The Form and the Shadow

A Substrate-Surface Paradigm for Visual World Models

First Intuition Research Team

Abstract

Within twelve months, two leading laboratories on two continents have independently shipped the same architectural fragmentation: an editable, persistent 3D scene generator alongside an interactive but amnesic video model, joined by nothing more than hand-coded routing rules. We argue this convergent failure mode is the empirical signature of a missing paradigm-level commitment. We propose the *Substrate-Surface Paradigm* (SSP): a class of controllable visual world models defined by four structural axioms — ontological separation of substrate from surface, physically-governed (not statistically-derived) substrate dynamics, a generative surface with capacity-bounded short-horizon memory, and gated bidirectional channels mediated by a learned commitment operator. The commitment operator is a named architectural slot whose contribution here is normative rather than implementational: we argue the slot must exist, must be learned (not engineered), must be capacity-bounded relative to the surface model, and must mediate four distinct write-back channels — external observation, user edit, imagination commitment, and agent interaction — with the last being ontologically primary. We exhibit one concrete instance, SSP-PMGS, satisfying the four axioms within a 33 ms-per-frame illustrative budget on consumer hardware; the paradigm survives its substitution. We commit to five falsifiable predictions and engage directly with the JEPA, Dreamer, active-inference, and representation-conditioned-generation programs.

"I like to think that the moon is there even when I am not looking at it."

— Albert Einstein, as related by Abraham Pais (1979)

1 Introduction & Position

1.1 The Back-Door Convergence

Within twelve months, two leading laboratories on two continents have, independently and almost simultaneously, shipped the same architectural fragmentation. World Labs, the company founded by Fei-Fei Li, Justin Johnson, Ben Mildenhall (the first author of NeRF), and Christoph Lassner (the author of Pulsar), ships **Marble** [40] — a 3D Gaussian Splatting (3DGS) generator that produces persistent, downloadable, editable scenes — alongside **RTFM** [41], an autoregressive diffusion transformer that streams pose-conditioned frames on a single H100 with no explicit geometry. The two products share a vision but not a runtime: Marble is persistent and static; RTFM is interactive and amnesic. Tencent Hunyuan has released seven world-model-adjacent systems in the same window, from **HunyuanWorld 1.0** [33] (layered panorama-to-mesh) through **HunyuanWorld-Voyager** [35] (joint RGBD video diffusion with a reprojected "world cache"), **Hunyuan-GameCraft 1.0 / 2.0** [30, 31] (14B-parameter video MoE whose authors openly concede semantic drift past ~500 frames), to **HY-World 2.0** [36] (a four-stage 3D-GS pipeline whose

second stage — *WorldNav* — is a hand-coded camera-trajectory planner that does, by engineering fiat, exactly the bridging job a learned arbiter should do).

This is not a coincidence of product strategy. It is the empirical signature of a problem the field has correctly diagnosed and incorrectly architected. Both labs have concluded — independently of each other and of the academic literature — that a 21st-century visual world model needs two things: a *persistent, queryable, editable representation of geometry and identity*, and a *generative process that can imagine the appearance, motion, and texture of what that representation alone cannot specify*. Neither lab has joined the two through anything more sophisticated than a fixed routing rule. Both ship the disjunction as a product line.

We argue that the disjunction is not a product gap. It is a missing architectural primitive: a **calibrated fuser** that arbitrates, at every step and at every spatial region, between persistent state and generative imagination on the basis of explicit uncertainty channels emitted by each substrate. The current generation of systems has converged on the diagnosis through the back door. This paper proposes putting the door — and the lock — in the front.

1.2 Scope: What We Mean (and Do Not Mean) by "World Model"

The phrase *world model* is overloaded to the point of dishonesty. Karpathy's 2024 observation that it conflates incompatible objectives is correct [16]. Before we propose anything, we commit to which target we are after.

A *world model* in the contemporary literature can mean any of:

1. **Data-conditional generation**: produce a plausible video given a prompt (Sora, Veo, Kling, Hunyuan-Video).
2. **Action-conditional simulation**: predict the next observation given an action (DreamerV3 [12], V-JEPA 2-AC [3], GameNGen [37]).
3. **Model-based planning**: roll out latents under a learned dynamics to choose actions (MuZero [28], LeCun JEPA planning [19]).
4. **Controllable persistent simulation**: maintain an editable, queryable representation of a place such that an external agent or human can act in it and re-enter it (Genie 3 [9], Marble, GameCraft).

These four are *architecturally incompatible*. (1) optimizes pixel likelihood; (2) optimizes one-step prediction; (3) optimizes return; (4) optimizes consistency under intervention and re-query. A system tuned for one is mis-tuned for the others, and a position paper that pretends to address all four is engaging in marketing.

This paper takes a position on objective (4): controllable, editable, 3D-consistent simulation with a human or agent in the loop. We do not contest the value of (1)-(3) — Sora, DreamerV3, and V-JEPA-2 are real achievements in their respective objectives — but we argue that (4) is a *substantively different object* that requires its own architecture, and that this difference is not yet acknowledged in the literature. The controllable-simulator objective imposes two requirements that the other three do not:

- **C1 (Immediate Editability)**: a user-initiated edit to a named object — moving a chair, deleting a wall, changing a material — must be visible in the next rendered frame with the correct geometric, occlusion, and contact consequences propagated. We require edit-to-visible latency under 100 ms.

- **C2 (Loop-Closure Consistency)**: a camera trajectory γ that returns to a previously-visited pose must yield a geometric state within ϵ of the original, measured by Chamfer distance over visible regions, for trajectories at least 60 s long.

The empirical record so far supports a sharp claim: under C1 and C2, no monolithic end-to-end neural network — however scaled — will outperform a system whose substrate, surface, transition, and commitment operator are explicitly factored. The remainder of the paper argues for this position from first principles and specifies one architecture that satisfies it. A reader who works on (1), (2), or (3) is welcome to read this as a parallel paradigm, not a competing one; the architectures genuinely diverge at the objective level, not at the scale-up frontier.

1.3 The Position

We propose the **Substrate-Surface Paradigm (SSP)**: a class-level commitment about what a controllable visual world model *is*, prior to any choice of architecture. SSP rests on a single ontological claim — that there exist *two* representations of a world, not one — and four axioms that follow from it.

> **Position.** A visual world model is not a single neural network. It is a system that simultaneously maintains an explicit **substrate** $\mathcal{S}$ — the ontology of what persistently *subsists* — and a derived visual **surface** o — the appearance generated for an observer at a viewpoint. The two are bound by explicit, learned, gated bidirectional channels. The paradigm is defined by four axioms (§4):

Axiom	Content
A1 (Ontological Separation)	Two distinct representations: substrate (*what is*) and surface (*what appears*). Neither reduces to the other.
A2 (Physical Substrate)	Substrate evolution is governed by physical laws (persistence, locality, conservation), not by the statistical distribution of training data alone.
A3 (Generative Surface)	Surface $= R(\mathcal{S}, \mathcal{C}, \tau)$ is a (learned) rendering function with at most short-horizon imagination buffer. It does not store long-horizon world state.
A4 (Gated Bidirectional Channels)	Surface $\rightarrow$ substrate write-back exists but is *explicit and event-driven*, not per-frame fusion. Channels: external observation, user edit, imagination commitment, agent interaction.

> A system that violates any axiom is not, in our sense, a world model — it may be a generator, a renderer, or a controller, but it cannot simultaneously satisfy immediate editability (C1) and loop-closure consistency (C2). Every existing system we know of violates at least one axiom; §3 and §4.3 enumerate the violations. The remainder of the paper specifies a concrete architecture, *SSP-PMGS*, that satisfies all four, and commits the engineering to a 33 ms-per-frame budget.

Beyond correctness, the four axioms have a second consequence the field has not yet articulated: they make the system's per-frame compute scale with *what has changed in the world*, not with the wall-clock frame rate. Substrate-explicit storage lets the rasterizer skip frames in which the substrate has not been edited; the gated channels keep the commitment operator quiet between events; the surface refiner runs only on tiles where the rendered substrate has revised. §7 derives a 3×–10× streaming throughput gain from this property alone. SSP is not only a *correctness* paradigm; it is a *controlled-context* paradigm. Efficiency is a derived consequence, not a separate concern.

This position commits us to falsifiable empirical claims, given in §8. The two strongest:

- **P1.** By end of 2027, no pure-video world model under 1T parameters will achieve under 5% Chamfer drift on 60-second re-entry without an explicit geometric memory.
- **P2.** The winning architecture on controllable-world benchmarks (C1+C2 satisfying systems) will not be end-to-end trained; it will exhibit a factored interface between persistent state and a generator.

If either fails, the position is wrong.

1.4 Roadmap

§2 diagnoses why each of the three pure routes — pixel-only generative, latent-only predictive, geometry-only renderable — fails C1 and C2, with quantified evidence. §3 examines World Labs and Tencent Hunyuan as case studies in the back-door convergence we open with. §4 states the Substrate-Surface Paradigm as four axioms, gives a violation table for the current literature, and defines a concrete architectural instance (*SSP-PMGS*) that satisfies all four. §5 specifies the four *channels* by which surface content writes back to substrate — external observation, user edit, imagination commitment, agent interaction — and argues that the last is ontologically primary: a thing subsists in the substrate because it can be acted upon. §6 formalizes the paradigm mathematically. §7 lays out a compute-optimal deployment with a 33 ms latency budget. §8 makes axiom-level predictions, acknowledges what would falsify them, and lists open problems. Throughout, we engage directly with Ha and Schmidhuber [11], Friston's active inference [10, 27], Hafner's Dreamer line [12], LeCun's JEPA program [3, 19], and He's representation-conditioned generation [21, 22], because the SSP thesis is best understood as a precise position taken across their disagreements.

2 Diagnosis: Three Failure Modes

We argue that any system that collapses substrate and surface — or that fails to maintain explicit channels between them — into a single monolithic mechanism exhibits one or more of three quantifiable failure modes. We name them: the *Re-Entry Wall*, the *Externalization Gap*, and the *Imagination Gap*. The first afflicts pixel-only video generators; the second afflicts pure-latent predictors; the third afflicts pure-3D reconstruction pipelines. We treat each in turn.

2.1 The Re-Entry Wall (afflicts pixel-only generation)

A *re-entry wall* is the empirically observed phenomenon by which a video world model, having shown the user a region of a scene, cannot reproduce that region's geometry, identity, or content when the camera returns to it after a horizon of seconds to minutes. The wall is not a soft degradation. It is a sharp ceiling reproducible across all current systems.

Empirical evidence. Wu et al. [43] construct a controlled benchmark in which a camera moves away from a scene and returns. Three current video world models — TrajectoryCrafter, DaS, and Wan2.1-Inpainting — score PSNR 11.7 on re-entry. The same models augmented with an explicit 3D memory score 19.1 — a 58% jump. DeepMind reports that Genie 3 maintains "about a minute of visual memory" before degradation becomes obvious [9]; Decart Oasis drifts within seconds [8]; GameNGen carries roughly 3 seconds of frame history [37]; Hunyuan-GameCraft-2's own technical report concedes "the model's lack of an explicit long-term memory mechanism ... relies instead on the finite capacity of its KV cache," with semantic drift after about 500 frames [31]. VBench-2.0 [14] reports that Sora scores 8.06% on dynamic attribute and 62.22% on mechanics; the "fake eating, fake cutting, fake walking" failure class is documented at over 80% across leading systems.

Why the wall exists. The wall is not an engineering bug. It is an information-capacity theorem. Let R_g denote the rate (in bits) of the time-invariant component of a scene — its geometry, materials, identity. Let $R_d(t)$ denote the rate of the time-varying component — motion, lighting, viewpoint. For typical indoor scenes, $R_g \approx 10^7$ to 10^8 bits while $\int R_d(t)\,dt \ll R_g$ over horizons of minutes. The mutual information $I(\text{frame}_t; \text{geometry})$ is asymptotically R_g at every t, but a pure video generator carries no persistent variable storing the geometry; instead the model re-emits R_g on every frame from a slowly-decaying KV-cache. The retrievable mutual information of the cache decays as $\exp(-t/\tau)$ with τ on the order of 60 seconds for current attention architectures, which is exactly where the wall appears empirically. A code that re-emits R_g at every step uses $\Theta(T \cdot R_g)$ bits where the optimal code uses $R_g + T \cdot R_d$. The redundancy ratio over a 1000-frame trajectory is approximately 10^3. This is not a tunable inefficiency; it is a rate-distortion bound.

