STATE-AWARE HYBRID DYNAMICS FOR GENERATIVE 4D GAUSSIANS

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent approaches to 4D generation represent dynamic objects using deformable 3D Gaussians whose transformations are predicted by continuous spatiotemporal fields. Such direct coordinate-time mappings, however, propagate no motion state between adjacent timestamps, and provide no mechanism for adapting how strongly each primitive contributes at a given timestamp. We propose State-Aware Hybrid Dynamics (SAHyD), which assigns each Gaussian a 3-D motion state in center-coordinate space and updates it through a hybrid rule combining a learned continuous drift with an expected latent-event response under soft motion regimes. The resulting regime and event features condition the continuous deformation representation before Gaussian attributes are decoded, so the state shapes the predicted offsets rather than correcting them afterwards. To regularize the learned transition across step sizes, Fine-Grained Motion Integration (FMint) learns a residual center-motion correction from a full-step versus two-half-step consistency reference. Time-Varying Gaussian Existence (TVGE) then predicts a soft participation probability that gates opacity and suppresses residual primitives, while preserving the number of Gaussians. Extensive experiments on video-to-4D generation demonstrate consistent improvements in temporal quality and overall reconstruction fidelity over the continuous-deformation backbone.

1 INTRODUCTION

Recent advances in generative modeling have enabled high-quality 2D image and static 3D content generation (Poole et al., 2023; Lin et al., 2023; Wang et al., 2023; Liu et al., 2023; 2024; Shi et al., 2024; Tang et al., 2024; Yi et al., 2024), stimulating growing interest in dynamic 4D content. By extending static 3D representations with temporal evolution, 4D generation supports applications such as digital content creation, immersive media, and dynamic scene synthesis. Among existing representations, 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) is particularly attractive for its explicit representation, efficient differentiable rendering, and flexible optimization. Recent work has explored both text-driven 4D synthesis (Singer et al., 2023) and Gaussian-based dynamic generation (Ren et al., 2023; Wu et al., 2024b; Zeng et al., 2024), including video-to-4D reconstruction from monocular videos (Wu et al., 2024b; Zeng et al., 2024).

A prevailing approach maintains a canonical set of Gaussians and models their temporal evolution with a continuous spatiotemporal deformation field. Given a Gaussian coordinate and timestamp, the field predicts time-dependent transformations of Gaussian attributes. For example, STAG4D (Zeng et al., 2024) adopts a HexPlane-based representation to model time-varying positions, scales, and rotations, providing a compact formulation for dynamic objects.

Despite their flexibility, direct coordinate–time mappings encode temporal variation only implicitly through spatial coordinates and timestamps. They lack an explicit dynamic state that organizes how local motion configurations evolve across neighboring observations. As a result, structurally distinct configurations may be represented by similar coordinate–time queries without sufficient constraints on their local temporal evolution. As shown in Figure 1, this can yield plausible global motion while introducing local structural deviations.

We address this limitation by introducing temporal state evolution as a *training-time dynamic inductive bias*, rather than persistent sequence-level memory. Within sampled temporal windows, a

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

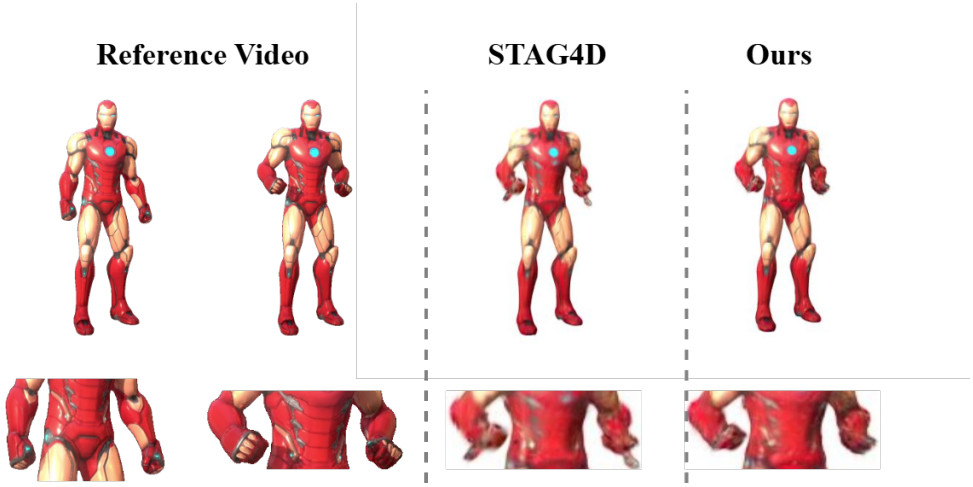


Figure 1: **Motivation for state-aware dynamic modeling.** STAG4D captures the overall motion but exhibits local structural deviations as the object configuration evolves. The enlarged regions highlight these discrepancies, whereas our method better preserves the evolving local structure.

compact per-Gaussian motion state evolves across adjacent timestamps and structures the deformation through state-dependent regime and event features. At inference, the learned deformation remains timestamp-conditioned and can be queried independently, without full-sequence state rollout. This design uses recurrent local dynamics to regularize neighboring temporal configurations while retaining flexible rendering at arbitrary timestamps.

Based on this principle, we introduce **State-Aware Hybrid Dynamics (SAHyD)**. Each Gaussian is associated with a 3-D motion state in center-coordinate space, which evolves through a learned continuous drift and an expected latent-event response. Frame-level Gaussian statistics and time encoding provide temporal context for inferring soft motion regimes and event-conditioned dynamic features. These features are injected into the deformation representation before Gaussian attributes are decoded. The state therefore acts as a structured conditioning variable for continuous deformation, rather than as a post-hoc geometric correction.

We further develop **Fine-Grained Motion Integration (FMint)** to improve local transition consistency. FMint compares one full-step transition with two successive half-step transitions and learns a residual center-motion correction, using the detached two-half-step result as an internal reference. This encourages coherent dynamics under changes in temporal discretization. In addition, **Time-Varying Gaussian Existence (TVGE)** models temporal primitive participation through a differentiable opacity gate, suppressing temporally irrelevant primitives while preserving a fixed canonical Gaussian set.

On Consistent4D, our full model reduces FVD from 995.11 to 793.10 over STAG4D, improves CLIP from 0.909 to 0.924, and decreases LPIPS from 0.126 to 0.120. It achieves the best FVD and CLIP among the methods in Table 1. Controlled experiments further show that these gains do not rely on persistent full-sequence state propagation or additional parameter capacity. Our main contributions are:

- We introduce **State-Aware Hybrid Dynamics (SAHyD)**, which uses windowed recurrent motion-state evolution to regularize continuous 4D Gaussian deformation. Per-Gaussian states evolve through continuous drift and expected latent-event responses under soft motion regimes, providing structured dynamic conditioning without requiring persistent state rollout at inference.
- We develop **Fine-Grained Motion Integration (FMint)** and **Time-Varying Gaussian Existence (TVGE)** for robust temporal modeling. FMint enforces local transition consistency through full-step and two-half-step integration, while TVGE adaptively controls temporal primitive participation through a differentiable opacity gate.

- Extensive experiments on video-to-4D generation demonstrate consistent improvements over the STAG4D backbone, including FVD from 995.11 to 793.10, CLIP from 0.909 to 0.924, and LPIPS from 0.126 to 0.120 on Consistent4D. Controlled ablations attribute these gains to structured local dynamics, step-consistent motion refinement, and adaptive primitive participation rather than increased model capacity.

2 Related Work

Video-to-4D Generation. Recent generative methods extend static 3D synthesis to dynamic 4D content. Consistent4D (Jiang et al., 2024) formulates monocular dynamic object reconstruction as a 4D generation problem and employs diffusion-based supervision to improve spatiotemporal consistency. Text-driven methods, including 4D-fy and Align Your Gaussians (Bahmani et al., 2024; Ling et al., 2024), explore diffusion-guided dynamic 3D generation. Video-conditioned Gaussian methods, such as DreamGaussian4D (Ren et al., 2023), SC4D (Wu et al., 2024b), and STAG4D (Zeng et al., 2024), further improve the efficiency and quality of monocular video-to-4D generation. Multiview and video diffusion methods strengthen cross-view and temporal supervision (Zhang et al., 2024; Huang et al., 2025; Xie et al., 2025). EG4D (Sun et al., 2025) explores explicit multiview video generation for image-conditioned 4D synthesis, while Gaussian Variation Field Diffusion (Zhang et al., 2025) directly models temporal Gaussian variations for video-to-4D generation. In this setting, our work focuses on motion-state evolution and temporal primitive participation within the underlying Gaussian representation.

Dynamic Gaussian Representations. 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) combines an explicit scene representation with efficient differentiable rendering, providing a foundation for dynamic Gaussian modeling. Earlier dynamic neural representations encode temporal variation through coordinate-conditioned deformation or factorized spatiotemporal features (Pumarola et al., 2021; Park et al., 2021a;b; Fang et al., 2022; Fridovich-Keil et al., 2023). Gaussian-based methods instead model dynamic scenes using explicit primitives that deform or evolve over time, including 4D Gaussian Splatting (Wu et al., 2024a), Deformable 3D Gaussians (Yang et al., 2024b), Native 4D Gaussians (Yang et al., 2024a), Dynamic 3D Gaussians (Luiten et al., 2024), and Spacetime Gaussian Feature Splatting (Li et al., 2024). Generative 4D methods further adopt time-conditioned representations to predict Gaussian transformations (Ren et al., 2023; Wu et al., 2024b; Zeng et al., 2024). In particular, STAG4D (Zeng et al., 2024) employs a HexPlane-based spatiotemporal representation for continuous Gaussian deformation. Building on this formulation, SAHyD learns per-Gaussian state evolution across adjacent timestamps within sampled temporal windows, whose regime and event features condition the deformation representation before Gaussian attributes are decoded.

