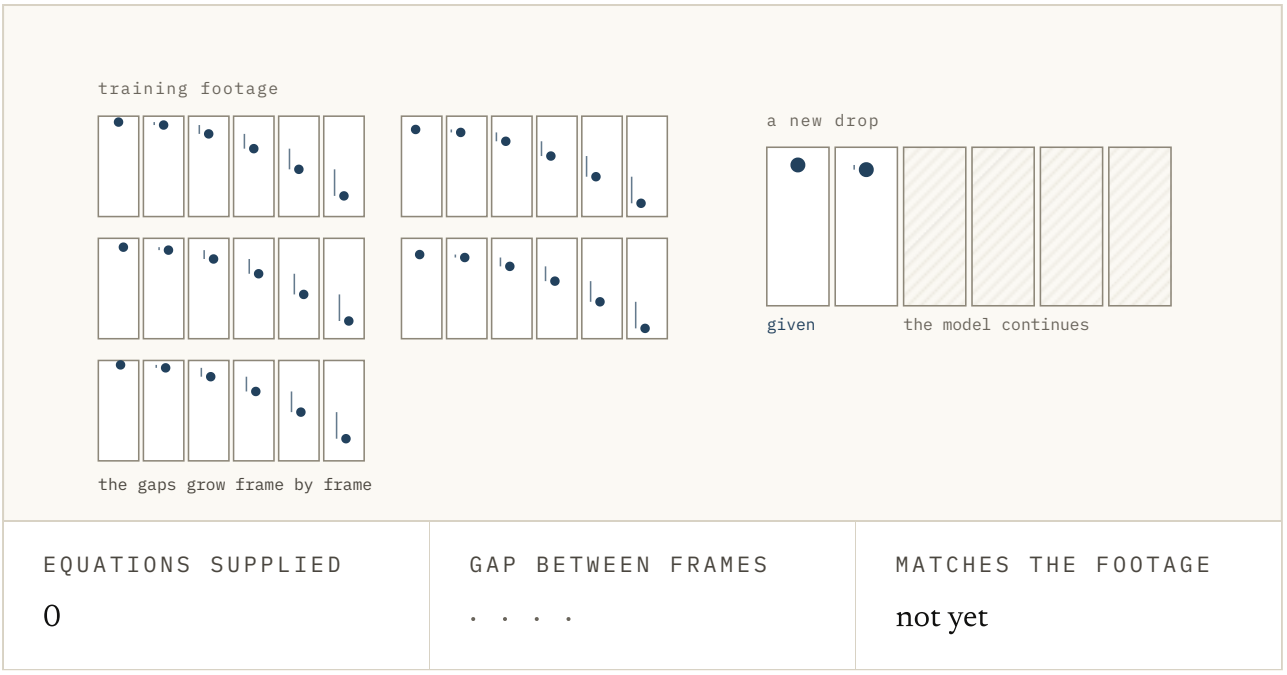


FIG. 8.6 On the left are the training clips, each one a thing being dropped, and each one speeds up on the way down. On the right, press Continue and the model finishes a drop it has never seen. It speeds up too, and nowhere in the figure is there an equation. Now switch the footage to the made-up kind, where things fall at a steady speed, and press Continue again. The regularity follows the footage. Illustrative.

The conclusion is where the trouble starts. "It learned physics" says the model picked up the rule, and that is testable. The test has never been whether the outputs look right on the footage it was trained on.

Bingyi Kang and colleagues ran one in 2024, in a paper called *How Far is Video Generation from World Model: A Physical Law Perspective*. They trained video models on simple simulated scenes of balls moving and colliding, then asked for speeds the footage never showed. Figure 8.7 runs that test twice: once on speeds the footage covered, and once on speeds past them.

FIG. 8.7 Here is that test as a scorecard. Each tile is one scene, ordered by speed, and the shaded ones are the speeds the footage covered. Press Score it, then switch the bench to scenes from past where the footage ends and score it again. Same model, same procedure, and the only thing that changed is where the scenes came from. The speeds and the errors are illustrative.

Everything inside the band fits with having learned physics. It fits just as well with having learned the answers to the questions it was asked, and Kang's group concluded their models were doing the second: matching each new scene to the training scenes it most resembled and copying what happened there. Nobody tests outside the band, because that is where the footage runs out. I call that the band problem.