Why scaling does not fix it. Kang et al. [15] scaled video diffusion from 22M to 310M parameters and training trajectories from 30,000 to 3,000,000. Out-of-distribution physical generalization did not improve. The authors conclude that current video diffusion models "case-match the nearest training example rather than learn Newton's laws." Whatever the right structure of generalization in this domain is, the data shows it cannot be reached by scale alone within current architectures. LeCun [19] gives a related theoretical argument: for autoregressive pixel-token generation with per-token error probability ε, $P(\text{coherent sequence of length } n) = (1-\varepsilon)^n$; the probability of long-horizon coherence decays exponentially. Whether one believes the argument as a fully general indictment of autoregressive generation or only as an asymptotic worst case, the empirical decay rate of current systems — 60-second visual memory in Genie 3, ~500-frame drift in GameCraft-2, 3-second history in GameNGen — sits comfortably within its prediction.

Compute consequence. The same redundancy that drives the re-entry wall also drives a quantifiable compute cost. A 3D Gaussian Splatting representation renders 1080p at 100+ frames per second on under 5 TFLOPS [17]. A video diffusion model renders 720p at 24 frames per second on roughly 100 TFLOPS. The 20× compute gap exists for one reason: the diffusion model re-derives the geometry on every frame, while the Gaussian model stores it once and rasterizes. This is a Chinchilla-style allocation failure [13]: ~95% of the diffusion model's per-frame budget is spent on R_g, which should be paid once. Scaling the diffusion model further moves more compute into the bottleneck rather than out of it.

2.2 The Externalization Gap (afflicts pure-latent prediction)

The opposite failure mode afflicts systems that, like LeCun's V-JEPA family [2, 3, 5], explicitly avoid pixel reconstruction and predict only in a learned latent space. The argument for this design is sound: a latent prediction objective grants the model the "right to ignore" unpredictable pixel-level detail (texture noise, fluid motion), and information-theoretically dedicates capacity to predictable structure. The cost is the *externalization gap*: the model can predict, but cannot show. Three

downstream consequences follow.

No human-comparable rollouts. A V-JEPA latent can be evaluated against another V-JEPA latent, but its rollouts cannot be displayed to a user or compared pixel-wise to ground truth. Evaluation collapses to representation-quality probes — IntPhys score (V-JEPA 2 achieves 98% on intuitive physics [3]) and downstream task linear-probe accuracy — neither of which captures whether the model could *render* what it has predicted. For the controllable-simulator objective (4), this is fatal: by construction, the user must be able to see and edit what the model predicts.

No interface to existing tooling. A simulator that is purely latent cannot be loaded into Blender, Unity, Isaac Sim, or any human-facing inspection tool. The space of useful external collaborators — physics engines, mesh editors, robotics simulators — speaks in explicit geometric tokens, not in 768-dimensional vectors. Marble's commercial bet is precisely on this: it exports 3DGS and mesh, because the customer base is artists, game studios, and digital-twin builders [40]. A pure-JEPA system cannot serve this customer.

Latent rollout drift. V-JEPA 2-AC, despite its design philosophy, autoregresses in latent space and exhibits the same long-horizon drift as pixel autoregression, simply in a different metric [3]. The "right to ignore" buys robustness against fine-grained nuisance variance, not against accumulated transition error. A 60-second open-loop latent rollout is not stable in any current JEPA model. LeCun's proposed cure — energy-based mode-2 inference with iterative latent refinement — has not yet been shown to scale to natural scenes.

These are not arguments against the JEPA program. They are arguments that JEPA solves a different problem: it provides a high-quality *transition* function $\mathcal{T}$ in latent space, which is precisely one of the four SSP roles. The SSP-PMGS instance of §4.5 adopts JEPA as the transition substrate. SSP rejects JEPA's renunciation of an externalizable surface as overshoot: predicting in a smart latent is correct; refusing to ever decode pixels is a 2022 over-correction that the controllable-simulator objective cannot afford.

2.3 The Imagination Gap (afflicts pure-3D reconstruction)

The third failure mode is the mirror image of the first. A system that maintains only an explicit 3D state — a NeRF, a Gaussian splat, a textured mesh — is editable, persistent, and renderable; it satisfies C1 and C2 trivially. It cannot, however, *imagine*. It can show what it has been shown. It cannot fill an occluded room, animate a static object, generate a counterfactual lighting condition, or synthesize the texture of an unobserved surface. Marble can produce a static, downloadable room from a panorama; it cannot generate a person walking through it [40]. WonderWorld and LucidDreamer [6, 44] augment 3DGS with inpainting priors, but the priors are 2D diffusion models bolted to a 3D substrate, and their outputs are one-shot and dynamics-free.

Pure-3D pipelines also inherit four structural limitations of explicit geometry that we name to avoid overclaim:

1. **Dynamics are expensive.** 4D Gaussian variants [23, 42] achieve real-time playback at narrow motion complexity; modeling human motion, fluid, deformable cloth, and articulated objects remains research-grade.
2. **Materials, lighting, and transparency are ill-posed.** 3DGS bakes view-dependent appearance into spherical harmonics; disentangling geometry from materials and lighting from images is mathematically under-determined.
3. **Capacity scales poorly to world-size scenes.** Millions of Gaussians per room scale into the billions for a city; compression techniques (FlashGS, hierarchical GS) help but do not match the implicit compression of a learned generator.

4. **Topology changes are hard.** Adding objects is straightforward; deforming the topology of an existing object (a hand opening, a door becoming a window) is awkward in explicit primitives and is exactly what generative video handles best.

The conclusion mirrors §2.1: explicit 3D is a necessary substrate but not a complete simulator. It needs a surface to render what state alone cannot specify, and it needs *channels* through which the surface can — under controlled conditions — write back into the substrate. The same observation holds in reverse for pure video. The argument of §2 is therefore that no single-tier system is sufficient: only the two-tier paradigm of §4, with explicit gated channels between them, simultaneously satisfies C1 and C2. The remainder of the paper formalizes that paradigm.

3 Industrial Convergence: Two Case Studies

§1 opened with the claim that two leading laboratories have, independently, shipped the same architectural fragmentation. We now examine each in detail. The two cases — World Labs (closed-source, Western, founded by 3D-rendering veterans) and Tencent Hunyuan (open-source, Chinese, an established generative AI lab) — share no common code, no common leadership, and no obvious common pressure beyond the technical objective itself. Their convergence is therefore informative about the structure of the problem rather than about any particular research culture.

3.1 World Labs: Marble + RTFM

World Labs was founded in early 2024 by Fei-Fei Li, Justin Johnson, Ben Mildenhall, and Christoph Lassner. The team composition is itself architectural evidence: Mildenhall is the first author of NeRF [25]; Lassner is the author of Pulsar [18], the differentiable sphere-based renderer that is a direct intellectual ancestor of 3D Gaussian Splatting [17]; Johnson contributed foundational work in neural rendering and visual reasoning. Half the founding team is from the explicit neural-rendering lineage. This is not a video-diffusion company. Fei-Fei Li's published thesis statement — "From Words to Worlds" [20] — argues for "3D or 4D-aware methods for tokenization, context, and memory" and notes that "AI-generated videos often lose coherence after a few seconds."

World Labs ships two products that we read as two halves of an unfinished architecture.

Marble is a 3D Gaussian Splatting generator. It accepts text, single or multiple images, short video, panoramic image, or coarse 3D layouts (via a tool called *Chisel* that decouples scene structure from style) and produces an editable 3DGS or mesh that is downloadable and importable into Blender, Unity, or Vision Pro [40]. Editing is explicit and immediate — Justin Johnson, in his Latent Space interview, describes the workflow as "I can just go in there and grab the 3D block that represents the couch and move it somewhere else." Marble satisfies C1 (immediate editability) and C2 (loop closure) by construction, because both are properties of the geometric data structure. Marble is also, by design, *static*. It contains no time-axis, no dynamics, no actors, no physics. The official blog explicitly emphasizes persistence — "no time limits, no morphing" — and conspicuously avoids any claim about animation.

RTFM is, in World Labs' own description, "an autoregressive diffusion transformer operating on sequences of frames, trained end-to-end on large-scale video data to predict the next frame conditioned on previous frames" [41]. It runs at interactive frame rates on a single H100. It maintains no explicit 3D representation; persistence is achieved by assigning each generated frame a pose in 3D Euclidean space, and at decode time the model attends to the *spatially nearest* past frames rather than the *temporally most recent* — a mechanism the team calls "context juggling." The 3D structure is, in their own phrasing, "a weak prior, without forcing it to explicitly predict

3D geometry." The team explicitly acknowledges that RTFM is static and non-interactive in the agent sense, with dynamic worlds and user interaction listed as future work.

The two products do not share weights, an output format, or a runtime. Marble outputs 3DGS; RTFM outputs frames. Marble is static and editable; RTFM is dynamic and amnesic. The same lab, with the same vision, has shipped the two halves of the architecture we propose — and has not joined them. Three sentences from World Labs' own materials — "AI-generated videos lose coherence after a few seconds," "3D as code: a universal interface for space," and "context juggling: attend to the spatially nearest past frames" — diagnose the problem, prescribe the substrate for state, and prescribe a hand-coded approximation of the fuser, in that order. The architectural primitive missing from this picture is precisely the learned commitment operator SSP names.

We add one further observation that the paper will return to under A2. Marble's substrate is *appearance-3D*: Gaussian opacity, view-dependent spherical harmonics, learned texture — a representation optimized for novel-view synthesis under fixed geometry. It is a substrate in the *renderable* sense, not the *physical* sense. A ball will pass through a Marble wall without protest, because Marble's substrate carries no mass, no contact graph, no Lagrangian. This is not a bug World Labs will patch in a future release; it is the structural consequence of deriving the substrate from a *data-driven* reconstruction objective rather than from a physical-law commitment. The position we will defend is that this path — explicit 3D learned from video, without physical-law structure — is not a stepping stone to a world model; it is a different artifact altogether, and confusing the two is the single largest framing error in the current literature.

A partial symmetry must be acknowledged. The SSP-PMGS substrate of §4.5 also obtains its slot attributes — geometry, material, mass, contact graph — from learning, ultimately from data. The distinction A2 commits to is not at the level of *attribute provenance* but at the level of *update rule*: an A2-compliant substrate, once its attributes are estimated, evolves under structural physical laws regardless of what the training distribution looked like. A Marble Gaussian field, by contrast, has no update rule independent of further rendering; its "evolution" is whatever the next image happens to be. Both pipelines learn from data; only one is structurally committed to physics at runtime. The distinction we draw is sharp at the update-rule layer, blunt at the attribute-acquisition layer, and the position depends on the former, not the latter.

3.2 Tencent Hunyuan: Seven Models in Twelve Months

Tencent Hunyuan presents the same fragmentation in open-source form. Between June 2025 and April 2026, the Hunyuan team released seven world-model-adjacent systems, which we tabulate to expose the pattern:

Model	Output	Mechanism
HunyuanWorld 1.0 [33]	Layered mesh + panorama	Text/image → 3-layer panoramic decomposition → mesh
HunyuanWorld-Voyager [35]	Joint RGBD video	Diffusion produces RGB+depth jointly; reprojected world cache conditions next chunk
Hunyuan-GameCraft 1.0 [30]	Game video	History-conditioned video on 1M AAA clips
HunyuanWorld-Mirror [34]	Multi-view → unified 3D	DUSt3R-style feed-forward reconstruction

Model	Output	Mechanism
HY-World 1.5 / WorldPlay [32]	24 fps streaming video	Dual-action conditioning; "reconstituted context memory"; "context forcing" distillation
Hunyuan-GameCraft-2 [31]	Game video	14B MoE, instruction-injected; **authors admit semantic drift past ~500 frames**
HY-World 2.0 [36]	Persistent 3DGS + mesh	4-stage pipeline: Pano → **WorldNav** → WorldStereo → WorldMirror

The list spans the full 3D–video spectrum from one family. Two observations make this evidence load-bearing for our position.

First, the same lab has built systems that prove each pure approach fails. Hunyuan-GameCraft-2 is the strongest open-source instance of pure-video world modeling, with a 14-billion-parameter mixture-of-experts trained on a million game clips; its own technical report concedes the re-entry wall in plain language. HY-World 2.0 is the strongest open-source instance of pure-3D scene generation; its output is a persistent 3DGS, but the system is offline (four stages, no real-time imagination) and the world is essentially a frozen rendering of the input panorama. Hunyuan has, in effect, shipped its own §2.1 and §2.3 as products.