Temporal Dynamics Modeling. Existing 4D generation methods typically model temporal evolution using timestamps, temporal features, video priors, or consistency objectives (Jiang et al., 2024; Ren et al., 2023; Wu et al., 2024b; Zeng et al., 2024). K-Planes (Fridovich-Keil et al., 2023) employs factorized space–time representations with temporal smoothness priors, providing an efficient model of continuous scene dynamics. More broadly, latent state-space and neural differential models explicitly represent hidden temporal dynamics (Rabiner, 1989; Chen et al., 2018; Rubanova et al., 2019; Kidger et al., 2020). FMint-SDE (Yuan et al., 2025) learns corrections to coarse SDE trajectories, while stochastic differential equations with jumps provide a classical framework for combining continuous evolution with jump components (Platen & Bruti-Liberati, 2010). Switching dynamical systems model latent regimes (Linderman et al., 2017), whereas Neural Jump SDEs incorporate state-dependent stochastic jumps (Jia & Benson, 2019). In contrast, SAHyD targets Gaussian deformation by combining learned continuous drift with expected latent-event responses under soft motion regimes, while FMint and TVGE further regularize state transitions and temporal primitive participation.

3 Method

Given a monocular dynamic video, we optimize a canonical set of 3D Gaussians using the continuous spatiotemporal deformation field of STAG4D as the backbone. Figure 2 illustrates the three proposed modules. SAHyD evolves a per-Gaussian motion state and conditions deformation fea-

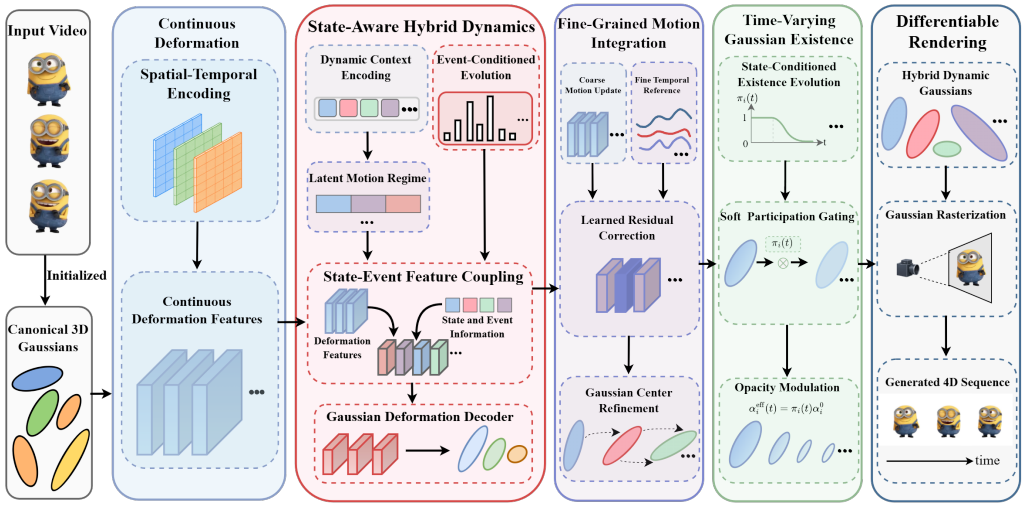


Figure 2: **Overview of our framework.** Given a monocular video, we optimize canonical 3D Gaussians with a continuous spatiotemporal deformation field. State-Aware Hybrid Dynamics (SAHyD) evolves per-Gaussian motion states under latent motion regimes and event-conditioned dynamics. The resulting regime and event features condition the deformation representation before Gaussian attributes are decoded. Fine-Grained Motion Integration (FMint) learns a residual state correction from a full-step versus two-half-step consistency reference, and the refined state increment adjusts the decoded Gaussian centers. Time-Varying Gaussian Existence (TVGE) controls each Gaussian's temporal participation through soft opacity gating. The refined geometry and modulated opacity form the hybrid dynamic Gaussians, which are differentiably rendered to produce the output sequence.

tures on latent motion regimes and event responses before attribute decoding. FMint regularizes state transitions across step sizes and learns a residual correction for the resulting center motion. TVGE controls the temporal participation of each primitive through a predicted opacity gate. The resulting Gaussians are then differentiably rendered to generate the output sequence.

3.1 Continuous Gaussian Deformation

We represent a dynamic object with a canonical set of Gaussian primitives, $\mathcal{G}_0 = \{\mathbf{g}_i^0\}_{i=1}^N$, where $\mathbf{g}_i^0 = (\boldsymbol{\mu}_i^0, \mathbf{s}_i^0, \mathbf{q}_i^0, \alpha_i^0, \mathbf{a}_i^0)$ denotes the center, scale, rotation, opacity, and appearance of the i-th primitive.

A spatiotemporal field encodes each canonical center together with its timestamp and produces a continuous deformation feature,

$$\mathbf{h}_i^t = \mathcal{H}_\theta(\boldsymbol{\mu}_i^0, t), \qquad \Delta\mathbf{g}_i^t = \mathcal{D}_\theta(\mathbf{h}_i^t), \tag{1}$$

where $\Delta\mathbf{g}_i^t = (\Delta\boldsymbol{\mu}_i^t, \Delta\mathbf{s}_i^t, \Delta\mathbf{q}_i^t)$ contains the position, scale, and rotation offsets. While this field captures continuous coordinate-dependent deformation, coordinates alone provide limited information about the underlying motion state. We therefore explicitly couple the deformation field with temporally evolving latent dynamics before decoding Gaussian attributes.

3.2 State-Aware Hybrid Dynamics

SAHyD assigns each Gaussian a 3-D motion state $\mathbf{z}_{i,t}$ in center space, initialized from the current canonical center $\boldsymbol{\mu}_i^0$ and propagated within each sampled temporal window. Latent regime and event representations derived from this state condition the continuous deformation field.

Dynamic Context and Latent Motion Regime. Motion evolution should depend not only on an individual primitive, but also on the instantaneous configuration of the object. We therefore

summarize each frame by a 14-D descriptor ϕ_t consisting of the mean and standard deviation of Gaussian centers, mean scale and rotation, and mean opacity. Together with a multi-frequency time encoding $\gamma(t)$, it defines a compact temporal context $\mathbf{c}_t = \mathcal{E}_c\big(\phi_t, \gamma(t)\big)$, where the resulting 8-D vector $\mathbf{c}_t$ is shared by all active Gaussians at timestamp t.

Conditioned on the current state and temporal context, SAHyD infers a distribution over latent motion regimes:

$$\begin{aligned} \widehat{\mathbf{p}}^{\,s}_{i,t} &= \mathrm{Softmax}\left(\mathcal{R}_\theta(\mathbf{z}_{i,t}, \mathbf{c}_t, t)/\tau\right), \\ \mathbf{p}^{s}_{i,t} &= \mathrm{Norm}\left(\tfrac{1}{2}\mathbf{p}^{s}_{i,t-1}\mathbf{A}_{i,t} + \tfrac{1}{2}\widehat{\mathbf{p}}^{\,s}_{i,t}\right). \end{aligned} \tag{2}$$

Here, τ controls the softmax temperature. A rate head predicts nonnegative transition rates from the current dynamic representation and converts them into the row-stochastic transition matrix $\mathbf{A}_{i,t}$. The propagated prior and instantaneous estimate are equally weighted, and $\mathrm{Norm}(\cdot)$ renormalizes the result. The regimes are learned without predefined semantic labels. During training, their soft representation is $\mathbf{e}^{\mathrm{state}}_{i,t} = \mathbf{p}^{s}_{i,t}\mathbf{E}_s$, where $\mathbf{E}_s$ is a learnable embedding matrix. At inference, we use the embedding of the most probable regime while retaining the full distribution for regime-prior propagation.

Event-Conditioned Evolution. Inspired by jump dynamics (Platen & Bruti-Liberati, 2010), we augment the continuous drift with a structured expected event response conditioned on $\mathbf{z}_{i,t}$, $\mathbf{c}_t$, and $\mathbf{e}^{\mathrm{state}}_{i,t}$. The dynamics module predicts a continuous drift $\mathbf{f}_{i,t}$, event intensity $\lambda_{i,t}$, event-type distribution $\boldsymbol{\rho}_{i,t}$, candidate responses $\{\mathbf{J}^{(m)}_{i,t}\}_{m=1}^{M}$, and an event embedding $\mathbf{e}^{\mathrm{evt}}_{i,t}$. The coarse transition is

$$\mathbf{z}^{c}_{i,t} = \mathbf{z}_{i,t} + \mathbf{f}_{i,t}\Delta t + \lambda_{i,t}\Delta t \sum_{m=1}^{M} \rho^{(m)}_{i,t}\mathbf{J}^{(m)}_{i,t}. \tag{3}$$

The drift models smooth evolution, while the expected event response captures state-dependent deviations. We evaluate this response deterministically rather than sampling discrete jumps; "event" therefore denotes an internal latent motion transition rather than an observed event. For brevity, we write $\mathbf{z}^{c}_{i,t} = \Psi_{\Delta t}(\mathbf{z}_{i,t})$.

State-Event Feature Coupling. The inferred dynamics influence deformation before geometric attributes are decoded. Specifically,

$$\widetilde{\mathbf{h}}^{t}_{i} = \mathbf{h}^{t}_{i} + \mathcal{P}_{\mathrm{state}}\left(\mathbf{e}^{\mathrm{state}}_{i,t}\right) + \mathcal{P}_{\mathrm{evt}}\left(\mathbf{e}^{\mathrm{evt}}_{i,t}\right), \tag{4}$$

and $\mathcal{D}_\theta$ maps $\widetilde{\mathbf{h}}^{t}_{i}$ to position, scale, and rotation offsets. Thus, the deformation field is no longer determined solely by space-time coordinates, but adapts to the evolving motion state.

3.3 Fine-Grained Motion Integration

A learned transition may become sensitive to the chosen temporal step size, leading to accumulated trajectory errors. FMint reduces this discrepancy by constructing an internal fine-step reference and using it to refine the coarse transition.

Full-Step / Two-Half-Step Consistency. The coarse update is $\mathbf{z}^{c}_{i,t} = \Psi_{\Delta t}(\mathbf{z}_{i,t})$. We obtain a finer reference by composing two half-step transitions:

$$\mathbf{z}^{f}_{i,t} = \Psi_{\Delta t/2}\left(\Psi_{\Delta t/2}(\mathbf{z}_{i,t})\right). \tag{5}$$

Both half steps share frozen timestamp-level conditioning (time encoding and frame-level global context), while only the motion state evolves sequentially. We stop gradients through $\mathbf{z}^{f}_{i,t}$ and use it as a self-generated consistency target.