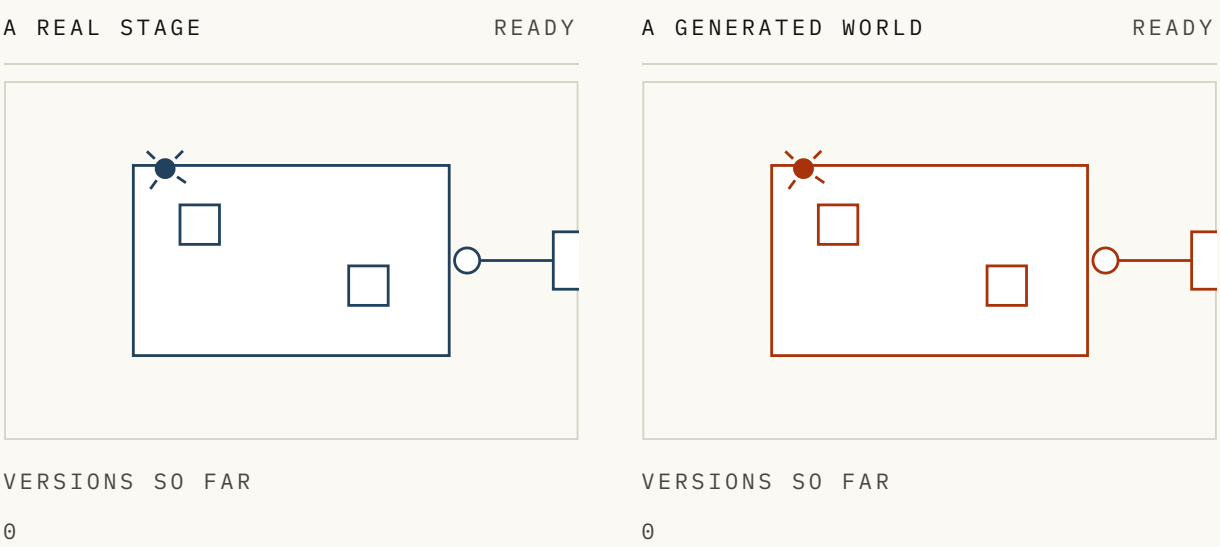
So the defensible claim is narrower. These systems copy a great many physical regularities across the situations they were trained on. Whether that is the same as having the rule, no demo you have seen can settle, because every one of them was inside the band.

> The band problem is not special to video. Any fitted function has it. It feels different here only because a wrong answer that looks like a photograph convinces people in a way a wrong number never could.

What they are for

None of that argues against using them. Answering "it has not learned physics" with "so it is a toy" is the mirror-image error, and I see it nearly as often.

For training other systems a generated world is useful. It avoids contact with the real environment and costs GPU time instead. It resets at once, and it can produce a thousand versions of a scene that would take a week to stage. Press the buttons in the figure below a few times and you'll see what I mean.