Second, and more pointedly: HY-World 2.0 contains an explicit module — *WorldNav* — that plans a camera trajectory after the panorama is generated and before the stereo expansion. It is a separate, hand-coded model whose job is to decide *where to look next* and *what to imagine next*. Tencent has, in product form, recognized that the bridge between the 3D state and the generative process requires its own routing logic. They have implemented that logic as fixed engineering rather than as a learned component. WorldNav is the closest existing artifact to the role we assign to the calibrated commitment operator κ — and its existence, as a separately published model in an open-source pipeline, is the clearest available evidence that this slot is a real architectural primitive, not a synthetic abstraction. Tencent has named the slot. They have not yet learned its contents.

World Labs and Tencent, then, instantiate the same diagnosis with opposite resources — closed vs. open, Western vs. Chinese, NeRF lineage vs. diffusion lineage — and arrive at the same fragmentation. Four features recur across both cases.

3.3 The Convergence Pattern

The two cases display the same four features. Each lab independently (i) accepts a diagnosis matching §2.1 and §2.3, (ii) commits to an explicit 3D representation as the persistent substrate for state, (iii) commits to a generative video process for imagination, and (iv) bridges the two with a fixed, non-learned routing rule — either by shipping two disjoint products that never bridge (World Labs) or by inserting a hand-coded intermediate planner (Tencent's WorldNav). The third feature is correct. The fourth is, we argue, the present-day equivalent of the hand-tuned features that the deep-learning revolution replaced in the 2010s. The bridge between substrate and surface is the next natural object to learn — but before we specify how, we owe a precise definition of the paradigm whose bridge it is. The next section makes that paradigm formal.

4 The Substrate-Surface Paradigm: Four Axioms

4.1 The Paradigm in One Sentence

A visual world model is a system that maintains a **substrate** $\mathcal{S}$ — the ontology of what persistently *subsists*, governed by physical laws — and a derived **surface** o — the appearance of $\mathcal{S}$ at a given viewpoint, generated by a (learned) rendering process. The substrate is canonical and editable. The surface is derivable and ephemeral. They are connected by explicit, learned, *event-driven* channels that allow surface content (a user's edit, an observed scene, an imagined room) to write back into the substrate under controlled conditions. We name this commitment the **Substrate-Surface Paradigm**, abbreviated SSP.

SSP is not, by itself, an architecture. It is a definitional commitment — a claim about what *kind of object* a visual world model is. The paradigm yields two consequences that the field has implicitly sought but never named: **immediate editability with loop-closure consistency** (correctness), and **per-frame compute that scales with what has changed in the world rather than with frame rate** (efficiency, in the precise information-theoretic sense developed in §6.5 and §7.5). Both follow from the same four axioms. The remainder of §4 enumerates the axioms, exhibits the violation each current system commits, and specifies one concrete architectural instance — *SSP-PMGS* — that satisfies all four.

4.2 The Four Axioms

A system constitutes a visual world model *in the strong sense* of SSP if and only if it satisfies the following four axioms.

A1 (Ontological Separation). The system maintains two computationally distinct representations: a substrate $\mathcal{S}$ encoding *what exists* (geometry, identity, material, causal slot structure) and a surface o encoding *what is seen* (pixels, view-conditioned appearance, lighting). Neither is reducible to the other — $\mathcal{S}$ cannot be reconstructed losslessly from any finite collection of surface frames, and o cannot be recovered from $\mathcal{S}$ alone without a viewpoint and rendering parameters.

A2 (Physical Substrate). The substrate's evolution under action and time is governed by physical laws — persistence under non-observation, locality of interaction, conservation of mass and momentum, contact non-penetration, monotone entropy — that hold as *structural constraints*, not as soft training-time penalties. In a Platonic reading, the substrate's update rules play the role of *Forms*: universal, observer-independent regularities that any individual scene-instance is required to obey, regardless of what the training distribution happened to depict. The commitment is that for *whatever dynamics class a slot belongs to*, the substrate's update rule encodes that class's law structurally, rather than inheriting it from the statistical distribution of training video. A rigid-body slot is governed by Newtonian rigid-body dynamics; a granular-medium slot is governed by the granular-flow equations; a fluid slot is governed by Navier–Stokes. The ball cannot pass through the wall in $\mathcal{S}$ because the rigid-body Form forbids it, not because such trajectories were absent from training data. Phenomena whose dynamics class is not yet known to admit a clean structural formulation — biological tissue, fracture, articulated topology — are *open within SSP*, not exempt from A2: they are slots awaiting their Form. The paradigm requires the structural commitment; physics (and, by §6.2, JEPA-style pretraining over physical traces) supplies the rules as they become available.

A3 (Generative Surface with Bounded Surface Memory). The surface is a (possibly learned, possibly short-horizon stateful) function $o_t = R(\mathcal{S}_t, \mathcal{C}_t, \tau_t)$ of the substrate, the viewpoint $\mathcal{C}_t$, and optional rendering parameters τ_t (style, atmosphere, lighting overrides). To prevent A3 from collapsing into A1 (a substrate hiding inside the surface), we commit to a ratio bound: the

surface's internal memory horizon must be at most one-half of the loop-closure horizon over which C2 is evaluated. Operationally, a surface refiner whose KV-cache or recurrent state retains enough information to reconstruct the time-invariant scene content over a re-entry interval has absorbed substrate responsibilities, and the system is no longer SSP-compliant. Re-deriving o_t at a previously-visited pose $\mathcal{C}_t$ produces an image consistent with the substrate at that pose, *not* an image whose consistency depends on the surface's elapsed history.

A4 (Gated Bidirectional Channels). Surface content can update the substrate, but the write path is *explicit, gated, and event-driven* — not a default per-frame fusion. SSP requires the existence of at least four channels, each with a distinct semantic role and trigger: (i) **external observation** (sensor input updating $\mathcal{S}$ via reconstruction), (ii) **user edit** (direct mutation of substrate slots), (iii) **imagination commitment** (surface-imagined content promoted to substrate when stable or user-confirmed), and (iv) **agent interaction** (interaction with imagined content forces commitment, because physics requires substrate ground). §5 develops these in detail.

The axioms are not a list of desiderata. They are a logical consequence of demanding, simultaneously, that the system support immediate editability (C1), loop-closure consistency (C2), generative imagination of unseen content, and physical correctness under interaction. We claim — and §4.4 substantiates — that no single-representation system can satisfy all four requirements.

4.3 What the Axioms Forbid

Each axiom rules out a specific architectural class that the current literature instantiates. We tabulate the violations.

System	Violates	Symptom
Sora / Veo / Genie 3 [9]	**A1**: monolithic generator; substrate is implicit in attention	Re-entry wall at ~60s; 8% dynamic-attribute on VBench-2.0
Marble [40]	**A2**: substrate is appearance-3D (Gaussian opacity, view-dependent SH), not physical	Ball passes through wall; no contact, mass, or dynamics
LucidDreamer / WonderWorld [6, 44]	**A2** + **A3**: appearance substrate; surface is a one-shot inpainter, not a stateful refiner	No animation; no consistent re-rendering; no physics
World Labs (Marble + RTFM) [41]	**A4**: substrate + surface exist as disjoint products with no channel	The two never share runtime state
GEN3C / Voyager / WorldWarp [1, 26, 35]	**A4** partial: channel is hand-coded binary mask or noise schedule, not gated by learned commitment	Brittle; cannot generalize beyond hand-tuned mask rules
Hunyuan HY-World + WorldNav [36]	**A4** partial: explicit routing module but engineered, not learned	Tencent has *named the slot*; not yet learned its contents
DreamerV3 [12]	**A1**: substrate-surface collapsed into a flat recurrent latent	Cannot externalize state; not editable; single-agent only

System	Violates	Symptom
V-JEPA 2-AC [3]	**A1** + **A3**: no surface in our sense; predicts in an opaque latent without decoding	Cannot be displayed, edited, or interfaced with engines; latent rollout drift

This is, as far as we know, an exhaustive failure-mode typology of the current literature. No system simultaneously satisfies all four axioms. The omission is not coincidence; it is the empirical signature of a missing paradigm.

4.4 Why All Four Axioms Are Necessary

No single representation can simultaneously satisfy:

(i) **persistence under non-observation** — the chair remains when no one looks (Einstein's moon);
(ii) **editability with deterministic propagation** — moving the chair propagates all geometric, physical, and rendering consequences in the next frame;
(iii) **generative completeness** — the system can imagine the room it has never seen, the texture it has never rendered, the lighting condition that has never been observed;
(iv) **physical correctness under interaction** — when an agent touches an imagined object, contact forces apply and the object cannot pass through walls.

Requirements (i) and (ii) demand an addressable, persistent, structured representation: **A1** + **A2**. Requirement (iii) demands a generative process with unbounded representational reach: **A3**. Requirement (iv) demands that imagined content, once interacted with, becomes physically committed: **A4**, and within A4 the specifically primary role of agent-interaction channel that §5.3 develops.

(i)
- (ii) cannot live in a renderer alone — renderers do not preserve geometric structure under arbitrary edits. (iii) + (iv) cannot live in a substrate alone — substrates do not invent novel content. The two tiers are not a stylistic choice; they are the *minimum sufficient* decomposition for the four-requirement object the paradigm names.

Necessity, however, says nothing about realization. We now exhibit one architecture that meets all four axioms — not to claim it is the only such architecture, but to ground the paradigm in something a researcher can build.

4.5 A Concrete Instance: SSP-PMGS

SSP is a paradigm. We specify one architectural instance, *SSP-PMGS* (Substrate-Surface Persistent-Memory Generative Simulator), and use it throughout the remainder of the paper. A reader uninterested in this specific instance may treat §4.5 through §7 as an existence proof that the paradigm admits implementable systems.

SSP-PMGS is a tuple

$$\mathcal{W} = \langle \mathcal{S}, \mathcal{T}, R, \kappa \rangle$$

with the following commitments. The role names — substrate, transition, render, commitment operator — map one-to-one onto the four axioms.

Substrate $\mathcal{S}_t = (G_t, \Sigma_t)$. An explicit 3D Gaussian-splat field G_t together with a scene graph $\Sigma_t = \{\sigma_t^k\}_{k=1}^N$ of N semantic object slots. Each slot carries the tuple

$$\sigma_t^k \;=\; (\,\text{identity}_k,\ \text{pose}_k \in \text{SE}(3),\ \text{material}_k,\ \text{mass}_k,\ \text{contact-graph}_k,\ \text{confidence}_k\,).$$

Note that the per-slot attributes include *mass* and *contact-graph* — substrate is physical (A2), not merely geometric. Slots are addressable, editable, persistent across frames and re-entries, and renderable through a differentiable rasterizer.

Transition $\mathcal{T} : \mathcal{S}_t \times a_t \to \mathcal{S}_{t+1}$. A composition $\mathcal{T} = \mathcal{T}_{\text{phys}} \circ \mathcal{T}_{\text{latent}}$ of a differentiable physics step (semi-implicit Euler over slot generalized coordinates with LCP contact resolution and a learned residual for unmodeled phenomena) and a V-JEPA-style latent predictor [3] that handles semantic transitions the physics cannot specify (a character changes pose; a category implies likely motion). The physics step satisfies A2 by construction; the latent predictor extends predictive reach to soft, articulated, or category-conditioned dynamics.

Render $R : \mathcal{S}_t \times \mathcal{C}_t \times \tau_t \to o_t$. A two-stage rendering function. Stage 1 emits a deterministic skeleton image $\tilde{o}_t$ from G_t via differentiable rasterization, together with per-pixel rendering uncertainty $u_t^{\mathcal{S}}$ derived from Gaussian opacity, slot confidence, and visibility-since-last-observation. Stage 2 applies a state-conditioned diffusion refiner that fills regions of high rendering uncertainty with plausible content under temporal context, masked by $u_t^{\mathcal{S}}$ such that low-uncertainty regions are not overwritten. The refiner is a short-memory stateful component (its KV-cache spans seconds, not minutes) and satisfies A3.

Commitment operator κ. The mechanism by which surface content writes back to substrate, instantiating A4. We define κ in §5.

A *Scene-as-Memory Bus* organizes the data flow per frame: substrate rasterization → diffusion refinement → output; physics step → substrate update; commitment-gated write-back from surface to substrate via κ. The substrate is canonical: at every time t, $H(o_t \mid \mathcal{S}_t, \mathcal{C}_t, \tau_t)$ is small, and is exactly the irreducible R_d rate from §2.1. Edits to $\mathcal{S}$ propagate eagerly to the next frame's render; the diffusion's KV-cache is allowed to be one frame stale.