Learned Motion Correction. A lightweight corrector predicts a residual $\delta\mathbf{z}_{i,t}$ from the coarse state and its dynamic context. The refined state and Gaussian center become

$$\begin{aligned} \mathbf{z}^{*}_{i,t} &= \mathbf{z}^{c}_{i,t} + \eta\,\delta\mathbf{z}_{i,t}, \\ \boldsymbol{\mu}^{\mathrm{mot}}_{i,t} &= \boldsymbol{\mu}^{\mathrm{def}}_{i,t} + \left(\mathbf{z}^{*}_{i,t} - \mathbf{z}_{i,t}\right), \end{aligned} \tag{6}$$

Algorithm 1: Training Procedure

Data: Frames $\{I_t\}$, canonical Gaussians $\mathcal{G}_0$, step Δt
Result: Optimized parameters Θ
while *not converged* **do**
  Sample temporal windows $\mathcal{W}$;
  foreach $\mathcal{W}$ **do**
    Initialize $(\mathbf{z}_{i,t}, \mathbf{p}^s_{i,t}, l^{\text{exist}}_{i,t})$;
    foreach $t \in \mathcal{W}$ **do**
      Compute $\mathbf{h}^t_i$ and context $\mathbf{c}_t$;
      `// SAHyD`
      Infer regime/event variables and compute $\mathbf{z}^c_{i,t} = \Psi_{\Delta t}(\mathbf{z}_{i,t})$;
      `// FMint`
      Compute $\mathbf{z}^f_{i,t} = \Psi_{\Delta t/2}(\Psi_{\Delta t/2}(\mathbf{z}_{i,t}))$;
      Refine $\mathbf{z}^*_{i,t} = \mathbf{z}^c_{i,t} + \eta\delta\mathbf{z}_{i,t}$;
      Condition $\mathbf{h}^t_i$ on state/event features and decode $\Delta\mathbf{g}^t_i$;
      Update Gaussian center using $\mathbf{z}^*_{i,t} - \mathbf{z}_{i,t}$;
      `// TVGE`
      Update $\pi_i(t)$ and set $\alpha^{\text{eff}}_i(t) = \pi_i(t)\alpha^0_i$;
      Render $\mathcal{G}^{\text{hyb}}_t$ and accumulate losses;
      Propagate $(\mathbf{z}^*_{i,t}, \mathbf{p}^s_{i,t}, l^{\text{exist}}_{i,t})$ within $\mathcal{W}$;
    end
  end
  Compute $\mathcal{L} = \mathcal{L}_{\text{gen}} + \lambda_{\text{FM}}\mathcal{L}_{\text{FM}} + \lambda_{\text{dyn}}\mathcal{L}_{\text{dyn}} + \lambda_{\text{exist}}\mathcal{L}_{\text{exist}} + \lambda_{\text{temp}}\mathcal{L}_{\text{temp}}$;
  Update Θ with Adam;
end

Algorithm 2: Inference Procedure

Data: Canonical Gaussians $\mathcal{G}_0$, timestamps $\{t\}$
Result: Rendered sequence $\{\hat{I}_t\}$
foreach *query timestamp* t **do**
  Compute $\mathbf{h}^t_i$ and temporal context $\mathbf{c}_t$;
  Initialize $\mathbf{z}_{i,t} \leftarrow \boldsymbol{\mu}^0_i$ and timestamp-conditioned $l^{\text{exist}}_{i,t}$;
  `// SAHyD`
  Infer regime/event variables (using cached regime prior when available) and compute $\mathbf{z}^c_{i,t} = \Psi_{\Delta t}(\mathbf{z}_{i,t})$;
  `// FMint`
  Refine $\mathbf{z}^*_{i,t} = \mathbf{z}^c_{i,t} + \eta\delta\mathbf{z}_{i,t}$;
  Condition $\mathbf{h}^t_i$ on state/event features and decode $\Delta\mathbf{g}^t_i$;
  Update Gaussian center using $\mathbf{z}^*_{i,t} - \mathbf{z}_{i,t}$;
  `// TVGE`
  Predict $\pi_i(t)$ and set $\alpha^{\text{eff}}_i(t) = \pi_i(t)\alpha^0_i$;
  Form $\mathcal{G}^{\text{hyb}}_t$;
  Render $\hat{I}_t = \mathcal{R}(\mathcal{G}^{\text{hyb}}_t)$;
end

where η controls the correction magnitude and $\boldsymbol{\mu}^{\text{def}}_{i,t}$ is the center predicted by the state-conditioned deformation field. Scale and rotation remain decoded from $\widetilde{\mathbf{h}}^t_i$.

Within each recurrent training window, the refined state is propagated recursively, $\mathbf{z}_{i,t+1} \leftarrow \mathbf{z}^*_{i,t}$, so that FMint improves the local trajectory rather than applying an isolated frame-wise correction. Its step-consistency supervision is defined in Sec. 3.5.

3.4 Time-Varying Gaussian Existence

A fixed set of primitives does not imply that every Gaussian should contribute equally at every timestamp. TVGE therefore assigns each primitive a differentiable temporal participation probability while preserving the underlying canonical representation.

Conditioned on the refined state, temporal context, regime embedding, and event feature, an existence controller predicts a drift $d^{\text{exist}}_{i,t}$ and an event-dependent logit correction $j^{\text{exist}}_{i,t}$:

$$l^{\text{exist}}_{i,t} = l^{\text{exist}}_{i,t-1} + d^{\text{exist}}_{i,t}\Delta t + \kappa j^{\text{exist}}_{i,t}, \pi_i(t) = \sigma\left(l^{\text{exist}}_{i,t}\right), \tag{7}$$

where σ denotes the sigmoid function and $\kappa = 1$ in all experiments. The event term aggregates the expected influence of latent event types on temporal appearance and disappearance. The participation probability gates the canonical opacity: $\alpha^{\text{eff}}_i(t) = \pi_i(t)\,\alpha^0_i$, where $\pi_i(t) \in (0, 1)$ controls the instantaneous contribution of the i-th Gaussian. TVGE therefore models time-varying visibility without discrete insertion or removal during rendering. Together with the FMint-refined geometry, these gated primitives form the dynamic representation $\mathcal{G}^{\text{hyb}}_t$.

3.5 Training Objective

The representation is optimized using RGB and mask reconstruction together with view-conditioned generative supervision. We denote their combined objective by $\mathcal{L}_{\text{gen}}$. FMint is trained by matching the refined transition to its detached two-half-step reference:

$$\mathcal{L}_{\text{FM}} = \left\| \mathbf{z}^*_{i,t} - \text{sg}\left(\mathbf{z}^f_{i,t}\right) \right\|^2_2, \tag{8}$$

where $\text{sg}(\cdot)$ denotes stop-gradient. We further regularize latent dynamics, temporal participation, and local temporal consistency through $\mathcal{L}_{\text{dyn}}$, $\mathcal{L}_{\text{exist}}$, and $\mathcal{L}_{\text{temp}}$, respectively. The complete objective is

$$\mathcal{L} = \mathcal{L}_{\text{gen}} + \lambda_{\text{FM}}\mathcal{L}_{\text{FM}} + \lambda_{\text{dyn}}\mathcal{L}_{\text{dyn}} + \lambda_{\text{exist}}\mathcal{L}_{\text{exist}} + \lambda_{\text{temp}}\mathcal{L}_{\text{temp}}. \tag{9}$$

The complete training and inference procedures of our framework are summarized in Algorithms 1 and 2, respectively. More theoretical analysis is provided in Appendix A.

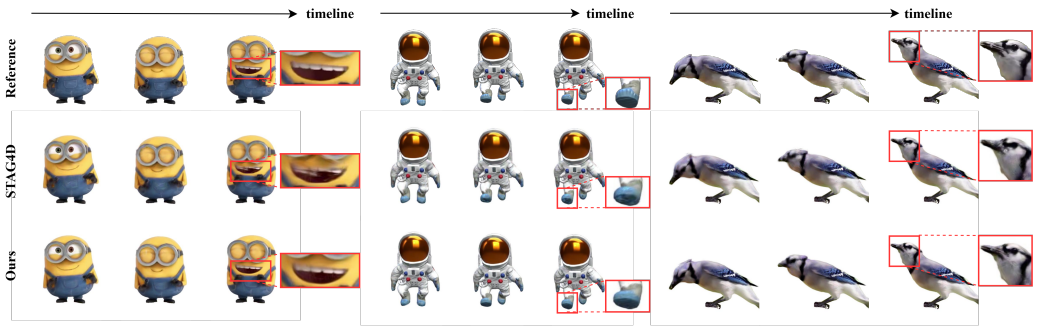


Figure 3: **Qualitative comparison of video-to-4D generation.** Reference frames, STAG4D, and our results are shown at representative timestamps. Red boxes and zoomed-in crops highlight challenging local regions, including the Minion's mouth, the astronaut's lower leg and foot, and the bird's head and neck. Our method better preserves these structures throughout motion.

4 Experiments

4.1 Experimental Setup

Datasets and Metrics. Following Consistent4D (Jiang et al., 2024), we evaluate video-to-4D generation on multiview videos of seven dynamic objects and additionally use Internet videos from STAG4D (Zeng et al., 2024) for qualitative evaluation on more diverse scenes. We render each optimized 4D Gaussian representation from four viewpoints and compare the resulting sequences with the corresponding ground-truth videos. We report LPIPS (Zhang et al., 2018) and CLIP similarity (Radford et al., 2021) for perceptual and semantic fidelity, respectively, and Fr'echet Video Distance (FVD) (Unterthiner et al., 2018) for spatiotemporal quality.

4.2 Implementation Details

We implement our method in PyTorch (Paszke et al., 2019) on top of the STAG4D (Zeng et al., 2024) training pipeline, using 512×512 input frames. Each scene is initialized with 10,000 Gaussian primitives and optimized for 8,000 steps, including 1,000 coarse-initialization steps. During dynamic optimization, we sample four temporal windows per step, each containing four consecutive frames with $\Delta t = 1/32$. Optimization uses Adam (Kingma & Ba, 2015), while the Gaussian optimization and view-conditioned guidance settings follow STAG4D. All experiments are conducted on a single NVIDIA L40S GPU.