STAGE · VERSIONS	STAGE · TIME SPENT	STAGE · REAL CONTACTS	STAGE · STATUS
0	0 days	0	ready

GENERATED · VERSIONS	GENERATED · TIME SPENT	GENERATED · REAL CONTACTS	GENERATED · STATUS
0	0 frames of GPU time	0	ready

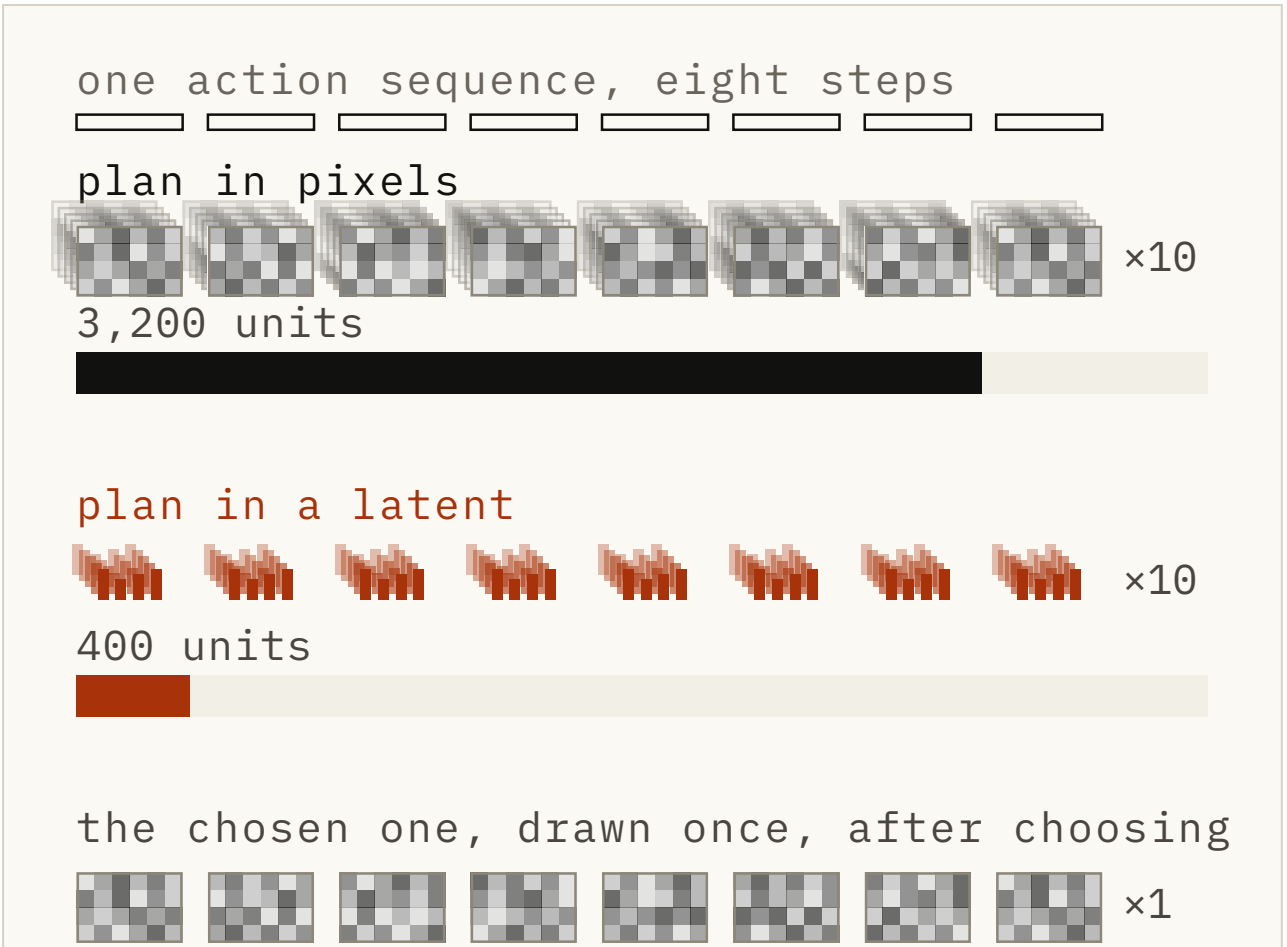
FIG. 8.8 One scene, two ways of getting more of it. On the left is a real stage: press Another and a day goes by and one new version arrives. On the right is a generated world: press Another and a version appears in the time it takes to draw a frame, with the lighting, the clutter and the camera moved. Press Reset on both and watch which one is ready. The counts and timings are illustrative.

NVIDIA, whose chips train most of these models, built Cosmos for this. Cosmos is a set of video generators. The 2025 paper is called *Cosmos World Foundation Model Platform for Physical AI*. It makes physically-aware video as training data for robots and vehicles, where real data is dearest.

NVIDIA sells Cosmos beside its explicit, hand-built simulators as one platform, and has not declared either the replacement for the other. I'd take that as a hint about where things stand.

For being a world you can move around in, they are already good. No other approach lets you walk into the picture. I think that is worth more than the sceptics admit and less than the launches imply.

For being the model an agent plans inside, raw video generators are an awkward shape. A planner may need to score a hundred action sequences, and rendering each one is expensive. Systems can instead distil a learned latent, a short internal code that stands in for the frames, and plan in that.



SEQUENCES	FRAMES RENDERED, PIXELS	CODES SCORED, LATENT
10	10 × 8 = 80	10 × 8 = 80
FRAMES DRAWN, LATENT	**COST, PIXELS**	**COST, LATENT**
8	3,200 units	400 units

FIG. 8.9 A planner has to score many action sequences before it picks one. Drag the number of sequences up and watch the two bills. Planning in pixels renders every frame of every sequence, and its bar runs off the chart. Planning in a latent, a short code standing in for each frame, scores the same sequences for a fraction of that, and only the chosen one is ever drawn. The costs are illustrative.