4.6 SSP-PMGS Is Not, and Why

We anticipate three confusions and head them off explicitly.

SSP-PMGS is not the Dreamer line. DreamerV3 [12] is the empirically dominant action-conditioned world model. Its state is an implicit, flat, single-agent latent (z_t categorical plus deterministic h_t) trained end-to-end with a policy for return maximization. SSP-PMGS's substrate is explicit, slotted, geometric, persistent across episodes, and shared across agents. Dreamer's encoder, dynamics, and decoder are co-trained for control; SSP-PMGS's components are factored and independently auditable, and the downstream consumer may be a renderer, a robot, an artist, or a multi-agent system rather than a single-task policy. Dreamer is the state of the art for *control*; SSP is a position for *simulation*. The two are orthogonal objects.

SSP is not Active Inference. Friston's active inference [10, 27] formalizes the normative objective of variational free energy minimization, and subsumes POMDP-Bellman control as a limiting case [7]. We make no contribution at the level of inference principle. SSP-PMGS is best read as a concrete, scalable *generative model class* $p(o, \mathcal{S})$ that the active-inference literature has thus far left underspecified (its published instantiations use linear-Gaussian or tabular p). A

reviewer who claims "SSP is just active inference" is correct at the inference-principle level and wrong at the generative-model level — and the latter is where the engineering of this paper lives.

SSP is not Ha and Schmidhuber's World Models. Ha and Schmidhuber [11] introduced the term and the canonical (V, M, C) triple — ConvVAE, MDN-RNN, and a linear controller. SSP extends rather than replaces: our R is a state-conditioned diffusion (where their V was a task-agnostic VAE); our $\mathcal{T}$ is action-conditioned with explicit physics (where their M was a recurrent net they themselves note suffers catastrophic forgetting); our $\mathcal{S}$ is a slotted explicit geometry with mass and contact (where they had no persistent state at all); κ is new. The seven years between the two formulations have moved the field from "fit a recurrent net to game pixels" to "specify the paradigm a queryable persistent simulator must satisfy." Their paper is an instance of the older paradigm; SSP names the new one.

The four axioms are necessary, but they leave A4 — the bidirectional channels — under-specified. A4 declares that channels must exist, must be explicit, and must be gated, but does not say *which* channels or *how* the gating works. §5 supplies that content: four channels, one of them ontologically primary, all mediated by a single learned commitment operator.

5 Channels: How the Surface Writes Back to the Substrate

Axiom A4 requires that substrate updates flow through *explicit, gated, event-driven* channels — not through a default per-frame fusion. This section makes A4 precise. We identify four channels, argue that the fourth (agent interaction) is ontologically primary, and define the commitment operator κ that gates all four.

5.1 Four Channels of Substrate Write-Back

The surface can update the substrate through four distinct channels. Each has its own evidence type, its own confidence model, and its own trigger.

Ch-Sense — External Observation. A camera frame, depth scan, touch sensor, or other external input arrives. The surface module produces a candidate substrate update through a feed-forward geometry head (a VGGT-style network [38] for monocular RGB, a SLAM-style update for depth, etc.). This is the classical state-estimation channel: pixels → features → substrate update with calibrated sensor noise. SSP requires it to exist; no new mathematics is contributed here.

Ch-Edit — User Edit. A user (or external program) directly mutates a substrate slot: `move(chair, Δx)`, `set_material(wall, marble)`, `delete(plant)`. This channel bypasses the surface entirely. Edits are authoritative — they are not gated by uncertainty, only by access control. They propagate deterministically: in the next frame, the rendered surface reflects the new substrate.

Ch-Imag — Imagination Commitment. The surface diffusion process imagines content not present in the substrate — a never-observed room behind a doorway, a never-rendered texture on a back wall, a completion of an occluded object. When the imagination is sufficiently *stable* (consistent across multiple frames of a fixed viewpoint), sufficiently *confident* (low denoising entropy), or sufficiently *user-confirmed* (the user prompted for it), it is committed to the substrate. After commitment, the content is no longer a transient pixel hypothesis; it is a substance with mass, contact, and persistence, indistinguishable in subsequent frames from content that arrived through Ch-Sense or Ch-Edit.

Ch-Act — Agent Interaction. When an agent — embodied or virtual — interacts with imagined content (touches it, occludes it, collides with it, applies a force to it), the content must commit, because physics is over substances. There is no coherent way to compute a contact force

against a pixel hypothesis; the object must be in the substrate, with mass and contact geometry, or the interaction has no defined consequence.

The four channels are not redundant. Ch-Sense grounds the substrate in external reality. Ch-Edit makes the substrate addressable for design. Ch-Imag extends the substrate into imagination — into rooms the user has never built but wants. Ch-Act enforces the operational meaning of "in the substrate": *to subsist is to be acted upon.*

5.2 Why Ch-Act Is Ontologically Primary

The four channels are not symmetric. We claim Ch-Act is *primary* — it is the operational test for substrate-hood that grounds the meaning of the other three.

The argument is Platonic in shape. A2 says the substrate is *what evolves under physical law.* But physics applies to what? Not to pixels — pixels have no mass, no contact, no momentum. Physics applies to things with causal consequences when acted upon. To belong to the substrate is, operationally, to be the kind of thing that can produce physical consequences under interaction; the substrate is the realm of what we may call (following the cave allegory) the *Forms* of the world — the entities behind, not on, the wall on which the surface paints its shadows. A wall in the substrate is a wall an agent can lean against; the load-bearing capacity is what makes it the Form of a wall rather than its shadow.

Ch-Sense, Ch-Edit, and Ch-Imag are channels by which content *becomes available* for interaction. Ch-Act is the channel that *enacts* the interaction. Without Ch-Act, the other three deliver content never tested against the semantics of substrate-hood — a wall imagined into existence via Ch-Imag but never touched is, in operational terms, indistinguishable from a wall painted on a backdrop, which is to say, a shadow. Ch-Act closes the loop: the moment an agent leans against the imagined wall, the wall must have load-bearing physics, or the interaction is undefined.

This is the position paper's *Cogito*-shaped argument, inverted and rerouted through Plato's cave. Not *I think therefore I am*; rather, *that which can be acted upon with physical consequence is what casts the shadow.* The architecture's commitment to Ch-Act as the primary channel is what distinguishes a substrate of Forms from a wall of shadows, and is what makes "physical substrate" (A2) an ontological commitment rather than an aesthetic preference.

5.3 The Commitment Operator κ

All four channels are mediated by a single learned operator that decides what, when, and how to write into the substrate. We define

$$\kappa : (\text{proposal},\ \text{conf},\ \text{intent},\ \text{interaction}) \to \Delta\mathcal{S}$$

where the four inputs encode the candidate content, its source confidence, the user/agent intent signal, and the interaction-trigger flag. The output $\Delta\mathcal{S}$ specifies the substrate mutation: which slots to create, which to modify, what physical attributes (mass, material, contact graph) to instantiate.

κ's gating policy varies by channel. For Ch-Sense (external observation), κ runs continuously over feed-forward geometry-head proposals, accepting updates roughly proportional to the prior substrate uncertainty in the affected region. For Ch-Edit (user edit), κ is a pass-through — the user's intent overrides confidence considerations. For Ch-Imag (imagination commitment), κ accumulates evidence across frames and triggers only when a stability or confidence threshold is met (or when Ch-Act fires). For Ch-Act (agent interaction), κ commits unconditionally — any imagined content under interaction must have physical ground.

In the Gaussian limit (uncorrelated, calibrated per-region variances; proposals from a single channel), κ's acceptance rule reduces to the classical Bayesian update — a Kalman gain on the substrate slot's parameters, $\alpha^\star(r) = \sigma^2_{\text{evidence}}(r)/(\sigma^2_{\mathcal{S}}(r) + \sigma^2_{\text{evidence}}(r))$. This is the floor that any learned κ must beat. The reason we do not stop at the closed form is that real proposals are non-Gaussian and spatially correlated: a diffusion-imagined silhouette has correlated error across its boundary, an external-observation depth map has structured occlusion error, an agent-interaction event has discrete (commit or not) rather than continuous semantics. κ is a learned correction over the Gaussian floor, parameterized as a small network (~10^7 parameters) that consumes the four inputs and emits a per-slot acceptance map.

5.4 The Per-Frame Rendering Path Is Not the Commitment Path

Two operations share the substrate-surface interface, and conflating them is the source of considerable confusion about what κ does. The runtime render — substrate is rasterized to a skeleton image, then refined by a state-conditioned diffusion to fill regions of high rendering uncertainty — happens every frame. The substrate commitment — surface content is promoted to substrate — happens only when one of the four channels fires (typically much less than once per frame). They are not the same operation:

- The **render path** ($\mathcal{S} \to \tilde{o} \to \hat{o}$) is a per-frame, one-way data flow. Stage 1 rasterizes; Stage 2 inpaints under a substrate-derived uncertainty mask that prevents the diffusion from overwriting well-rendered regions. The output $\hat{o}$ is what the user sees. No state is written back from this path by default.
- The **commitment path** ($\hat{o} \to \kappa \to \mathcal{S}$) is event-driven. It fires on Ch-Sense, Ch-Edit, Ch-Imag, or Ch-Act. When it fires, the operator κ writes back into $\mathcal{S}_{t+1}$ based on the channel's evidence, confidence, intent, and interaction signals.

A common simplification treats the slot as a single "fuser" that combines two pixel estimates per frame; this turns out to be a special case of the commitment path restricted to Ch-Imag with a per-frame trigger — useful as intuition (and we retain it as the Gaussian-limit special case in §6.3), but operationally narrower than the slot's full role.

5.5 The Slot, Not the Function, Is the Contribution

A natural reading of the preceding subsections is that we are claiming a specific learnable function κ — a small ResUNet trained on a composite loss — as the paper's central contribution. We are not. The paper's contribution at this slot is normative, not implementational: **we argue that any SSP-compliant world model must contain a learned commitment operator with specific functional properties, and we name that slot**. The specific κ sketched in this paper is one candidate satisfying the properties; we expect others, and the paradigm does not require ours.

The slot's normative properties — the constraints any candidate must satisfy to instantiate A4 — are four:

1. **Learnable, not engineered.** The slot's gating decisions cannot be hand-coded rules (which GEN3C, WorldWarp, and Hunyuan-WorldNav currently are). Hand-coded rules fail to generalize across scene types, lighting conditions, and channel-mixing regimes; the slot must learn from trajectories that span all four channels.

2. **Capacity-bounded relative to the surface refiner.** The slot must be at least two orders of magnitude smaller in parameters than the surface model it gates. This is a *ratio*, not an absolute, and it is what distinguishes an SSP commitment operator from a frontier-model-scale world model wearing a fusion mask. An SSP instance whose slot needs surface-model-scale parameters has collapsed back into a monolithic system with substrate-surface decoration.
3. **Calibrated-uncertainty-conditioned.** The slot's inputs must include explicit per-region uncertainty estimates emitted by the substrate and the surface respectively. Without this, the slot is being asked to *infer* uncertainty rather than *consume* it, and the parameter bound becomes incoherent.
4. **Multi-channel.** A single channel of write-back (e.g., GEN3C's mask, which only handles a degenerate Ch-Sense) is insufficient. The slot must mediate all four channels with distinct gating policies.

The candidate κ specified in §5.4 and §6.3 satisfies all four. Other candidates — non-Bayesian fusion, attention-based gating, learned routing forests, hierarchical commitment operators — may satisfy them differently. The paper's claim is that the slot exists, must be learned, and must satisfy the four properties; the specific function inside the slot is a separate engineering question.

This framing also clarifies the position with respect to the skeptical reading: it is not that κ has been carefully designed to *exclude* world-model knowledge; it is that any function satisfying the four normative properties cannot *contain* world-model knowledge by definition. The capacity ratio (property 2) and the calibrated-uncertainty conditioning (property 3) jointly enforce this. If a future implementation violates either, it has not falsified the paradigm — it has simply built a different object, which may or may not be a useful world model but is not an SSP instance.