At each timestamp, we construct a 14-D global summary of the Gaussian representation from the mean center (3-D), center standard deviation (3-D), mean scale (3-D), mean quaternion rotation (4-D), and mean opacity (1-D). These statistics are combined with a multi-frequency sinusoidal time encoding and projected into an 8-D dynamic context. Each Gaussian is associated with a 3-D motion state $\mathbf{z}_{i,t}$ defined in center-coordinate space. We use six soft motion regimes with 16-D regime embeddings and four latent event types with 32-D event features. All newly introduced MLPs have a hidden width of 64. More implementation details can be found in Appendix B.

4.3 Qualitative Results

Figure 3 compares our method with STAG4D (Zeng et al., 2024) at representative timestamps. Our method better preserves local structures under evolving object configurations, particularly in the Minion's mouth, the astronaut's lower leg and foot, and the bird's head and neck. The zoomed-in regions show closer agreement with the reference in shape and fine details, demonstrating that state-aware dynamic modeling improves local structural fidelity throughout motion.

Table 1: Quantitative comparison on Consistent4D. Best results are shown in bold.

Method	Reference	CLIP ↑	LPIPS ↓	FVD ↓
DG4D (Ren et al., 2023)	arXiv'23	0.880	0.140	853.21
4DGen (Yin et al., 2023)	arXiv'23	0.890	0.130	–
Consistent4D (Jiang et al., 2024)	ICLR'24	0.880	0.130	1133.22
SC4D (Wu et al., 2024b)	ECCV'24	0.900	0.140	879.88
STAG4D (Zeng et al., 2024)	ECCV'24	0.909	0.126	995.11
4Diffusion (Zhang et al., 2024)	NeurIPS'24	0.880	0.170	–
GVF Diffusion (Zhang et al., 2025)	ICCV'25	0.868	0.177	1101.24
MVTokenFlow (Huang et al., 2025)	ICLR'25	0.910	**0.120**	844.10
Ours	–	**0.924**	**0.120**	**793.10**

Table 2: Component ablation on the STAG4D backbone. Best results are shown in bold.

Variant	CLIP ↑	LPIPS ↓	FVD ↓
STAG4D	0.909	0.126	995.11
SAHyD	0.920	0.127	897.88
SAHyD + FMint	0.922	0.131	834.67
SAHyD + TVGE	0.920	0.127	846.20
SAHyD + FMint + TVGE	**0.924**	**0.120**	**793.10**

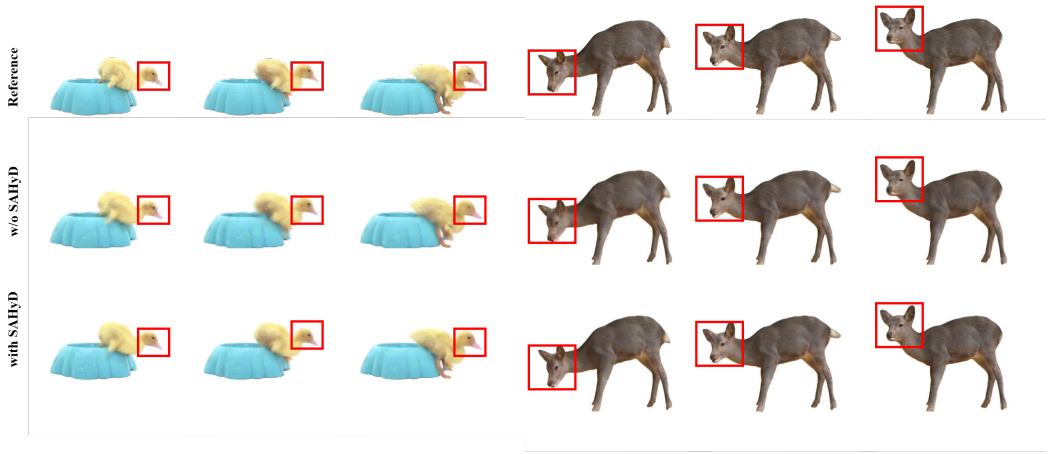


Figure 4: **Effect of State-Aware Hybrid Dynamics.** Comparison of the continuous-deformation baseline and SAHyD on two sequences. Red boxes highlight the duck and deer heads, where SAHyD better preserves the evolving reference geometry.

4.4 Quantitative Results and Comparison

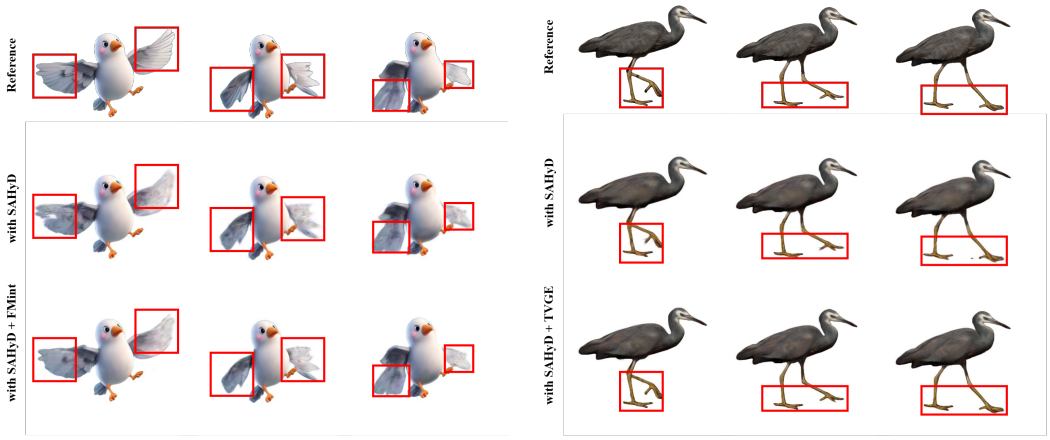
Table 1 compares our method with existing video-to-4D approaches on Consistent4D. Our method achieves the highest CLIP and lowest FVD while matching the best LPIPS at the reported precision. Compared with STAG4D, it reduces FVD from 995.11 to 793.10 (20.3%), increases CLIP from 0.909 to 0.924, and decreases LPIPS from 0.126 to 0.120. It also achieves a 6.0% lower FVD than MVTokenFlow (Huang et al., 2025), the strongest baseline in FVD. These gains indicate improved temporal coherence and reconstruction fidelity over the continuous-deformation backbone. Qualitative results further show more faithful evolving structures, supporting state-aware dynamic conditioning and temporal participation as complements to continuous Gaussian deformation.

4.5 Ablation Studies

We conduct quantitative and qualitative ablations to examine how SAHyD, FMint, and TVGE contribute to temporal coherence and reconstruction fidelity.

Component Ablation. Table 2 shows that SAHyD reduces FVD from 995.11 to 897.88 and improves CLIP from 0.909 to 0.920. FMint and TVGE further reduce FVD to 834.67 and 846.20, respectively, while the full model achieves the best overall result (CLIP 0.924, LPIPS 0.120, FVD 793.10). Additional internal and controlled ablations are provided in Appendices E and F.

Qualitative Ablation. Figures 4 and 5 show how the proposed modules improve local dynamic reconstruction. SAHyD better preserves the evolving head configurations of the duck and deer, sup-



(a) Fine-Grained Motion Integration **(b) Time-Varying Gaussian Existence**

Figure 5: **Qualitative ablation of FMint and TVGE.** (a) FMint improves agreement with the reference wing configurations during flapping. (b) TVGE reduces visible residual structures around the moving legs and feet. Red boxes highlight the corresponding regions across representative timestamps.

porting state-aware dynamic conditioning of Gaussian deformation. FMint improves the agreement of wing configurations with the reference sequence, consistent with its role in refining center motion through step-consistency learning. TVGE reduces residual structures around moving legs and feet, demonstrating the benefit of adapting primitive participation as the object configuration changes. Together with the FVD reductions in Table 2, these observations support more coherent temporal evolution and better preservation of local dynamic structure. Additional qualitative text-to-4D results are provided in Appendix H.

5 Conclusion

We presented SAHyD, a video-to-4D generation framework that augments continuous Gaussian deformation with state-conditioned dynamics and adaptive temporal participation. SAHyD introduces locally recurrent state evolution during training to guide timestamp-conditioned deformation through continuous drift and expected latent-event responses under soft motion regimes, without requiring persistent state rollout at inference. FMint promotes step-size-consistent transitions and refines center motion, while TVGE adaptively suppresses residual primitives through soft opacity gating without altering Gaussian cardinality. Experiments on Consistent4D demonstrate improved temporal coherence and reconstruction quality, achieving a 20.3% reduction in FVD over STAG4D together with consistent gains in CLIP and LPIPS. Ablation studies further validate the complementary contributions of the proposed components. Overall, the results demonstrate that state-conditioned dynamics and adaptive primitive participation provide an effective extension to continuous deformation for coherent 4D Gaussian generation.

AI Use Statement

Generative AI tools were used solely for language editing, structural refinement, methodological exposition, and figure-layout prototyping. They were not used to generate experimental data or evaluation results. All AI-assisted content was reviewed and verified by the authors, who take full responsibility for the final manuscript.

Reproducibility Statement

The model formulation and state evolution are described in Sec. 3, and the main training configuration is summarized in Sec. 4.2. The appendix provides full architectural specifications, loss definitions and weights, state initialization and temporal propagation details, optimization settings, and evaluation protocols required to reproduce the reported results.

References

Sherwin Bahmani, Ivan Skorokhodov, Victor Rong, Gordon Wetzstein, Leonidas Guibas, Peter Wonka, Sergey Tulyakov, Jeong Joon Park, Andrea Tagliasacchi, and David B. Lindell. 4d-fy: Text-to-4d generation using hybrid score distillation sampling. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7996–8006, 2024.

Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 2013. doi: 10.1093/acprof:oso/9780199535255.001.0001.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K. Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

Jiemin Fang, Taoran Yi, Xinggang Wang, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Matthias Nießner, and Qi Tian. Fast dynamic radiance fields with time-aware neural voxels. In *SIGGRAPH Asia 2022 Conference Papers*. ACM, 2022.

Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12479–12488, 2023.

Stéphane Gaubert and Zheng Qu. Dobrushin's ergodicity coefficient for Markov operators on cones. *Integral Equations and Operator Theory*, 81(1):127–150, 2015. doi: 10.1007/s00020-014-2193-2.

Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Advances in Neural Information Processing Systems*, volume 17, 2004.