Where the seams are

There are three seams where a demo can mislead you.

Consistency is learned, not enforced. Nothing stops the room from changing, and usually it does not. But usually is a statistical property of a trained system, and a guarantee is a property of an engine.

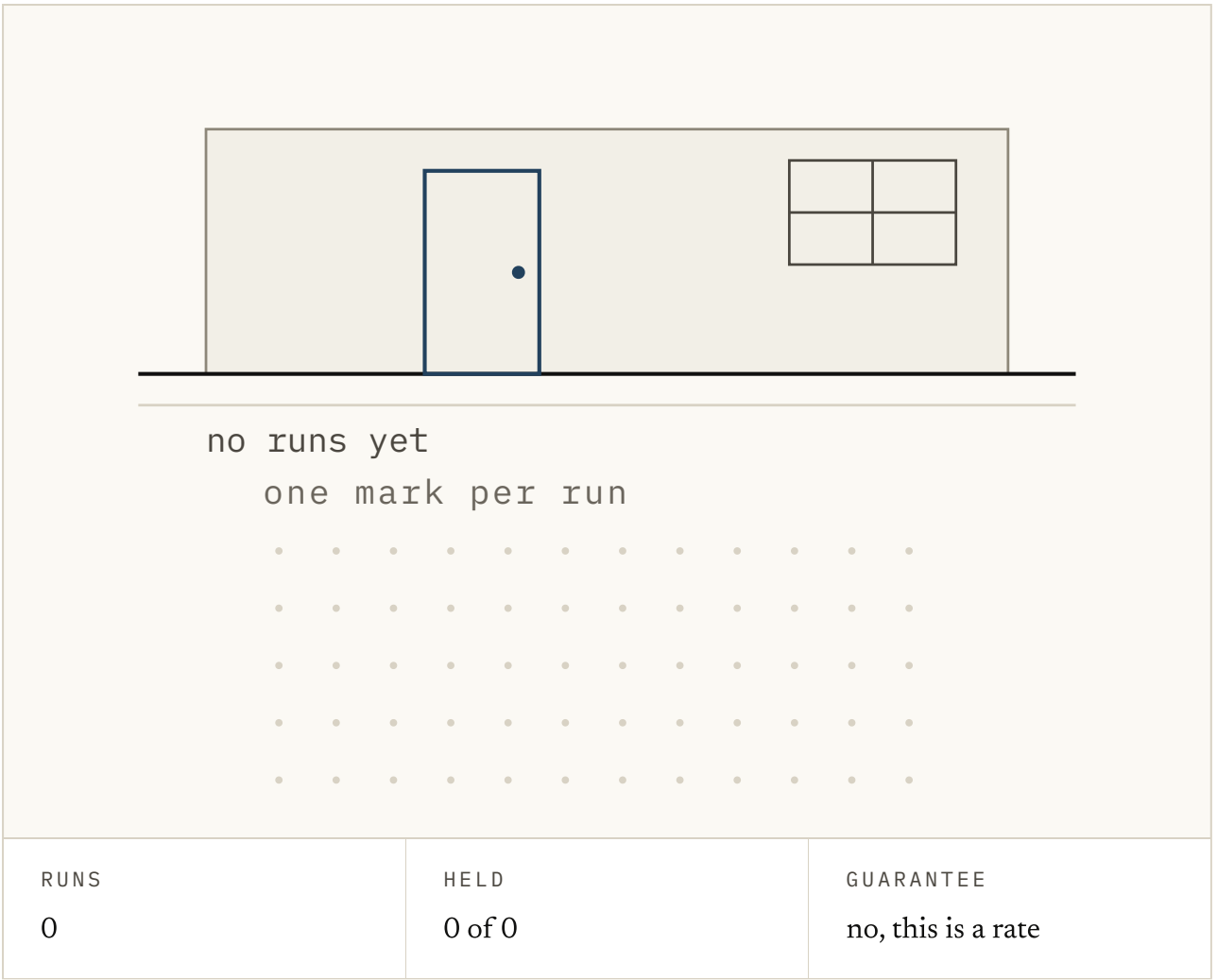
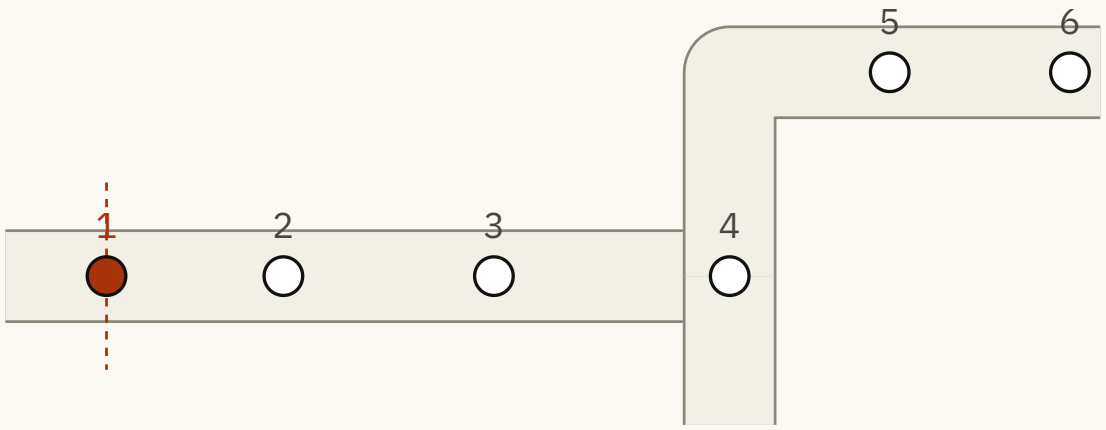