5.6 Training Signal

κ is trained on a composite objective whose terms map onto the four channels and onto the C1+C2 requirements:

- **Reconstruction** $\mathcal{L}_{\text{recon}} = \mathbb{E}\|\hat{o}_t - o_t^{\text{gt}}\|_1$ — the rendered output matches held-out views.
- **Loop-closure consistency** $\mathcal{L}_{\text{loop}} = \mathbb{E}_{\gamma:\mathcal{C}_T=\mathcal{C}_0}\|\mathcal{S}_T - \mathcal{S}_0\|_{\text{Chamfer}}$ — trajectories that return to their start do not drift the substrate. Direct optimization target for **C2**.
- **Edit idempotency** $\mathcal{L}_{\text{edit}} = \mathbb{E}\|(\text{edit} \circ \text{undo})(\mathcal{S}) - \mathcal{S}\|$ — the deterministic half of **C1**.
- **Commitment calibration** $\mathcal{L}_{\text{cal}}$ — the post-commitment substrate's predicted uncertainty matches actual reconstruction error after one frame of physics. Without this, κ's acceptance decisions are uninformative.
- **Interaction consistency** $\mathcal{L}_{\text{act}}$ — when an agent acts on imagined content, the resulting physics-step output matches what would have been produced had the content been substrate from the start. This term operationalizes "interaction commits."
- **Anti-collapse** — VICReg-style regularization on $\mathcal{T}_{\text{latent}}$'s embedding [4] plus He Kaiming's dispersive loss [39] in the refiner, to prevent the latent predictor and the refiner from collapsing under their respective objectives.

5.7 Why κ Is the Contribution

Existing systems already have most of the inputs the slot would consume; what they lack is recognition that the slot exists and must be learned. World Labs ships Marble (substrate) and RTFM (surface) and never connects them at runtime [40, 41]. Tencent Hunyuan ships WorldNav, an explicit routing module sitting in approximately the right architectural position, and does not learn it

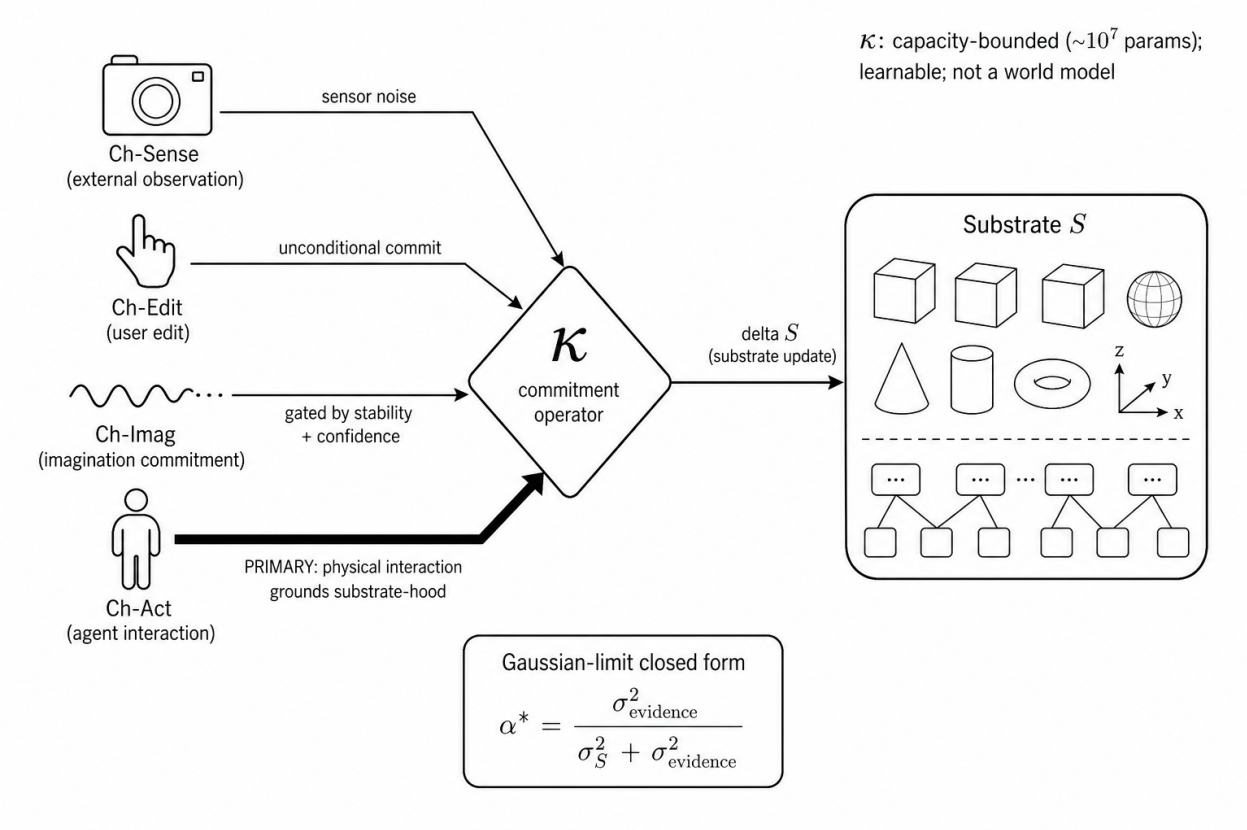

Figure 1: **Figure 2.** The four channels of substrate write-back. *Top:* the render path ($\mathcal{S} \to \tilde{o} \to \hat{o}$) is one-way, per-frame, and does not by itself commit anything. *Bottom:* the four commitment channels are event-driven, all mediated by κ. Ch-Sense (sensor observation) and Ch-Act (agent interaction) ground the substrate in external and physical reality; Ch-Edit (user edit) makes the substrate designable; Ch-Imag (imagination commitment) lets the surface propose new substrate when stable, confident, or user-intended. The Gaussian-limit special case of κ (inverse-variance weighting) recovers a classical Kalman update over substrate slot parameters.

[36]. GEN3C uses a binary mask — "trust the 3D cache where it has data, trust diffusion elsewhere" — a hand-coded degenerate occupant of the slot, lacking calibrated uncertainty and multi-channel coverage [26]. WorldWarp uses a varying-noise schedule that is similarly a hand-tuned, single-channel approximation [1]. Each occupies the architectural slot we are naming but treats it as engineering, not as a learnable object. The paper's contribution is to assert that this slot exists as a distinct architectural primitive, to specify the four normative properties any candidate must satisfy (§5.5), and to argue from physical ontology (§5.2) that one of the slot's channels — Ch-Act — is ontologically primary. The specific Bayesian-fusion candidate of §5.4 is offered as an existence proof that the properties are jointly satisfiable, not as the unique solution.

With the four axioms (§4) and the four channels (§5) in place, we now have everything required to specify the system mathematically. §6 collects the equations; readers comfortable with the conceptual framing may treat it as a reference appendix.

6 Mathematical Formalization

We give the mathematical specification in five equations, organized as two paradigm-level statements (§6.1, §6.5) that any SSP-compliant system must satisfy, and three instance-level statements (§6.2, §6.3, §6.4) that specify the SSP-PMGS candidate of §4.5. The equations are not novel as mathematics — Bayesian update, JEPA energy, RCG factorization, VICReg regularization, and rate-distortion bounds are standard tools — but their composition under the substrate-surface paradigm is. A reader who rejects the SSP-PMGS instance can retain §6.1 and §6.5 as the paradigm's mathematical core and discard the rest.

6.0.1 6.1 Generative Factorization *(paradigm-level)*

Generalizing He Kaiming's representation-conditioned generation [22] from a single image latent to a time-indexed scene state:

$$p(o_{1:T}, \mathcal{S}_{1:T} \mid a_{0:T-1}) = p(\mathcal{S}_0) \prod_{t=1}^{T} \underbrace{p(\mathcal{S}_t \mid \mathcal{S}_{t-1}, a_{t-1})}_{\text{transition } \mathcal{T}} \cdot \underbrace{p(o_t \mid \mathcal{S}_t, \mathcal{C}_t)}_{\text{observation } \mathcal{O}}.$$

The factorization commits to two structural properties: (i) observations are conditionally independent across time given the state, so a re-entry into a previously-visited pose has its observation determined by the state, not by the elapsed time; (ii) the state evolves under a Markovian action-conditioned transition, with no direct pixel-to-pixel dependency. Both properties are violated by current pure-video systems (where o_t depends on $o_{t-k:t-1}$ through the KV-cache rather than on $\mathcal{S}_t$) and are exactly what closes the re-entry wall.

Concretely. A user walks down a corridor for ten seconds, turns around, and walks back. Under the factorization above, the observation at second 11 (camera facing back into the corridor) depends only on $\mathcal{S}_{11}$ — which is essentially the same substrate as $\mathcal{S}_1$ minus a few physics-step micro-updates — and on the camera pose. It does *not* depend on the 240 frames the user has already seen. A pure-video system, by contrast, must reconstruct the corridor at second 11 from its (possibly degraded) memory of frames 1 through 240. The factorization is what makes "the moon is there when no one looks" computationally tractable: the moon's existence is encoded in $\mathcal{S}$, not in the running record of past frames.

6.0.2 6.2 Latent Transition Energy *(SSP-PMGS instance)*

Following V-JEPA 2-AC's action-conditioned latent prediction [3] but with the embedding target being the *explicit* state rather than the future frame:

$$\mathcal{E}_{\mathcal{T}}(\mathcal{S}_{t+1}; \mathcal{S}_t, a_t) = \left\| E_\phi(\mathcal{S}_{t+1}) - P_\psi(E_\phi(\mathcal{S}_t),\, a_t) \right\|_1,$$

where E_ϕ is a learned encoder of $\mathcal{S}$ (slot-level + global geometry) and P_ψ is the action-conditioned predictor. Planning is then a receding-horizon energy minimization over action sequences:

$$\hat{a}_{1:H} = \arg\min_{a_{1:H}} \sum_{h=1}^{H} \left\| P_\psi(E_\phi(\mathcal{S}_t),\, a_{1:h}) - E_\phi(\mathcal{S}^{\text{goal}}) \right\|_1.$$

This is identical in form to V-JEPA 2-AC's planning objective. The crucial difference is that the energy landscape is smoother because E_ϕ operates on explicit slot/geometry coordinates rather than on opaque video tokens, recovering the gradient-based planning regime that LeCun [19] motivated

but did not deliver. In the Platonic reading of A2: this JEPA-style training is precisely the mechanism by which the substrate *learns its Forms* — the universal regularities of how slots evolve under action — from a corpus of physical traces. Pretraining is the materialist on-ramp to an idealist substrate: the substrate's laws originate empirically, but once internalized, they are applied as structural constraints, not as data-distribution echoes.

Concretely. An agent needs to push a chair to a goal location 1.5 meters to its left. The planner samples action sequences (small impulses applied to the chair's slot at each step) and asks P_ψ to predict the resulting future-state embedding. Because E_ϕ encodes a slot whose pose lives in SE(3), the energy varies smoothly with the chair's planned position — a 1 cm change in the action's intended impulse produces a 1 cm change in the predicted slot pose, and a correspondingly tiny change in the energy. Gradient descent on the action sequence converges in tens of steps. The same exercise in a pixel-latent world model would fail: a 1 cm chair shift produces a discontinuous change in the pixel-space rendering (the chair's pixels move discretely), and the gradient signal is noise.

6.0.3 6.3 The Commitment Operator (Bayesian Limit) *(SSP-PMGS instance)*

The central new equation. In the Gaussian limit (uncorrelated, calibrated, per-region variances over slot parameters), a substrate update $\Delta\mathcal{S}(r)$ proposed by any channel of §5 is accepted with the variance-minimizing weight

$$\alpha^\star(r) = \frac{\sigma^2_{\text{evidence}}(r)}{\sigma^2_{\mathcal{S}}(r) + \sigma^2_{\text{evidence}}(r)}, \qquad \mathcal{S}^{\text{new}}(r) = \alpha^\star(r) \cdot \text{proposal}(r) + (1 - \alpha^\star(r)) \cdot \mathcal{S}^{\text{prior}}(r),$$

which is the standard Kalman gain applied per slot. The learned operator κ generalizes this to the non-Gaussian, multi-channel, spatially-correlated case:

$$\Delta\mathcal{S}(r) = g_\kappa\big(\text{proposal}(r),\, \text{conf}(r),\, \text{intent}(r),\, \text{interaction}(r);\, \text{context}(r)\big),$$

with g_κ a small ResUNet-shaped network (~10^7 parameters), and "context(r)" a spatial receptive field around the affected slot. The training objective forces κ to recover $\alpha^\star$ in the Gaussian, single-channel limit and to learn the correction in the multi-channel case. The four channels of §5 correspond to four distinct input combinations: Ch-Sense sets intent and interaction to bottom and conf to the sensor noise level; Ch-Edit sets intent to "user" and conf to infinite, forcing an unconditional commit; Ch-Imag sets conf to the denoising entropy and leaves intent and interaction at bottom; Ch-Act sets interaction to top, forcing an unconditional commit whenever physically required.