GraphDeco-INRIA. Differential Gaussian rasterization: Forward rendering implementation. GitHub source repository, n.d. URL `https://github.com/graphdeco-inria/diff-gaussian-rasterization/blob/main/cuda_rasterizer/forward.cu`. Accessed September 20, 2026. This citation concerns the public implementation; record the exact commit used in experiments separately.

Ernst Hairer, Syvert P. Nørsett, and Gerhard Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems*, volume 8 of *Springer Series in Computational Mathematics*. Springer, 2 edition, 1993. doi: 10.1007/978-3-540-78862-1.

Hanzhuo Huang, Yuan Liu, Ge Zheng, Jiepeng Wang, Zhiyang Dou, and Sibei Yang. Mvtokenflow: High-quality 4d content generation using multiview token flow. In *International Conference on Learning Representations (ICLR)*, 2025.

Junteng Jia and Austin R. Benson. Neural jump stochastic differential equations. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Yanqin Jiang, Li Zhang, Jin Gao, Weiming Hu, and Yao Yao. Consistent4d: Consistent 360-degree dynamic object generation from monocular video. In *International Conference on Learning Representations (ICLR)*, 2024.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), 2023. SIGGRAPH 2023.

Patrick Kidger, James Morrill, James Foster, and Terry Lyons. Neural controlled differential equations for irregular time series. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 2020.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8508–8520, 2024.

Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 300–309, 2023.

Scott W. Linderman, Matthew J. Johnson, Andrew C. Miller, Ryan P. Adams, David M. Blei, and Liam Paninski. Bayesian learning and inference in recurrent switching linear dynamical systems. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 914–922, 2017.

Huan Ling, Seung Wook Kim, Antonio Torralba, Sanja Fidler, and Karsten Kreis. Align your gaussians: Text-to-4d with dynamic 3d gaussians and composed diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8576–8588, 2024.

Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9298–9309, 2023.

Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. In *International Conference on Learning Representations (ICLR)*, 2024.

Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *International Conference on 3D Vision (3DV)*, 2024.

Keunhong Park, Utkarsh Sinha, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Steven M. Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5865–5874, 2021a.

Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M. Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *ACM Transactions on Graphics*, 40(6), 2021b. SIGGRAPH Asia 2021.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Eckhard Platen and Nicola Bruti-Liberati. *Numerical Solution of Stochastic Differential Equations with Jumps in Finance*, volume 64 of *Stochastic Modelling and Applied Probability*. Springer, Berlin, Heidelberg, 2010. ISBN 978-3-642-13694-8. doi: 10.1007/978-3-642-13694-8.

Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *International Conference on Learning Representations (ICLR)*, 2023.

Thomas Porter and Tom Duff. Compositing digital images. In *Proceedings of the 11th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 253–259. ACM, 1984. doi: 10.1145/800031.808606.

Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10318–10327, 2021.

Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, pp. 8748–8763, 2021.

Jiawei Ren, Liang Pan, Jiaxiang Tang, Chi Zhang, Ang Cao, Gang Zeng, and Ziwei Liu. Dreamgaussian4d: Generative 4d gaussian splatting. *arXiv preprint arXiv:2312.17142*, 2023.

Yulia Rubanova, Ricky T. Q. Chen, and David K. Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5320–5330, 2019.

Yichun Shi, Peng Wang, Jianglong Ye, Long Mai, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. In *International Conference on Learning Representations (ICLR)*, 2024.

Uriel Singer, Shelly Sheynin, Adam Polyak, Oron Ashual, Iurii Makarov, Filippos Kokkinos, Naman Goyal, Andrea Vedaldi, Devi Parikh, Justin Johnson, and Yaniv Taigman. Text-to-4d dynamic scene generation. In *International Conference on Machine Learning (ICML)*, pp. 31915–31929, 2023.

J. Michael Steele. An Efron–Stein inequality for nonsymmetric statistics. *The Annals of Statistics*, 14(2):753–758, 1986. doi: 10.1214/aos/1176349952.

Qi Sun, Zhiyang Guo, Ziyu Wan, Jing Nathan Yan, Shengming Yin, Wengang Zhou, Jing Liao, and Houqiang Li. Eg4d: Explicit generation of 4d object without score distillation. In *International Conference on Learning Representations (ICLR)*, 2025.

Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. In *International Conference on Learning Representations (ICLR)*, 2024.

Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric and challenges. *arXiv preprint arXiv:1812.01717*, 2018.

Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023.

Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20310–20320, 2024a.

Zijie Wu, Chaohui Yu, Yanqin Jiang, Chenjie Cao, Fan Wang, and Xiang Bai. Sc4d: Sparse-controlled video-to-4d generation and motion transfer. In *European Conference on Computer Vision (ECCV)*, pp. 361–379. Springer, 2024b.

Yiming Xie, Chun-Han Yao, Vikram Voleti, Huaizu Jiang, and Varun Jampani. Sv4d: Dynamic 3d content generation with multi-frame and multi-view consistency. In *International Conference on Learning Representations (ICLR)*, 2025.

Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. In *International Conference on Learning Representations (ICLR)*, 2024a.

Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20331–20341, 2024b.

Taoran Yi, Jiemin Fang, Junjie Wang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Qi Tian, and Xinggang Wang. Gaussiandreamer: Fast generation from text to 3d gaussians by bridging 2d and 3d diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6796–6807, 2024.

Yuyang Yin, Dejia Xu, Zhangyang Wang, Yao Zhao, and Yunchao Wei. 4dgen: Grounded 4d content generation with spatial-temporal consistency. *arXiv preprint arXiv:2312.17225*, 2023.

Jiaxin Yuan, Haizhao Yang, and Maria Cameron. Fmint-sde: A multimodal foundation model for accelerating numerical simulation of sdes via error correction. *arXiv preprint arXiv:2510.27173*, 2025.

Yifei Zeng, Yanqin Jiang, Siyu Zhu, Yuanxun Lu, Youtian Lin, Hao Zhu, Weiming Hu, Xun Cao, and Yao Yao. Stag4d: Spatial-temporal anchored generative 4d gaussians. In *European Conference on Computer Vision (ECCV)*, 2024.

Bowen Zhang, Sicheng Xu, Chuxin Wang, Jiaolong Yang, Feng Zhao, Dong Chen, and Baining Guo. Gaussian variation field diffusion for high-fidelity video-to-4d synthesis. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12502–12513, 2025.

Haiyu Zhang, Xinyuan Chen, Yaohui Wang, Xihui Liu, Yunhong Wang, and Yu Qiao. 4diffusion: Multi-view video diffusion model for 4d generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 586–595, 2018.

A Theoretical Analysis

We study the proposed dynamics with fixed network parameters over a finite temporal window. The analysis establishes three properties: probability preservation and conditional perturbation propagation for SAHyD, consistency-controlled reference-trajectory error for FMint, and participation-induced rendering bounds for TVGE. These results concern the dynamics induced by the learned model; they do not imply convergence of the training optimizer, recovery of ground-truth motion, or monotonic improvement in perceptual metrics.

The recurrent results apply to the windowed rollout used during training. At evaluation time, timestamp-conditioned initialization permits analogous single-step statements, but does not correspond to a persistent rollout of the same recurrent system.

A.1 Notation and Assumptions

For the motion analysis, we suppress the Gaussian index and write $h = \Delta t > 0$. Probability vectors are treated as row vectors, and all vector norms are Euclidean unless the subscript 1 is explicitly used. For a matrix A, define $\|A\|_{\mathrm{row}} = \max_i \sum_j |A_{ij}|$.

The regime update is

$$p_n = \frac{1}{2} p_{n-1} A_n(z_n, c_n) + \frac{1}{2} \widehat{p}_n(z_n, c_n), \tag{10}$$

where $p_n \in \Delta^{K-1}$ is the regime distribution and c_n denotes the temporal context. Under the stochasticity conditions introduced below, the right-hand side already sums to one; hence, in exact arithmetic, the normalization used in the algorithm leaves it unchanged.

We define the effective deterministic motion field as

$$b_n(z, p, c) = f_n(z, p, c) + \lambda_n(z, p, c) \sum_{m=1}^{M} \rho_n^{(m)}(z, p, c) J_n^{(m)}(z, p, c). \tag{11}$$

The corrected state update is

$$z_{n+1} = z_n + h\, b_n(z_n, p_n, c_n) + r_n, \qquad r_n = \eta\, \delta z_n. \tag{12}$$

Eq. 11 directly uses the expected latent-event response. The resulting transition is therefore deterministic; our analysis does not assume a sampled Poisson process or an explicit stochastic jump trajectory.

Assumption 1 (Stochastic regime update and input sensitivity)**.** *The initial regime probability satisfies $p_{-1} \in \Delta^{K-1}$. At every step, $A_n(z,c)$ is row stochastic and $\widehat{p}_n(z,c) \in \Delta^{K-1}$. The dependence on p_{n-1} is exactly the explicit dependence shown in Eq. 10.*

There exist nonnegative constants a_z and a_c such that, on the comparison domain,

$$\frac{1}{2}\|A_n(z,c) - A_n(\tilde{z},\tilde{c})\|_{\mathrm{row}} + \frac{1}{2}\|\hat{p}_n(z,c) - \hat{p}_n(\tilde{z},\tilde{c})\|_1 \le a_z\|z-\tilde{z}\| + a_c\|c-\tilde{c}\|. \tag{13}$$

This assumption makes explicit the dependency structure required by the regime analysis. With finite network weights and a fixed positive softmax temperature, the corresponding mappings are locally regular on bounded domains. If the temperature varies, a positive lower bound is needed to obtain a uniform sensitivity constant. Any additional dependence of the prediction heads on the previous regime probability would require corresponding sensitivity terms.

Assumption 2 (Local regularity of the effective motion field)**.** *On the comparison domain, the effective motion field satisfies $\|b_n(z,p,c)\| \le B$ and*

$$\|b_n(z,p,c) - b_n(\widetilde{z},\widetilde{p},\widetilde{c})\| \le L_z\|z-\widetilde{z}\| + L_p\|p-\widetilde{p}\|_1 + L_c\|c-\widetilde{c}\|. \tag{14}$$

where the constants are finite and uniform over the analyzed window with fixed network parameters.

For instance, if $\|f_n\| \le F$, $0 \le \lambda_n \le \Lambda$, $\|J_n^{(m)}\| \le J_{\max}$, and $\rho_n \in \Delta^{M-1}$, then one may take $B = F + \Lambda J_{\max}$. If the component mappings admit Lipschitz constants $L_f, L_\lambda, L_\rho, L_J$ under a common input metric, then

$$L_b \le L_f + J_{\max}L_\lambda + \Lambda L_J + \Lambda J_{\max} L_\rho. \tag{15}$$

The training regularizers may favor moderate dynamics, but they do not by themselves imply such pointwise bounds over the entire domain.