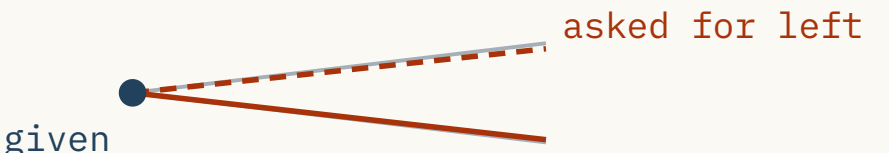
FIG. 8.10 One room, one door, and a walk away and back every time you press. Press Again a few times and the door comes back where it was, which is what usually looks like. Keep pressing and one run comes back with the door somewhere else. Then flip the switch to an engine holding the scene and there is nothing left to fail. The rate is illustrative.

The interesting moments are the vague ones. A model trained on pixels is pushed towards the average of the possible futures wherever the future is open. We saw in chapter 7 what that average looks like: a picture of something that cannot happen.

The future is most open when something is about to happen. So the steering-wheel test is hardest to pass exactly where passing it matters. Step along the clip in the figure below and you'll see why.



moment 1



how far apart they should have come out

how far apart they did

MOMENT	AT STAKE HERE	OBEYED
1 of 6	almost nothing	yes, and it did not matter

FIG. 8.11 One clip with a junction in it, and the steering-wheel test run at every moment along the way. Step through the moments and read the two bars: how far apart the two commands should have come out, and how far apart they did. In the corridor it obeys, and nothing was at stake. At the junction the two commands come out on top of each other. The moments and the numbers are illustrative.

The scores do not compose. Benchmarks measure visual quality, geometry, action fidelity and persistence one at a time. There is no accepted rule for combining them into one ranking.

So I will not tell you whether Genie 3 beats GameNGen. A claim about improvement belongs to one particular test. Click through the measures below and watch the winner change.