Concretely. Walk through three scenes in the same room.

Scene 1 (Ch-Edit: user edit). The user clicks "place a vase on the table." The intent flag is on; the confidence is effectively infinite; the substrate's prior uncertainty in that region is irrelevant. κ outputs $\alpha = 1$ — the vase becomes a substrate slot with default mass and material, immediately renderable from the next frame onward.

Scene 2 (Ch-Imag rejected: confident substrate, hallucinated content). The diffusion refiner, sampling the same region a few seconds later, "imagines" a second vase on the table — it has seen many table-with-vase scenes in training. The denoising entropy is low (high diffusion confidence). But the substrate's prior uncertainty in that exact location is also low (we just placed a real vase there; the slot has $\sigma^2_{\mathcal{S}} \approx 0$). The Kalman formula gives $\alpha = \sigma^2_{\mathcal{O}}/(0 + \sigma^2_{\mathcal{O}}) = 1$ — the substrate wins, and the imagined second vase is rejected. The rendered output continues to show only the real vase.

Scene 3 (Ch-Imag accepted: doorway behind the wall). The user walks toward a doorway the substrate has never reconstructed. The substrate's rendering of that region is empty: $\sigma_{\mathcal{S}}^2 \to \infty$. The diffusion refiner produces a plausible interior room visible through the doorway with $\sigma_{\mathcal{O}}^2$ moderate. The Kalman formula gives $\alpha \to 0$ — the substrate cedes, and the diffusion's content is admitted. After three consistent frames of the imagined room from different sub-viewpoints, κ's stability gate triggers and the room commits: a new set of substrate slots appears, complete with the materials and approximate geometry the refiner produced.

The same closed-form operator handles all three cases without any conditional logic. Channel-specific behavior is encoded entirely in which input fields are populated.

6.0.4 6.4 The Composite Loss *(SSP-PMGS instance)*

The full training objective is

$$\begin{aligned}\mathcal{L} \;=\; &\underbrace{\mathbb{E}\|\hat{o}_t - o_t^{\mathrm{gt}}\|_1}_{\text{render recon.}} + \lambda_1 \underbrace{\mathcal{E}_{\mathcal{T}}}_{\text{transition}} + \lambda_2 \underbrace{\mathcal{L}_{\text{loop}}}_{\text{C2}} + \lambda_3 \underbrace{\mathcal{L}_{\text{edit}}}_{\text{C1}} \\ &+ \lambda_4 \underbrace{\mathcal{L}_{\text{cal}}}_{\text{calibration}} + \lambda_5 \underbrace{\mathcal{L}_{\text{act}}}_{\text{interaction}} + \lambda_6 \underbrace{\mathcal{L}_{\text{vic+disp}}}_{\text{anti-collapse}},\end{aligned}$$

with all six auxiliary terms required, not optional. The loop and edit losses directly target the C1/C2 requirements. The calibration loss ensures κ's inputs carry the information it is asked to weight. The $\mathcal{L}_{\text{act}}$ consistency loss enforces that imagined content committed under interaction behaves physically. The anti-collapse term prevents the latent transition and the diffusion refiner from collapsing under their respective objectives.

Concretely, which terms dominate when. Consider the corridor-revisit trajectory from §6.1. On the initial walk-out (frames 1–240), the render-reconstruction term $\|\hat{o}_t - o_t^{\mathrm{gt}}\|_1$ dominates: the system is learning to render what it sees, gradients flow primarily into R and into substrate-slot updates via Ch-Sense. On the return walk (frames 241–480), if the substrate has been correctly committed, the reconstruction term is small (the rendered output matches ground truth by construction) — but the loop term $\mathcal{L}_{\text{loop}}$ becomes active, measuring the Chamfer drift between $\mathcal{S}_{240}$ and $\mathcal{S}_{480}$. If $\mathcal{L}_{\text{loop}}$ is large here, the gradient flows into κ's commitment decisions during the walk-out and into $\mathcal{T}$'s physics step. On a separate edit-and-undo trajectory, the reconstruction and loop terms are zero by construction; $\mathcal{L}_{\text{edit}}$ dominates and trains the substrate's mutation operators. On a Ch-Act trajectory (an imagined object that gets touched), the $\mathcal{L}_{\text{act}}$ term is the only non-zero gradient signal: it teaches κ when an interaction event must trigger commitment. Each loss term targets a distinct trajectory class. Joint training on a mixture of all four trajectory types is what gives κ its multi-channel competence without ever exposing κ to a single "do everything" objective.

6.0.5 6.5 Rate-Distortion Lower Bound *(paradigm-level)*

Finally, the information-theoretic floor that any SSP-compliant simulator must respect. Let R_g denote the rate of the time-invariant substrate component and R_d the rate of the time-varying surface component. For a system that stores the substrate explicitly and rasterizes from it, the per-frame rate is $R_d + O(R_g/T)$. For a system that violates A1 (no explicit substrate), the per-frame rate is bounded below by the conditional entropy $H(o_t \mid \text{context})$, which is empirically $\approx R_g + R_d$ for all current systems with context windows shorter than the re-entry horizon. The asymptotic compute ratio between the two regimes is the 20× gap of §2.1. The lower bound is tight in the SSP-compliant regime and loose by exactly the redundancy factor in the A1-violating regime. This bound is simultaneously the formal statement of the position and the formal statement of the

efficiency advantage: *any system that meets C1 and C2 within a bounded compute budget must, in the limit of long horizons, satisfy A1 and store R_g explicitly.* There is no representation-learning regime that escapes this floor.

Concretely. The key ratio is R_d/R_g. In typical interactive trajectories — a user walking through a furnished scene, occasional object motion, varying lighting — R_d per frame is dominated by camera pose updates and small dynamic perturbations and is *empirically several orders of magnitude smaller than R_g measured per frame*, where R_g is the geometry, material, and identity content of the scene. (We do not commit to specific bit counts here, which depend on scene complexity and compression assumptions; the load-bearing claim is the order-of-magnitude separation, which is well-attested across indoor-scene benchmarks and compressed-video bit-rate studies.) Under any such regime, the implicit-substrate code's $\Theta(T \cdot R_g)$ rate diverges from the explicit-substrate code's $\Theta(R_g + T \cdot R_d)$ rate by a factor that grows linearly in horizon length. For horizons of minutes, the ratio is in the orders, not in the percents. This is not a $3\times$ engineering inefficiency; it is the entire reason a video diffusion model cannot, even in principle, scale to long horizons under bounded compute. The substrate-surface separation is the only way to escape a $\Theta(T \cdot R_g)$ rate.

The bound closes the §6 mathematical specification. Equations 6.1–6.4 describe how the system computes; equation 6.5 describes why it must be organized this way at all. The mathematics is now in place. The next two questions — *can it run in real time?* and *what would prove this paradigm wrong?* — are engineering and epistemic, and form §7 and §8 respectively.

7 Engineering and Compute-Optimal Allocation

A position paper that does not engage with deployment is a manifesto, not a proposal. We address C1's latency requirement (under 100 ms edit-to-visible) and the framework's per-frame compute envelope directly.

A note on the status of this section. The numbers, parameter counts, latency tables, and compute splits below are *illustrative budgets under the SSP-PMGS instance of §4.5*, not committed measurements. They serve two purposes: to demonstrate that the paradigm admits a viable engineering envelope on consumer hardware, and to expose the specific quantities a future reference implementation would have to deliver on. They do not constitute empirical claims and should not be cited as such. The paradigm's load-bearing predictions are in §8; the engineering numbers below are conditional on the SSP-PMGS choices and on the distillation/parameter assumptions stated. A reader skeptical of the specific instance can treat §7 as an existence proof of feasibility, not as a benchmark.

7.1 Per-Frame Latency Budget

Targeting 30 fps streaming with C1 satisfied on consumer hardware (24 GB VRAM, ~40 TFLOPS sustained):

Stage	Operation	Budget
Edit ingest (Ch-Edit)	Mutate $\mathcal{S}_t$ via slot handle; sparse CRDT-style log	2 ms (CPU)
Physics ($\mathcal{T}_{\text{phys}}$)	Semi-implicit Euler + LCP contact + learned residual	5 ms (GPU)
Read $\mathcal{S}_t$ (render)	3DGS rasterization at 1080p, emits $\tilde{o}_t + u_t^{\mathcal{S}}$	12 ms (GPU)

Stage	Operation	Budget
Diffusion refiner	1-step distilled, conditioned on $\tilde{o}_t$ and $u_t^{\mathcal{S}}$	8 ms (GPU, INT8)
Commitment operator κ	Small ResUNet over channel-specific inputs; runs only when a channel fires	3 ms (GPU, amortized)
Substrate write	Sparse slot update via κ's acceptance map	$\leq$3 ms (amortized, mostly Ch-Sense / Ch-Imag)
Total		**~33 ms**

Edit-to-visible latency is 2 ms (mutate) + 12 ms (read on next frame) = 14 ms, well under the 100 ms target.

The budget is tight in three places: the rasterization assumes Gaussian count under 50 million (~2 GB VRAM); the distilled diffusion assumes a 1-step refiner with at most a 2-FID degradation from a 50-step teacher (which is the central engineering risk — see §8); and the physics step is fast only for rigid bodies and small particle counts (fluid/cloth dynamics blow the budget and must be batched).

7.2 Compute Reallocation Under the Paradigm

The Chinchilla-style implication of the paradigm is that fixed per-frame compute budget should shift from observation (which currently absorbs ~95% in Sora-class systems, regenerating geometry every frame) toward substrate maintenance and commitment. A reasonable illustrative split for SSP-PMGS — substrate ~40%, transition ~25%, render+refine ~25%, commitment ~10% — is the inverse of the current allocation. The 40% substrate share is not an optimization; it is where the bulk of the bits structurally live (R_g from §6.5). The 10% commitment share reflects κ's event-driven invocation rate (channels fire less than once per frame on stable trajectories), not its per-call cost. The relocation is the point; the specific percentages are illustrative.

7.3 Axioms $\neq$ Hand-Engineered Features

We anticipate an objection in the spirit of the Bitter Lesson [29]: any architecture composed of hand-designed modules and explicit data structures will, in the long run, lose to a sufficiently scaled end-to-end neural network. We accept the spirit and reject the letter. The Bitter Lesson is about learned *features* outperforming hand-crafted *features*. It is not about learned *interfaces* outperforming standardized *interfaces*: TCP/IP was not learned, and yet every learned network on the internet runs over it. SSP commits to standardized *axioms* — substrate-surface separation, physical substrate dynamics, gated channels — and to learned *contents* at every functional role. The four axioms describe the *shape* of the system, not its parameters; the parameters are learned end-to-end within that shape. This is a different commitment from hand-crafted features, and the empirical bet is that axioms, like protocols, are the part of the system that benefits from being human-readable and stable across scale.

7.4 The Engineering Expectation

Putting the latency budget and the compute reallocation together, an SSP-PMGS reference implementation at the parameter counts illustrated in §7.1 (substrate $\sim 10^7$ Gaussians, transition $\sim$ 300M, refiner $\sim$ 1B, commitment operator at least two orders of magnitude smaller than the

refiner) should sustain 30 fps streaming on a single consumer RTX-class GPU with edit-to-visible latency under 50 ms. Per §7's opening disclaimer, this is an *expectation* about a specific instance, not a paradigm-level commitment: a reference implementation that honors the instance's structural choices and misses this target by more than 3× would refute the SSP-PMGS instance, but the paradigm survives because §4.5 was always one architectural candidate among possible others.

7.5 Efficiency and Streaming as a Consequence of the Paradigm

A common misreading of factored architectures is that the factoring trades efficiency for clarity — the substrate and surface are separated for *legibility*, and the system pays a coordination tax. We argue the opposite. SSP's substrate-surface separation is the source of its efficiency. The commitment operator κ does not just gate substrate write-back; together with the explicit uncertainty channels, it dictates *what information has to cross the substrate-surface boundary in the first place*, and the answer is far less than current end-to-end systems transmit. This section makes that argument precise.