Assumption 3 (Well-defined internal reference)**.** *For a shared sequence of external conditions, let $Q_{n,h}$ denote the implemented two-half-step reference map and $F_{n,h}$ the corrected update map. Both are defined on the states under comparison, and*

$$\|Q_{n,h}(u) - Q_{n,h}(v)\| \le L_{Q,n}\|u-v\|. \tag{16}$$

Any internally propagated variables are either included in the state or held fixed as common external inputs. In particular, the comparison does not implicitly recompute different external condition sequences.

We define the consistency residual and its trajectory-averaged loss by

$$\varepsilon_n = F_{n,h}(z_n) - Q_{n,h}(z_n), \qquad \ell_n = \|\varepsilon_n\|^2, \qquad \overline{\mathcal{L}}_{\mathrm{FM}} = \frac{1}{N}\sum_{n=0}^{N-1}\ell_n. \tag{17}$$

These quantities are definitions rather than additional assumptions. Eq. 17 uses an unweighted mean over the analyzed trajectory. If the implementation additionally averages over coordinates or Gaussian primitives, the corresponding normalization must be incorporated. Stop-gradient operations affect differentiation, but not the forward values appearing here.

Assumption 4 (Ideal compositing with fixed scene attributes)**.** *Fix a pixel, camera, Gaussian geometry, colors, candidate set, and front-to-back ordering. For the P candidate primitives, let $w_i = \alpha_i^0 G_i \in [0,1]$, where G_i denotes the Gaussian footprint at the pixel. The primitive colors and background color belong to a common convex set of finite diameter D.*

The compositing model is

$$C(a) = \sum_{i=1}^{P}\left[\prod_{j<i}(1-a_j)\right]a_i c_i + \left[\prod_{j=1}^{P}(1-a_j)\right]c_{\mathrm{bg}}. \tag{18}$$

We assume no gate-dependent opacity clipping, threshold-based skipping, or early termination. Conditional on the fixed scene attributes and participation probabilities, the auxiliary masks satisfy $B_i \sim \mathrm{Bernoulli}(\pi_i)$ independently.

Eq. 18 is the standard front-to-back alpha-compositing model used in Gaussian splatting (Porter & Duff, 1984; Kerbl et al., 2023). The independent Bernoulli masks are introduced only as an analytical device; they do not imply independent physical visibility or calibrated existence probabilities.

A.2 Probability Preservation and Perturbation Propagation

Theorem 5 (Probability preservation and conditional perturbation propagation)**.** *Under Assumptions 1 and 2, $p_n \in \Delta^{K-1}$ at every step.*

Consider two trajectories generated with the same network parameters and step size. For $\beta \geq \max\{1, L_p\}$, define

$$V_n = \|z_n - \widetilde{z}_n\| + h\beta\|p_{n-1} - \widetilde{p}_{n-1}\|_1. \tag{19}$$

Let $K = L_z + (L_p + \beta)a_z$ and $C_c = L_c + (L_p + \beta)a_c$. Then

$$V_{n+1} \leq (1 + hK)V_n + hC_c\|c_n - \widetilde{c}_n\| + \|r_n - \widetilde{r}_n\|. \tag{20}$$

Consequently, for $T = Nh$,

$$V_N \leq e^{KT}\left[V_0 + \sum_{n=0}^{N-1}\left(hC_c\|c_n - \widetilde{c}_n\| + \|r_n - \widetilde{r}_n\|\right)\right]. \tag{21}$$

If the state and context input sequences are identical, the regime recurrences satisfy

$$\|p_n - \widetilde{p}_n\|_1 \leq 2^{-(n+1)}\|p_{-1} - \widetilde{p}_{-1}\|_1. \tag{22}$$

Proof. A row-stochastic matrix maps a probability row vector to another probability row vector, and any convex combination of probability vectors remains in the simplex. Applying this observation inductively to Eq. 10 proves probability preservation.

For any row vector v and row-stochastic matrix A,

$$\|vA\|_1 \leq \sum_i |v_i| \sum_j A_{ij} = \|v\|_1. \tag{23}$$

This is the standard ℓ_1 nonexpansiveness of stochastic matrices underlying total-variation contraction (Gaubert & Qu, 2015).

Let $d_n = \|z_n - \widetilde{z}_n\|$, $e_n = \|p_{n-1} - \widetilde{p}_{n-1}\|_1$, and $u_n = \|c_n - \widetilde{c}_n\|$. Adding and subtracting $\frac{1}{2}\widetilde{p}_{n-1}A_n(z_n, c_n)$ in the difference of the two regime updates gives

$$\begin{aligned}
\|p_n - \widetilde{p}_n\|_1 &\leq \frac{1}{2}\|(p_{n-1} - \widetilde{p}_{n-1})A_n(z_n, c_n)\|_1 \\
&\quad + \frac{1}{2}\|\widetilde{p}_{n-1}[A_n(z_n, c_n) - A_n(\widetilde{z}_n, \widetilde{c}_n)]\|_1 \\
&\quad + \frac{1}{2}\|\widehat{p}_n(z_n, c_n) - \widehat{p}_n(\widetilde{z}_n, \widetilde{c}_n)\|_1 \\
&\leq \frac{1}{2}e_n + a_z d_n + a_c u_n.
\end{aligned} \tag{24}$$

The last step uses $\|\widetilde{p}_{n-1}\|_1 = 1$, Eq. 23, and Assumption 1.

Applying Assumption 2 to the corrected state update yields

$$\begin{aligned}
d_{n+1} &\leq (1 + hL_z)d_n + hL_p\|p_n - \widetilde{p}_n\|_1 + hL_c u_n + \|r_n - \widetilde{r}_n\| \\
&\leq [1 + h(L_z + L_p a_z)]d_n + \frac{hL_p}{2}e_n \\
&\quad + h(L_c + L_p a_c)u_n + \|r_n - \widetilde{r}_n\|.
\end{aligned} \tag{25}$$

Adding $h\beta$ times Eq. 24 to Eq. 25 gives

$$V_{n+1} \le [1 + hL_z + h(L_p + \beta)a_z]d_n + \frac{h}{2}(L_p + \beta)e_n + h[L_c + (L_p + \beta)a_c]u_n + \|r_n - \widetilde{r}_n\|. \tag{26}$$

Because $\beta \ge L_p$, $(L_p + \beta)/2 \le \beta$, and hence Eq. 20 follows.

Unrolling the scalar recurrence and using $(1 + hK)^N \le e^{KNh}$ gives Eq. 21. Finally, when $d_n = u_n = 0$, Eq. 24 reduces to $\|p_n - \widetilde{p}_n\|_1 \le e_n/2$. Iterating this inequality proves Eq. 22. □

Remark 6 (Scope of the perturbation bound). Theorem 5 is a conditional perturbation result, rather than an unconditional closed-loop contraction result. Differences in the temporal context and correction terms remain on the right-hand side of Eq. 20; controlling them requires additional assumptions on the corresponding feedback paths.

The probability discrepancy is also weighted by $h\beta$ in V_n. The bound therefore applies directly to a fixed step size and does not yield a step-size-uniform estimate for the unweighted probability error as $h \to 0$.

A.3 FMint Consistency and Reference-Trajectory Error

Theorem 7 (Consistency-controlled reference-trajectory error)**.** *Under Assumption 3, consider*

$$z_{n+1} = F_{n,h}(z_n), \qquad y_{n+1} = Q_{n,h}(y_n), \tag{27}$$

with the same external condition sequence. Then

$$\|z_N - y_N\| \le \left(\prod_{j=0}^{N-1} L_{Q,j}\right)\|z_0 - y_0\| + \sum_{k=0}^{N-1}\left(\prod_{j=k+1}^{N-1} L_{Q,j}\right)\sqrt{\ell_k}. \tag{28}$$

where an empty product is defined as one.

If $L_{Q,n} \le e^{Lh}$ for every step, then

$$\|z_N - y_N\| \le e^{LT}\left[\|z_0 - y_0\| + N\sqrt{\overline{\mathcal{L}}_{\mathrm{FM}}}\right], \qquad T = Nh. \tag{29}$$

Proof. A standard one-step error-propagation argument (Hairer et al., 1993) gives

$$\begin{aligned}\|z_{n+1} - y_{n+1}\| &= \|F_{n,h}(z_n) - Q_{n,h}(y_n)\| \\ &\le \|Q_{n,h}(z_n) - Q_{n,h}(y_n)\| + \|\varepsilon_n\| \\ &\le L_{Q,n}\|z_n - y_n\| + \sqrt{\ell_n}.\end{aligned} \tag{30}$$

Iterating this inequality proves 28.

If $L_{Q,n} \le e^{Lh}$, every product in 28 is bounded by e^{LT}. Furthermore, Cauchy–Schwarz yields

$$\sum_{n=0}^{N-1}\sqrt{\ell_n} \le \sqrt{N\sum_{n=0}^{N-1}\ell_n} = N\sqrt{\overline{\mathcal{L}}_{\mathrm{FM}}}. \tag{31}$$

Substitution gives Eq. 29. □

Theorem 7 does not require the internal reference map to approximate the solution of an ODE. The stronger interpretation below applies only when the two-half-step construction satisfies additional numerical-integration assumptions.

Corollary 8 (Conditional frozen-field integration bound)**.** *Suppose, in addition, that*

$$E_{n,s}(z) = z + s\bar{b}_n(z), \qquad Q_{n,h} = E_{n,h/2} \circ E_{n,h/2}, \tag{32}$$

where the same vector field, independent of the substep size, is used at both stages.