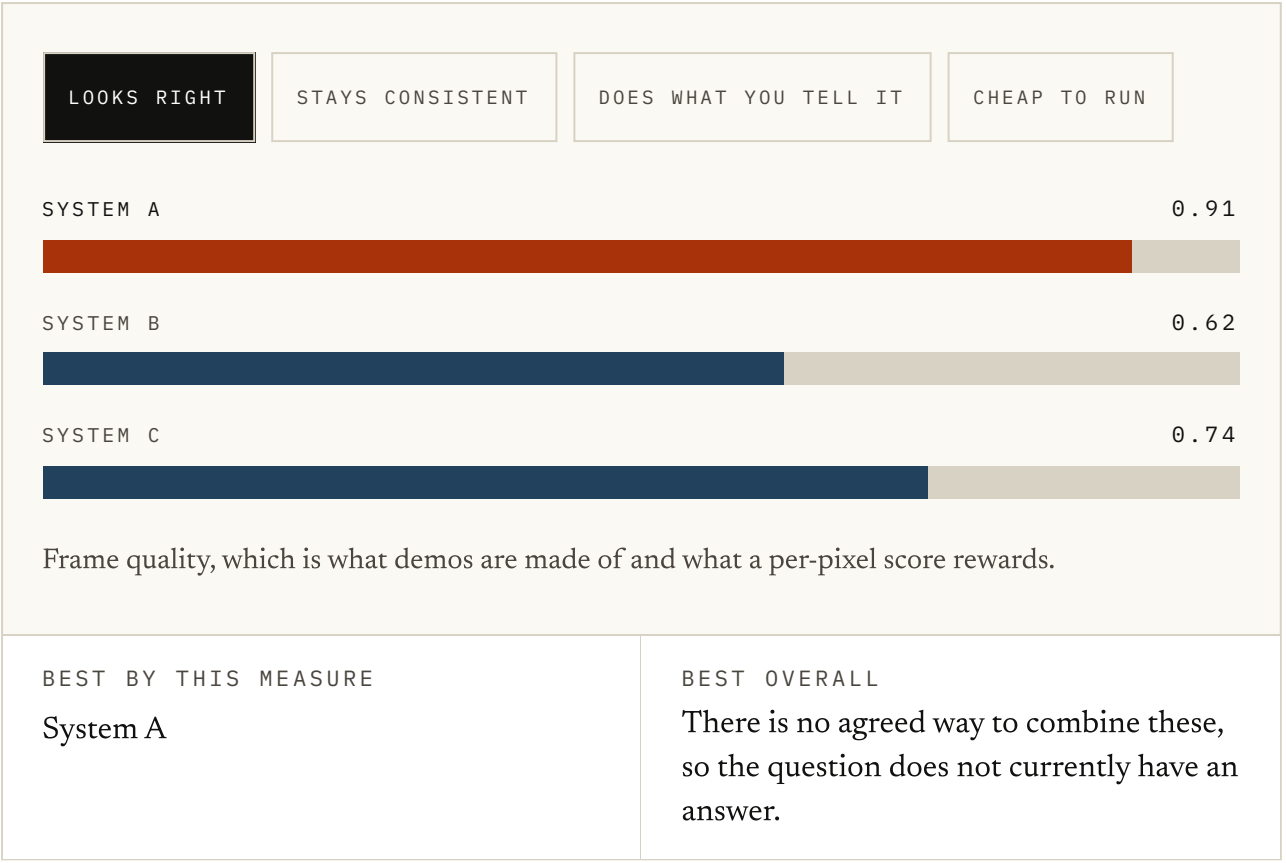


FIG. 8.12 The scores are invented and the systems are not real, but the disagreement is. Click through the four measures and watch the winner change. There is no accepted way to combine them into one ordering, so "better" only ever means better at one of these.

So, are video models world simulators? My answer is that they are somewhere you can be, and that is new and real. Whether one obeys you, and whether it holds the rules rather than the footage, are two separate measurements, and the demo you saw did not make either of them. Ask for the steering-wheel test, and ask what was outside the band.

Hopefully this chapter helped. There is a short quiz below if you want to check the ideas stuck. If I have got something wrong, or you know a better source than the ones I used, the About page has a corrections section and I would be glad to hear from you.

Try it

1 What change makes a video model capable of being steered, without proving that it will obey?

a. Higher resolution
b. A longer context window
c. More parameters
d. Conditioning each generated frame on an action as well as on the frames before it

> ANSWER d. Conditioning each generated frame on an action as well as on the frames before it. Action conditioning supplies the interface. Controllability still has to be tested by varying the action from the same start and measuring whether the requested futures separate.

2 A generated room stays as you left it when you turn back. What can you conclude about its memory?

a. Some internal state supports persistence, but the behaviour does not reveal whether it is a scene graph, context or another mechanism
b. It must use a separate object database
c. It must contain exportable geometry
d. It has no memory of any kind

> ANSWER a. Some internal state supports persistence, but the behaviour does not reveal whether it is a scene graph, context or another mechanism. Behaviour establishes persistence, not implementation. Genie 3 is reported to refer back to its trajectory, while exposing no persistent geometry or object store to inspect.