7.5.1 The Minimal-Condition Principle The next frame's rendered output $\hat{o}_{t+1}$ is a function of $\mathcal{S}_{t+1}$, the camera $\mathcal{C}_{t+1}$, and the rendering parameters τ_{t+1}. By the chain rule of conditional dependence, $\hat{o}_{t+1}$ changes only when at least one of these changes — and $\mathcal{S}_{t+1}$ changes only when one of the four commitment channels of §5 has fired between t and $t+1$. The minimal information that must cross the substrate-surface boundary per frame is therefore:

the change in the substrate inputs that the render path actually consumes.

Concretely, between consecutive frames at times t and $t+1$:

- If the camera has not moved and no commitment channel has fired, $\mathcal{S}_{t+1} = \mathcal{S}_t$ and $\mathcal{C}_{t+1} = \mathcal{C}_t$. *The rendered $\tilde{o}_{t+1}$ is bitwise identical to $\tilde{o}_t$ — the rasterizer can be skipped.*
- If the substrate has not changed and the refiner is at temperature zero, $\hat{o}_{t+1} = \hat{o}_t$. *The refiner can be skipped.*
- If both hold, the system's only per-frame work is updating its clocks. The entire stack reduces to a memory read.

This is a stronger efficiency claim than "skip the diffusion when the scene is static." It is a derived consequence of A1 (substrate-surface separation): the only work the system must do per frame is the work whose output the render path would consume differently. Everything else is, in a precise information-theoretic sense, surplus.

7.5.2 The Δ-Packet Protocol The principle suggests a concrete transmission protocol between substrate and surface. At each frame, the two tiers emit *delta packets* containing only what has changed since the last transmission:

Packet	Source	Payload	Triggered when
$\Delta\mathcal{S}_t$	$\mathcal{S} \to R$	Region-tagged Gaussian/slot updates since last frame	Edit, transition, commitment channel fired
$\Delta\mathcal{C}_t$	viewpoint $\to R$	Camera pose delta	Camera moved

Packet	Source	Payload	Triggered when
$\Delta\hat{o}_t$	$R \to$ display	Patches where the rendered output has revised	$\Delta\mathcal{S}_t$ or $\Delta\mathcal{C}_t$ non-empty
Channel events	$\to \kappa$	C1/C2/C3/C4 trigger packets	Sensor input, user edit, stable imagination, or interaction

A stationary camera over an unedited scene transmits zero bytes after the first frame. A camera panning across a previously-mapped region transmits only the newly-visible strip — a fraction of full-frame bandwidth proportional to the panning rate. An edit (C2) propagates only to the affected slot's tile. The intuition matches event cameras in neuromorphic vision: most pixels do not change between frames, so most should not be retransmitted. The novelty here is that the trigger is not a per-pixel intensity threshold but an *axiom-consistent substrate change*: a pixel intensity change that does not correspond to a real substrate or viewpoint update is not propagated because no such change can have occurred.

Figure 3 illustrates the packet flow under a typical mixed trajectory: most frames carry near-empty packets; bandwidth spikes only at channel-fired events and at the moments the camera enters new territory.

7.5.3 Concrete Architectures That Realize the Protocol Three architectural primitives, all already in production use for adjacent problems, suffice to implement the Δ-packet protocol without further research:

1. Streaming KV-cache with locality. The diffusion refiner runs as a continuous attention process whose KV-cache is *spatially* indexed rather than *temporally* — tiles enter the cache when first observed and are evicted only when they leave the visible frustum. This is structurally similar to GEN3C's 3D point-cloud cache [26] but with a learned per-tile retention policy. Bandwidth cost: O(newly-visible tiles), not O(full frame).

2. State-space models for the transition predictor. The latent transition $\mathcal{T}$ benefits from an SSM-style architecture (Mamba, RWKV, S4) over the slot sequence rather than a transformer: SSMs have constant per-token state and naturally support the "frame-rate-adaptive" computation the protocol requires. When the slot dynamics are static, the SSM trivially passes through; when an edit or physics step changes a slot, only that slot's hidden state advances.

3. Event-driven invocation of κ. The commitment operator runs only when one of its four channels fires. Ch-Sense fires on each new sensor frame; Ch-Edit fires on user edit; Ch-Imag fires when the refiner's denoising entropy crosses a learned threshold within a region of high substrate uncertainty; Ch-Act fires when the physics engine reports a contact involving uncommitted content. The trigger thresholds are learned, not hand-tuned, and couple to the same Bayesian arithmetic that defines κ.

All three primitives share a common signature: their compute scales with the *change* in the world, not with the wall-clock frame rate. On a static camera over a static scene, the entire stack reduces asymptotically to a memory read. On a fully novel trajectory through never-seen geometry, it reduces to the worst-case rate of a video diffusion model — but no current system beats this worst case either. The asymmetry is the gain.

7.5.4 Quantitative Consequences We can bound the savings analytically. Let $\beta \in [0, 1]$ be the fraction of frame area that has a non-negligible Δ-packet at a given frame (the Δ-*occupancy*). For typical first-person camera trajectories — head motion, slow pans, occasional edits — empirical motion-vector analyses on existing datasets put β in the range $0.1 \leq \beta \leq 0.3$. The SSP-PMGS per-frame cost then drops from the §7.2 baseline by a factor of roughly β on the diffusion-refiner and render stages (which dominate the budget), corresponding to a 3×–10× streaming throughput gain on the same hardware over the §7.1 numbers. On a fully static camera ($\beta \to 0$), the gain is unbounded modulo the once-per-frame rasterization, which is itself sub-millisecond on consumer GPUs.

A complementary corollary applies to multi-agent settings. Two agents observing the same scene from different viewpoints share the *same* canonical substrate $\mathcal{S}_t$; their viewpoint deltas $\Delta\mathcal{C}$ are computed at each viewpoint, but $\mathcal{S}$ itself is transmitted once. The protocol's asymmetric trust law — canonical substrate, derived surface — is also the basis for asymmetric communication cost: $|\mathcal{S}|$ scales with the world, while $|\Delta o|$ scales with the *changes* each agent sees.

7.5.5 The Connection Back to the Paradigm The reason this works is that A1 (substrate-surface separation) gives the system a *legible* answer to the question "when does the next frame's output depend on this frame's computation?" In an end-to-end video model, this question is unanswerable — every pixel of every frame is in principle a function of every preceding pixel through opaque attention weights, and the model has no internal handle by which to skip computation safely. In SSP, the handle is the substrate-surface boundary: if the substrate has not changed and the viewpoint has not moved, the surface has not changed. Efficiency follows from the same axiom that gave the system loop-closure consistency and edit determinism. The factoring is not a tax. It is the source of the savings.

With both correctness (§5–§6) and efficiency (§7) accounted for, the paradigm has discharged its constructive work. The remaining duty of a position paper is to make itself refutable. §8 states what would prove SSP wrong.

8 Predictions, Limitations, Open Problems

A position paper without falsifiable predictions is rhetoric. We state ours.

8.1 Predictions

P1 (A1 is necessary, at any scale). By the end of 2027, no system that violates A1 — i.e., that maintains no explicit substrate representation distinct from its surface — will achieve under 5% Chamfer drift on a standardized 60-second loop-closure benchmark, *regardless of parameter count, training-data scale, attention variant, or context-window length.* We deliberately remove the parameter cutoff from a previous draft: the rate-distortion argument of §2.1 does not depend on parameter count, and stating a specific cutoff (1T, 10T) would gerrymander the prediction. The claim is that the redundancy cost of re-emitting R_g each frame is structural, not parametric. If a pure-video system at *any* scale passes the benchmark, A1 is unnecessary and the rate-distortion argument is wrong.

P2 (A2 separates winners from losers). Among systems that satisfy A1, those that violate A2 — substrate is appearance-3D without physical dynamics — will fail physics-correctness benchmarks (object-permanence under contact, conservation under interaction) at a rate at least 3× higher than A2-compliant systems. The empirical line is sharp: a Gaussian field without mass and contact graph cannot pass an object-permanence-under-contact test, regardless of training scale.

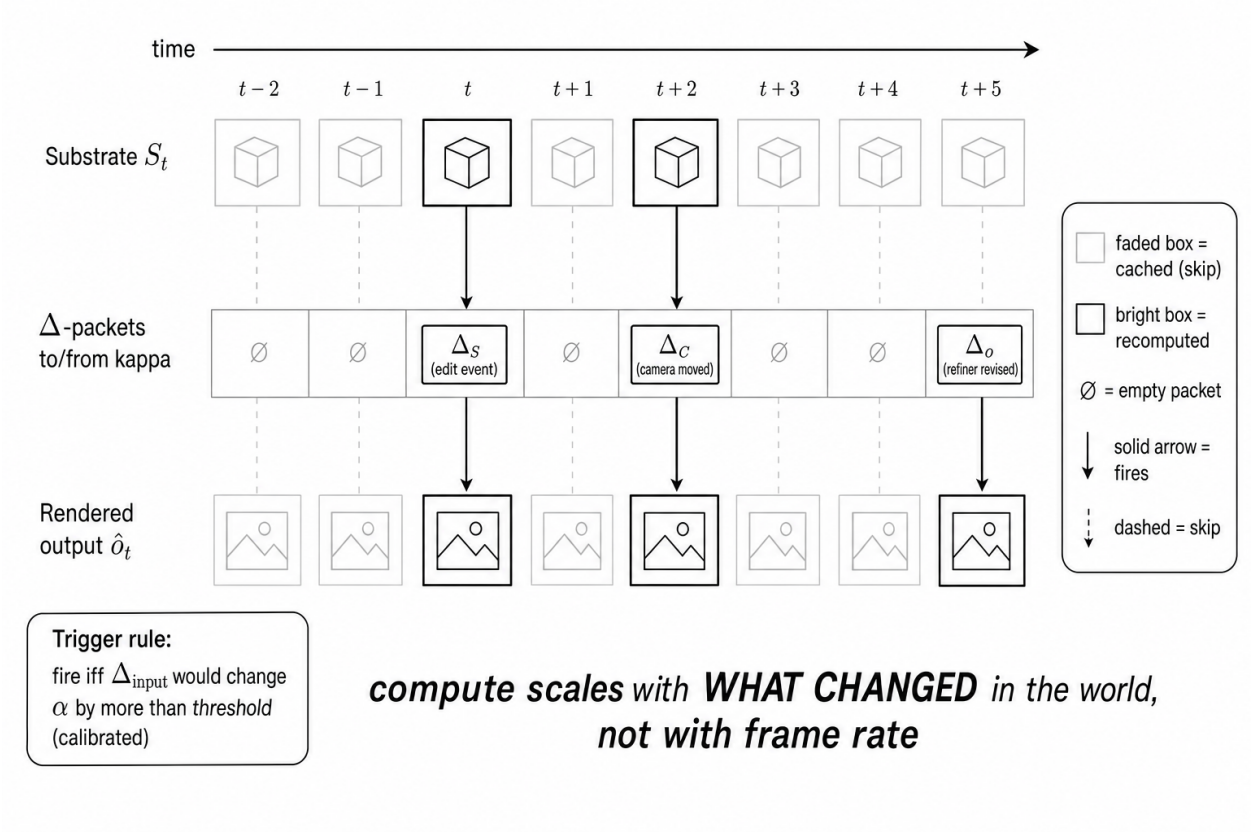

Figure 2: **Figure 3.** The Δ-packet streaming protocol. *Top:* the video diffusion track $(o_t, \sigma_{\mathcal{O}})$; most time steps are cached (faded), recomputation only occurs at moments when uncertainty actually changes. *Bottom:* the 3D scene state track $(\mathcal{S}_t, \sigma_{\mathcal{S}})$, similarly mostly cached. *Middle:* per-frame packets between the substrates. Most are empty ($\emptyset$ skip). A Δu-packet fires at t (uncertainty change from a new visible region); a Δs-packet fires at $t{+}3$ (an edit event). *Inset:* the trigger rule that decides packet vs. skip is the same Bayesian arithmetic that defines κ. Inter-substrate bandwidth is dictated by substrate-changes that drive surface-changes, not by frame rate.

P3 (The winning architecture will be SSP-shaped). The architecture that leads controllable-simulator benchmarks at the end of 2027 will satisfy all four axioms. It will not be end-to-end trained as a monolithic neural network. It will exhibit (i) an explicit, queryable substrate, (ii) physical update rules over that substrate, (iii) a learned rendering function, and (iv) at least one learned commitment channel. If the leader satisfies fewer than three of the four axioms, our central claim is falsified.

P4 (The commitment operator becomes a named primitive). Within 24 months, at least three independent research groups will publish architectures that include a *learned* operator between substrate and surface, mediating multiple write-back channels. Naming will vary across groups, but the functional role and its multi-channel structure will recur. If no such convergence occurs, the slot we are naming is either unnecessary or wrong.