Assume that the field is defined on an open neighborhood of a common convex domain containing all numerical stages and exact local trajectories under comparison. On this domain, $\|\bar{b}_n(z)\| \le B$, and $\bar{b}_n$ is L-Lipschitz. Let $\Phi_{n,h}$ denote its local autonomous flow and define $x_{n+1} = \Phi_{n,h}(x_n)$. For $0 < h \le h_0$ and $T = Nh$,

$$\|z_N - x_N\| \le e^{LT}\left[\|z_0 - x_0\| + C_{\mathrm{ref}} Th + N\sqrt{\mathcal{L}_{\mathrm{FM}}}\right], \tag{33}$$

where

$$C_{\mathrm{ref}} = \frac{LB}{4}\left(1 + \frac{Lh_0}{4}\right). \tag{34}$$

If, in addition, $\bar{b}_n$ is C^2 with uniformly bounded first and second derivatives on the required neighborhood, then, for a fixed state and fixed conditioning,

$$\Phi_{n,h}(z) - E_{n,h}(z) = \frac{h^2}{2} a_n(z) + O(h^3), \tag{35}$$

$$\Phi_{n,h}(z) - F_{n,h}(z) = \frac{h^2}{4} a_n(z) - \varepsilon_{n,h}(z) + O(h^3), \tag{36}$$

where $a_n(z) = D\bar{b}_n(z)\bar{b}_n(z)$ and $\varepsilon_{n,h}(z) = F_{n,h}(z) - Q_{n,h}(z)$.

If $a_n(z) \neq 0$ and $\varepsilon_{n,h}(z) = O(h^3)$, then

$$\lim_{h\to 0} \frac{\|\Phi_{n,h}(z) - F_{n,h}(z)\|}{\|\Phi_{n,h}(z) - E_{n,h}(z)\|} = \frac{1}{2}. \tag{37}$$

Proof. For an Euler step of length s, the integral equation for the exact flow gives

$$\begin{aligned}\|\Phi_{n,s}(z) - E_{n,s}(z)\| &\le \int_0^s \|\bar{b}_n(\Phi_{n,u}(z)) - \bar{b}_n(z)\|\, du \\ &\le L\int_0^s \|\Phi_{n,u}(z) - z\|\, du \\ &\le LB\int_0^s u\, du = \frac{LB}{2}s^2.\end{aligned} \tag{38}$$

Moreover,

$$\|E_{n,s}(u) - E_{n,s}(v)\| \le (1 + sL)\|u - v\|. \tag{39}$$

Set $s = h/2$. Since the frozen autonomous flow satisfies $\Phi_{n,h} = \Phi_{n,s} \circ \Phi_{n,s}$, combining the two half-step defects gives

$$\begin{aligned}\|Q_{n,h}(z) - \Phi_{n,h}(z)\| &\le (1 + sL)\frac{LB}{2}s^2 + \frac{LB}{2}s^2 \\ &= \frac{LBh^2}{4}\left(1 + \frac{Lh}{4}\right) \\ &\le C_{\mathrm{ref}} h^2.\end{aligned} \tag{40}$$

In addition,

$$\mathrm{Lip}(Q_{n,h}) \le \left(1 + \frac{hL}{2}\right)^2 \le e^{Lh}. \tag{41}$$

Using $F_{n,h}(z_n) = Q_{n,h}(z_n) + \varepsilon_n$, we obtain

$$\begin{aligned}\|z_{n+1} - x_{n+1}\| &\le \|Q_{n,h}(z_n) - Q_{n,h}(x_n)\| + \|\varepsilon_n\| \\ &\quad + \|Q_{n,h}(x_n) - \Phi_{n,h}(x_n)\| \\ &\le e^{Lh}\|z_n - x_n\| + \sqrt{\ell_n} + C_{\mathrm{ref}} h^2.\end{aligned} \tag{42}$$

Unrolling this recurrence and applying Cauchy–Schwarz as in Theorem 7 proves Eq. 33.

Under the additional smoothness assumption, the classical local flow expansion (Hairer et al., 1993) gives

$$\Phi_{n,h}(z) = z + h\bar{b}_n(z) + \frac{h^2}{2} D\bar{b}_n(z)\bar{b}_n(z) + O(h^3). \tag{43}$$

For the two-half-step map,

$$Q_{n,h}(z) = z + \frac{h}{2}\bar{b}_n(z) + \frac{h}{2}\bar{b}_n\left(z + \frac{h}{2}\bar{b}_n(z)\right) = z + h\bar{b}_n(z) + \frac{h^2}{4} D\bar{b}_n(z)\bar{b}_n(z) + O(h^3). \tag{44}$$

Subtracting these expansions and using $F_{n,h} = Q_{n,h} + \varepsilon_{n,h}$ yields Eqs. 35 and 36.

If $a_n(z) \neq 0$ and $\varepsilon_{n,h}(z) = O(h^3)$, the leading terms of the coarse and corrected local error norms are respectively $\frac{h^2}{2}\|a_n(z)\|$ and $\frac{h^2}{4}\|a_n(z)\|$. Their ratio therefore converges to $1/2$, which proves Eq. 37. □

Remark 9 (Accuracy order and reference interpretation). Corollary 8 concerns the flow induced by a learned frozen vector field, rather than ground-truth physical motion. It applies only when the implementation follows the common-field construction in Eq. 32.

Reducing the leading local error coefficient by one half does not alter the $O(h^2)$ local truncation order and, by itself, does not imply second-order global accuracy. For a fixed horizon T, maintaining an $O(h)$ bound in Eq. 33 requires $\sqrt{\mathcal{L}_{\mathrm{FM}}} = O(h^2)$. A small consistency loss observed at a single step size does not establish this asymptotic scaling.

The ratio limit additionally assumes a consistent family of corrected maps as h varies. The frozen-field comparison does not account for nonautonomous time-freezing errors or discrepancies caused by different context-feedback sequences.

A.4 TVGE Rendering Guarantees

Theorem 10 (Ideal expected rendering and participation-induced variance)**.** *Under Assumption 4, define $C_B = C(w \odot B)$ and $C_\pi = C(w \odot \pi)$. Then*

$$\mathbb{E}[C_B] = C_\pi. \tag{45}$$

For any fixed target color y,

$$\mathbb{E}\|C_B - y\|^2 = \|C_\pi - y\|^2 + \mathbb{E}\|C_B - C_\pi\|^2. \tag{46}$$

Moreover,

$$\mathbb{E}\|C_B - C_\pi\|^2 \leq D^2 \sum_{i=1}^{P} w_i^2 \pi_i (1 - \pi_i), \tag{47}$$

and, for two participation vectors with the same fixed scene attributes,

$$\|C_\pi - C_{\widetilde{\pi}}\| \leq D \sum_{i=1}^{P} w_i |\pi_i - \widetilde{\pi}_i|. \tag{48}$$

More precisely, let $\overline{C}_{>i}$ denote the expected composite color of the layers behind primitive i, including the background. Then the variance admits the exact decomposition

$$\mathbb{E}\|C_B - C_\pi\|^2 = \sum_{i=1}^{P} w_i^2 \pi_i (1 - \pi_i) \prod_{j<i} (1 - 2w_j\pi_j + w_j^2\pi_j) \|c_i - \overline{C}_{>i}\|^2. \tag{49}$$

Proof. By conditional independence,

$$\mathbb{E}\left[w_i B_i \prod_{j<i} (1 - w_j B_j) \right] = w_i \mathbb{E}[B_i] \prod_{j<i} \mathbb{E}[1 - w_j B_j] = w_i \pi_i \prod_{j<i} (1 - w_j \pi_j). \tag{50}$$

Similarly, the background term satisfies

$$\mathbb{E}\left[\prod_{j=1}^{P}(1-w_jB_j)\right]=\prod_{j=1}^{P}(1-w_j\pi_j). \tag{51}$$

Substituting these identities into Eq. 18 gives Eq. 45. Expanding the squared error around $\mathbb{E}[C_B]=C_\pi$ then yields Eq. 46.

Next, fix all opacities except a_i and write

$$C(a)=C_{<i}+T_i\left[a_ic_i+(1-a_i)C_{>i}\right], \tag{52}$$

where $T_i=\prod_{j<i}(1-a_j)\in[0,1]$. The tail color $C_{>i}$ is a convex combination of the colors behind primitive i and the background, and therefore belongs to the same convex color set. Changing only a_i thus gives

$$\|C(a)-C(a')\|=T_i|a_i-a_i'|\|c_i-C_{>i}\|\le D|a_i-a_i'|. \tag{53}$$

Changing the opacities one at a time and applying the triangle inequality therefore yields

$$\|C(a)-C(a')\|\le D\sum_i|a_i-a_i'|. \tag{54}$$

Taking $a_i=w_i\pi_i$ and $a_i'=w_i\widetilde{\pi}_i$ proves Eq. 48.

For the variance bound, let $B^{(i)}$ denote the vector obtained by replacing B_i with an independent copy B_i'. Applying the Efron–Stein inequality coordinatewise and summing over the color dimensions (Steele, 1986; Boucheron et al., 2013) gives

$$\mathbb{E}\|C_B-\mathbb{E}C_B\|^2\le\frac{1}{2}\sum_i\mathbb{E}\|C_B-C_{B^{(i)}}\|^2. \tag{55}$$

Equation 53 implies

$$\|C_B-C_{B^{(i)}}\|\le Dw_i|B_i-B_i'|. \tag{56}$$

Since $\mathbb{E}|B_i-B_i'|^2=2\pi_i(1-\pi_i)$, Eq. 55 yields Eq. 47.