3 Dropped objects in generated video accelerate downwards, and nobody supplied gravity. What does that establish?

a. Nothing at all
b. The model has learned the law of gravity
c. That the regularity is in the training footage, and reproducing it is required to predict the footage well
d. The model contains a physics engine

> ANSWER c. That the regularity is in the training footage, and reproducing it is required to predict the footage well. The observation is real. The question is whether reproducing a regularity across the range you trained on amounts to having the rule, and the observation alone does not settle it.

4 In Figure 8.7, the model is within a few per cent across the band it was trained on. Why is that not evidence it has the rule?

a. The measurement is unreliable
b. Matching inside the band is equally consistent with having learned the answers to the questions it was asked
c. A few per cent is too large an error
d. The band was too narrow

> ANSWER b. Matching inside the band is equally consistent with having learned the answers to the questions it was asked. Having the rule and having a fit through the sampled region only come apart outside the band, and outside is where nobody tests because there is no footage to compare against.

5 Why is a generated video world the wrong shape for an agent to plan inside?

a. It cannot be conditioned on actions
b. It has no reward signal
c. It is not accurate enough
d. Planning means running the model many times over, and a frame is the opposite of a compact state

> ANSWER d. Planning means running the model many times over, and a frame is the opposite of a compact state. A planner needs to try many action sequences and score them cheaply. These models are expensive to run and produce frames rather than something small to reason over.

6 What are generated worlds unambiguously good for right now?

a. Producing training data and being a place you can move around in
b. Long-horizon planning
c. Compressing video
d. Replacing physics engines

> ANSWER a. Producing training data and being a place you can move around in. They avoid real-world contact, reset instantly and produce broad variation, while still costing compute. No other approach in this course gives you an actual picture to walk into.

7 Why does a wrong answer from one of these systems feel more convincing than a wrong number?

a. It usually is not wrong
b. The errors are smaller
c. It arrives as something that looks like a photograph
d. The models are better calibrated

> ANSWER c. It arrives as something that looks like a photograph. The extrapolation problem here is the ordinary problem with any fitted function. What is different is how persuasive the output is when it fails.

8 A system improves its geometry score but loses action fidelity. Is it better overall?

a. Yes, geometry is the only objective measure

b. Not without a declared use case or rule for combining the dimensions
c. No, action fidelity always dominates
d. Yes, any benchmark gain establishes overall progress

> ANSWER b. Not without a declared use case or rule for combining the dimensions. Benchmarks now measure several useful dimensions. The unresolved part is how to combine them when systems and applications value different contracts.

FIG. 8.13 Eight questions on the move and the claim. If a demo has just impressed you, go straight to the middle three.

Sources

Genie: Generative Interactive Environments BRUCE ET AL., 2024 ↗

Latent actions learned from unlabelled video, which is the move that turns a video model into somewhere you can be.

https://arxiv.org/abs/2402.15391

Genie 3 GOOGLE DEEPMIND, 2025 ↗

The reported numbers this chapter quotes: 720p, 24 frames a second, minutes of coherence. A lab report rather than a result anyone outside has repeated.

https://deepmind.google/blog/genie-3-a-new-frontier-for-world-models/

Diffusion Models Are Real-Time Game Engines (GameNGen) VALEVSKI ET AL., 2024 ↗

DOOM generated frame by frame from previous frames and inputs, fast enough to play. The clearest demonstration that this is playable rather than merely watchable.

https://arxiv.org/abs/2408.14837

How Far is Video Generation from World Model: A Physical Law Perspective KANG ET AL., 2024 ↗

The measured version of Figure 8.7. Video models match physical laws within the distribution they were trained on, and do not extrapolate outside it.

https://arxiv.org/abs/2411.02385

Video generation models as world simulators OPENAI, 2024 ↗

The report that made the emergent-physics claim a mainstream one. Worth reading for exactly what it does and does not assert.

https://openai.com/index/video-generation-models-as-world-simulators/

Cosmos World Foundation Model Platform for Physical AI NVIDIA, 2025 ↗

Generated video built as training data for robots and vehicles, which is the use these systems are unambiguously good for.

https://arxiv.org/abs/2501.03575

Cosmos NVIDIA ↗

The product framing, and a useful boundary case: generative video beside explicit simulation, sold as one platform.

https://www.nvidia.com/en-us/ai/cosmos/

FIG. 8.14 I've ordered these for someone building on this rather than for historical completeness. Kang and the OpenAI report sit together as claim and measurement.