P5 (Literature converges on the learned slot). Within 24 months, at least five distinct peer-reviewed publications will propose architectures whose bridge between an explicit substrate representation and a generative surface is *trained end-to-end with the substrate and surface modules* rather than hand-coded as a routing rule, schedule, or mask. The criterion is whether the bridge's parameters appear in the optimizer's parameter list — a yes/no test, not an interpretive one. We deliberately frame this as a *literature* prediction rather than a *commercial* one — World Labs may keep Marble and RTFM disjoint for business reasons unrelated to architecture, and we will not let that contingency falsify a technical thesis. If the literature does not converge on a learned bridge in this window, our claim that this slot is the next natural object to learn is incorrect.

8.2 Honest Limitations

We do not claim SSP is the right paradigm for filmmaker-grade video generation (objective (1) of §1.2). We do not claim it is the right paradigm for single-agent return maximization on novel tasks (objective (3) — DreamerV3 [12] occupies this slot more efficiently). We do not claim SSP solves the open problems of fluid dynamics in physical substrates, articulated topology, long-horizon causal reasoning, or language-grounded action. We do not yet provide a prototype of SSP-PMGS with empirical validation of C1 + C2 against current systems, and the absence of that prototype is the largest honest weakness of this paper. We commit to releasing the reference implementation referenced in §7.4 within twelve months. If the implementation fails, the position remains — SSP is a paradigm, not a particular system — but our credibility in §7's specific numbers becomes manifesto.

We further acknowledge structural limitations of each tier, which any SSP instance inherits. The substrate must include physical attributes (mass, contact, material) — appearance-3D alone violates A2 — and obtaining those attributes from monocular video is an open inverse problem. The surface is subject to all known failure modes of current video diffusion, contained but not eliminated by the substrate-conditioned masking. The commitment operator's training requires trajectories with revisits, edits, and interaction events; this dataset construction is non-trivial and may need to be procedurally generated, which raises sim-to-real concerns. The architectural bet is that these failure modes are smaller, more isolated, and more diagnosable when the four axioms factor them than when they are tangled in a monolithic network. The bet is empirical and is what P1–P5 test.

8.3 Open Problems

Several problems are outside the scope of this paper but determine the long-run viability of the paradigm.

(i) **Slot discovery.** SSP-PMGS assumes a semantically slotted scene graph Σ_t. How that decomposition is discovered from monocular video remains open; slot-attention methods [24] and feed-forward geometric foundation models like VGGT [38] are promising components, but neither produces causal, persistent, manipulable slots end-to-end.

(ii) **Soft and biological matter.** Hair, cloth, water, foliage, and biological tissue do not admit clean slot decomposition; they live closer to the diffusion surface's natural territory. The current paradigm relegates them to "regions of permanently high substrate uncertainty" — a clean concession that lets the surface fill them per frame but defers the question of how (or whether) they ever commit to substrate. Ch-Act supplies a partial answer (commit on interaction), but interaction with cloth is itself research-grade physics.

(iii) **Multi-agent simulation.** The paradigm is single-substrate-shared; running multiple agents in $\mathcal{S}$ requires concurrency protocols (whose action commits first, how conflicting writes resolve, how partial observability is enforced). Each agent's surface is independent, but the substrate is shared. This is an active-inference [27] direction we touch but do not develop.

(iv) **Generative novelty.** A substrate-conditioned refiner has fewer degrees of freedom than a free-running video diffusion. Whether the loss in creative range is acceptable for the controllable-simulator objective is empirical; we expect it is — most edits and most novel content can be expressed as substrate updates plus rendering parameters τ — but we do not prove it. The Ch-Imag channel is the explicit mechanism by which novelty enters substrate, and its training is what determines the system's creative ceiling.

(v) **Commitment training data.** The commitment operator κ needs trajectories that revisit (for C2-style loop closure), edits (for C1 idempotency), stable imaginations across viewpoints (for Ch-Imag), and interaction events (for Ch-Act). Constructing such datasets at scale is non-trivial; procedural generation in simulators and replay from human-in-the-loop demonstrations are the two known approaches, both with sim-to-real and coverage concerns.

(vi) **The status of τ.** SSP's rendering function takes optional parameters τ (style, lighting, atmospheric override). Whether these are first-class substrate attributes, transient rendering parameters, or some intermediate category is unsettled. We have implicitly treated them as rendering parameters that do not participate in κ's commitment decisions. A reviewer who pushes on this might be right; we mark it as open.

9 Conclusion

Plato's prisoners, chained in the cave, mistook the shadows on the wall for the world. The current generation of visual world models has, on its current trajectory, done something similar: it has scaled the cave-wall painting to remarkable fidelity, and asked us to call the painting a world. Einstein objected: *the moon is there even when no one looks.* So did Plato: *the truth is not the shadow.* The diagnosis is now widely shared in the field even when its philosophical roots are not: pure video is not enough, and pure 3D is not enough. The architectures that have followed the diagnosis — Marble alongside RTFM, HY-World alongside GameCraft, GEN3C's binary mask, WorldWarp's noise schedule — have committed the same residual error: they bridge what *is* and what *appears* with fixed routing rules, leaving the most important slot in the architecture engineered instead of learned.

This paper offers a paradigm, not a system. We argue that a visual world model — in the strong sense the field is now reaching for — must commit to four axioms: an ontological separation between substrate and surface (A1), a physically-governed substrate (A2), a generative surface with bounded short-horizon memory (A3), and explicit, gated, learned bidirectional channels between them (A4). We name this the Substrate-Surface Paradigm. We argue, with a violation table for the current literature, that no existing system satisfies all four. We argue, from a physical-ontological reading of Plato's cave allegory, that the fourth axiom is grounded in interaction: a thing belongs to the substrate — participates in a Form, in the Platonic sense — when and because it can be acted upon with physical consequence. We specify one concrete instance, SSP-PMGS, that meets all four axioms within a 33 ms per-frame illustrative budget on consumer hardware, and admits a $3\times$–$10\times$ streaming efficiency multiplier as a derived consequence of the substrate-surface separation rather than a bolted-on optimization.

We commit to five falsifiable predictions, each pinned to a specific axiom or to the convergence story. We commit to a reference implementation within twelve months. If the predictions fail, the paradigm is wrong, and we will say so. If the implementation fails, the specific instance is wrong, and the paradigm survives. Either outcome is informative.

The paper closes where it opened. *The form and the shadow are not the same.* What is real in a world model is not what is rendered on the cave wall; it is what casts the shadow — the substrate whose laws produce the projection, whose state outlives any single observation, and whose existence is operationally established by the agent's interaction. A system that traffics only in shadows — pixels conditioned on pixels — is not a world model in disguise; it is the cave wall, lit by training data. The next generation will turn around. We hope to have helped name the direction.

References

[1] Anonymous. Worldwarp: Spatio-temporal fill-and-revise for online 3dgs world models, 2025.

[2] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *CVPR*, 2023.

[3] Mahmoud Assran, Adrien Bardes, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. 2025.

[4] Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *ICLR*, 2022.

[5] Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. V-jepa: Revisiting feature prediction for learning visual representations from video. 2024.

[6] Jaeyoung Chung et al. Luciddreamer: Domain-free generation of 3d gaussian splatting scenes, 2024. URL `https://luciddreamer-cvlab.github.io/`.

[7] Lancelot Da Costa, Noor Sajid, Thomas Parr, Karl Friston, and Ryan Smith. Reward maximisation through discrete active inference. *Neural Computation*, 2023.

[8] Decart and Etched. Oasis: A universe in a transformer, 2024. URL `https://oasis-model.github.io/`.

[9] Google DeepMind. Genie 3: A new frontier for world models. Google DeepMind blog, 2025. URL https://deepmind.google/blog/genie-3-a-new-frontier-for-world-models/.

[10] Karl Friston. The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, 11:127–138, 2010. doi: 10.1038/nrn2787.

[11] David Ha and Jürgen Schmidhuber. World models. 2018. NeurIPS 2018, oral as "Recurrent World Models Facilitate Policy Evolution".

[12] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640:647–653, 2025. doi: 10.1038/s41586-025-08744-2.

[13] Jordan Hoffmann et al. Training compute-optimal large language models, 2022.

[14] Ziqi Huang et al. Vbench-2.0: Advancing video generation benchmark suite for intrinsic faithfulness, 2025.

[15] Bingyi Kang et al. How far is video generation from world model: A physical law perspective, 2025. ICML 2025.

[16] Andrej Karpathy. On the confusion of "world models". Public commentary, X / Twitter, 2024. Personal communications and posts.

[17] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. In *SIGGRAPH*, 2023.

[18] Christoph Lassner and Michael Zollhöfer. Pulsar: Efficient sphere-based neural rendering. In *CVPR*, 2021.

[19] Yann LeCun. A path towards autonomous machine intelligence (v0.9.2). OpenReview, 2022. URL https://openreview.net/forum?id=BZ5a1r-kVsf. Position paper.

[20] Fei-Fei Li. From words to worlds: Spatial intelligence. Substack essay, 2025. URL https://drfeifei.substack.com/p/from-words-to-worlds-spatial-intelligence.

[21] Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise, 2026. CVPR 2026.

[22] Tianhong Li, Dina Katabi, and Kaiming He. Return of unconditional generation: A self-supervised representation generation method. In *NeurIPS*, 2024.

[23] Zhan Li et al. Spacetime gaussian feature splatting for real-time dynamic view synthesis, 2024. SIGGRAPH 2024.

[24] Francesco Locatello et al. Object-centric learning with slot attention. In *NeurIPS*, 2020.

[25] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020.

[26] NVIDIA Research. Gen3c: 3d-informed world-consistent video generation with precise camera control, 2025. CVPR 2025 Highlight.

[27] Thomas Parr, Giovanni Pezzulo, and Karl J. Friston. *Active Inference: The Free Energy Principle in Mind, Brain, and Behavior*. MIT Press, 2022. ISBN 9780262045353.

[28] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, and David Silver. Mastering atari, go, chess and shogi by planning with a learned model. volume 588, pages 604–609, 2020.

[29] Richard S. Sutton. The bitter lesson. Personal blog, 2019. URL http://www.incompleteideas.net/IncIdeas/BitterLesson.html.

[30] Tencent Hunyuan Team. Hunyuan-gamecraft 1.0: History-conditioned interactive game video generation, 2025.

[31] Tencent Hunyuan Team. Hunyuan-gamecraft-2: A 14b mixture-of-experts instruction-following game world model, 2025.

[32] Tencent Hunyuan Team. Hy-world 1.5 / worldplay technical report. Technical report, Tencent, 2025. URL https://3d-models.hunyuan.tencent.com/world/world1_5/HYWorld_1.5_Tech_Report.pdf.

[33] Tencent Hunyuan Team. Hunyuanworld 1.0: Layered panorama-to-3d scene generation, 2025. URL https://github.com/Tencent-Hunyuan/HunyuanWorld-1.0.

[34] Tencent Hunyuan Team. Hunyuanworld-mirror: Feed-forward multi-view 3d reconstruction. Technical report, Tencent, 2025. URL https://3d-models.hunyuan.tencent.com/world/worldMirror1_0/HYWorld_Mirror_Tech_Report.pdf.

[35] Tencent Hunyuan Team. Hunyuanworld-voyager: Joint rgb-depth video diffusion with a world cache, 2025.

[36] Tencent Hunyuan Team. Hy-world 2.0 technical report. Technical report, Tencent, 2026. URL https://3d-models.hunyuan.tencent.com/world/world2_0/HY_World_2_0.pdf.

[37] Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines (gamengen), 2024.

[38] Jianyuan Wang et al. Vggt: Visual geometry grounded transformer, 2025. CVPR 2025 Best Paper.

[39] Runqian Wang and Kaiming He. Diffuse and disperse: Image generation with representation regularization, 2025.

[40] World Labs. Marble: A 3d-native world model, 2025. URL https://www.worldlabs.ai/blog/marble-world-model.

[41] World Labs. Rtfm: A real-time frame model, 2025. URL https://www.worldlabs.ai/blog/rtfm.

[42] Guanjun Wu et al. 4d gaussian splatting for real-time dynamic scene rendering, 2024. CVPR 2024.

[43] Zhenheng Wu et al. Video world models with long-term spatial memory. 2025.

[44] Hong-Xing Yu et al. Wonderworld: Interactive 3d scene generation from a single image, 2025. CVPR 2025.