For the exact decomposition, define the Doob martingale

$$M_i=\mathbb{E}[C_B\mid B_1,\dots,B_i],\qquad M_0=\mathbb{E}[C_B]. \tag{57}$$

With $T_i=\prod_{j<i}(1-w_jB_j)$, conditional independence gives

$$M_i-M_{i-1}=T_iw_i(B_i-\pi_i)(c_i-\overline{C}_{>i}). \tag{58}$$

The martingale differences are orthogonal in L^2. Indeed, for $i<k$, conditioning on $B_1,\dots,B_{k-1}$ makes the conditional expectation of M_k-M_{k-1} equal to zero, while M_i-M_{i-1} is measurable with respect to the same σ-algebra. Consequently,

$$\mathbb{E}\|C_B-\mathbb{E}C_B\|^2=\sum_i\mathbb{E}\|M_i-M_{i-1}\|^2. \tag{59}$$

Finally,

$$\mathbb{E}[T_i^2]=\prod_{j<i}\mathbb{E}[(1-w_jB_j)^2]=\prod_{j<i}(1-2w_j\pi_j+w_j^2\pi_j). \tag{60}$$

Substituting this identity into Eq. 58 and summing over i proves Eq. 49. □

Connection to TVGE regularization. Let N_G denote the total number of Gaussian primitives, distinguishing it from the number N of temporal updates. For a fixed pixel, set $w_i=0$ for primitives that do not contribute to that pixel. Using the same primitive-wise normalization as the training objective, define

$$\mathcal{L}_{\text{conf}}=\frac{1}{N_G}\sum_{i=1}^{N_G}\pi_i(1-\pi_i),\mathcal{L}_{\text{smooth},n}=\frac{1}{N_G}\sum_{i=1}^{N_G}(\pi_{i,n+1}-\pi_{i,n})^2. \tag{61}$$

Theorem 10 immediately gives

$$\mathbb{E}\|C_B - C_\pi\|^2 \le D^2 N_G \mathcal{L}_{\text{conf}}. \tag{62}$$

For fixed geometry, colors, and ordering, Cauchy–Schwarz further yields

$$\begin{aligned}\|C_{\pi_{n+1}} - C_{\pi_n}\|^2 &\le D^2 \left[\sum_i w_i |\pi_{i,n+1} - \pi_{i,n}|\right]^2 \\ &\le D^2 \|w\|_2^2 \sum_i (\pi_{i,n+1} - \pi_{i,n})^2 \\ &= D^2 \|w\|_2^2 N_G \mathcal{L}_{\text{smooth},n}.\end{aligned} \tag{63}$$

Hence, the confidence penalty controls an upper bound on the rendering variance induced by the auxiliary independent-mask model, while the smoothness penalty controls rendering changes caused solely by temporal variation in participation. The term $\pi(1-\pi)$ is the variance of a Bernoulli variable, rather than Shannon entropy. Entropy minimization offers related motivation (Grandvalet & Bengio, 2004), but is not the source of the variance identity or the bound above.

Neither regularizer alone guarantees correct visibility or artifact identification. Both $\pi \approx 0$ and $\pi \approx 1$ have small Bernoulli variance, so reconstruction and generative objectives remain necessary to determine which primitives should participate.

Discrete temporal increment bound. For the TVGE logit update

$$l_{i,n+1} - l_{i,n} = h\, d_{i,n} + \kappa j_{i,n}, \qquad \pi_{i,n} = \sigma(l_{i,n}), \tag{64}$$

the sigmoid derivative bound $0 < \sigma'(l) \le 1/4$ gives

$$|\pi_{i,n+1} - \pi_{i,n}| \le \frac{1}{4} |h\, d_{i,n} + \kappa j_{i,n}|. \tag{65}$$

Combining this inequality with Eq. 48 yields

$$\|C_{\pi_{n+1}} - C_{\pi_n}\| \le \frac{D}{4} \sum_i w_i |h\, d_{i,n} + \kappa j_{i,n}|. \tag{66}$$

This is a discrete-increment bound. Because the event-dependent term $j_{i,n}$ is not explicitly multiplied by h, a continuous-time Lipschitz interpretation requires an additional scaling condition, such as $j_{i,n} = O(h)$.

Suppression of low-participation primitives. Let S be a subset of primitives satisfying $\pi_i \le \epsilon$ for all $i \in S$, and let $\pi^{(-S)}$ be obtained by setting their participation probabilities to zero while keeping all other attributes unchanged. Equation 48 gives

$$\|C_\pi - C_{\pi^{(-S)}}\| \le D\epsilon \sum_{i \in S} w_i. \tag{67}$$

Thus, low-participation primitives have a controlled influence on the rendered color. The bound does not, however, imply that every undesirable primitive will necessarily receive low participation probability.

A.5 Implementation Scope and Limitations

Ideal versus implemented compositing. The public Gaussian rasterizer employs opacity clipping, small-opacity skipping, and transmittance-based early termination (GraphDeco-INRIA, n.d.). These operations do not necessarily preserve the multi-affine dependence on participation variables used in Eq. 45. Theorem 10 therefore applies explicitly to the ideal compositor in Eq. 18.

If the implemented renderer $\widehat{C}$ satisfies $\|\widehat{C}(a) - C(a)\| \le \delta_{\text{rast}}$ for all compared inputs, then

$$\begin{aligned}&\|\mathbb{E}[\widehat{C}(w \odot B)] - \widehat{C}(w \odot \pi)\| \\ &\quad \le \mathbb{E}\|\widehat{C}(w \odot B) - C(w \odot B)\| + \|C(w \odot \pi) - \widehat{C}(w \odot \pi)\| \\ &\quad \le 2\delta_{\text{rast}}.\end{aligned} \tag{68}$$

Such an approximation guarantee must be established for the specific renderer implementation and is not assumed to be negligible.

Geometry changes versus gate changes. For time-varying scene attributes g_n,

$$C(g_{n+1}, \pi_{n+1}) - C(g_n, \pi_n) = \big[C(g_{n+1}, \pi_{n+1}) - C(g_{n+1}, \pi_n)\big] + \big[C(g_{n+1}, \pi_n) - C(g_n, \pi_n)\big]. \tag{69}$$

The TVGE sensitivity bound controls only the first term. The second captures changes in geometry, motion, deformation, appearance, and potential depth ordering, and therefore requires a separate analysis.

Hard regimes at evaluation. Soft regime embeddings vary continuously with the regime probabilities, whereas hard selection can be discontinuous at probability ties. The preceding regularity assumptions therefore apply directly to the soft-mode dynamics.

Suppose the largest regime probability satisfies the margin condition

$$p_{k_*} - \max_{j \neq k_*} p_j \geq m > 0. \tag{70}$$

Then every perturbation satisfying $\|p - \widetilde{p}\|_\infty < m/2$ preserves the selected regime. Moreover, defining $D_E = \max_{j,k} \|E_j - E_k\|$, we have

$$\left\| \sum_k p_k E_k - E_{k_*} \right\| \leq \sum_{k \neq k_*} p_k \|E_k - E_{k_*}\| \leq D_E(1 - p_{k_*}). \tag{71}$$

These are local statements about selection stability and the soft-to-hard embedding discrepancy; they do not imply global smoothness of the hard-mode dynamics. Empirical soft–hard comparisons are reported in Appendix F.

Interpretation of the guarantees. The analysis relates the proposed update rules and regularization terms to explicit conditional bounds. The stochastic-matrix arguments, numerical-integration estimates, martingale decomposition, and Efron–Stein inequality are classical tools. Their role here is to characterize the coupled SAHyD–FMint–TVGE construction and to make clear which conclusions follow under the stated assumptions, as well as where the guarantees cease to apply.

B Full Implementation Details

We implement the proposed dynamic modules on top of the STAG4D training pipeline (Zeng et al., 2024). This section summarizes the architectural configuration, state initialization, and temporal propagation details omitted from the main paper.

Dynamic context and state representation. At each timestamp, we construct a 14-D frame-level summary of the Gaussian representation, consisting of the mean Gaussian center (3-D), center standard deviation (3-D), mean scale (3-D), mean quaternion rotation (4-D), and mean opacity (1-D). These statistics are concatenated with a multi-frequency sinusoidal time encoding and projected into an 8-D dynamic context vector.

The per-Gaussian motion state $\mathbf{z}_{i,t}$ is 3-D and defined in Gaussian center-coordinate space. The latent-regime module uses six soft motion regimes with 16-D embeddings, while the event-conditioned evolution module uses four latent event types with 32-D event features. All newly introduced MLPs use a hidden width of 64.

Optimization configuration. The dynamic modules are optimized with a learning rate of 1.6×10^{-4}. The inherited continuous deformation field uses an exponential learning-rate schedule from 1.6×10^{-4} to 1.6×10^{-6}. FMint uses a residual correction scale $\eta = 0.1$, a 16-D event-mark embedding, and a correction-loss weight of 0.1. TVGE initializes its temporal participation prior to 0.99.

Each scene is initialized with 10,000 Gaussian primitives. Gaussian initialization, adaptive densification and pruning, and all other optimization settings follow STAG4D (Zeng et al., 2024) unless otherwise specified.

Windowed state propagation during training. During dynamic optimization, we sample four temporal windows per step, each containing four consecutive frames with $\Delta t = 1/32$. At the first timestamp of each window, no recurrent state is carried over from previous windows. The motion state $\mathbf{z}_{i,t}$ is initialized from the current canonical center $\boldsymbol{\mu}_i^0$, while the latent-regime probabilities and TVGE participation logit are initialized by their corresponding prediction modules. After each timestamp, the FMint-refined state $\mathbf{z}_{i,t}^*$, soft regime distribution, and participation logit are propagated to the next timestamp within the same window and reset when a new window is sampled.

State handling at evaluation. The reported evaluation uses chronological timestamp-conditioned initialization rather than persistent full-sequence motion-state rollout. At each timestamp, $\mathbf{z}_{i,t}$ is initialized from the current canonical center $\boldsymbol{\mu}_i^0$, TVGE uses its timestamp-conditioned participation logit, and the regime module may reuse the preceding frame-level regime distribution as a cached prior.

C Full Training Objectives

We summarize the regularization terms used to train the proposed dynamic modules. All experiments use deterministic expected latent-event responses without stochastic Poisson sampling.

Let $\bar{\mathbf{p}}_t$ be the frame-averaged regime probability, $\mathbf{p}_{i,t}$ the per-Gaussian regime probability, $\mathbf{A}_{i,t}$ the learned transition matrix, $\bar{\mathbf{J}}_{i,t}$ the expected latent-event response, $\mathbf{f}_{i,t}$ the drift term, and $\pi_i(t)$ the TVGE participation probability.

State and event regularization. We regularize the regime and event dynamics as

$$\mathcal{L}_{\text{reg-smooth}} = \|\bar{\mathbf{p}}_t - \bar{\mathbf{p}}_{t-1}\|_2^2, \quad \mathcal{L}_{\text{event-sparse}} = \mathbb{E}_i[\lambda_{i,t}], \quad \mathcal{L}_{\text{jump}} = \mathbb{E}_i\left[\left\|\bar{\mathbf{J}}_{i,t}\right\|_2^2\right]. \tag{72}$$

$$\mathcal{L}_{\text{drift}} = \mathbb{E}_i\left[\|\mathbf{f}_{i,t}\|_2^2\right], \quad \mathcal{L}_{\text{transition}} = \mathbb{E}_i\left[\|\mathbf{p}_{i,t} - \mathbf{p}_{i,t-1}\mathbf{A}_{i,t-1}\|_2^2\right]. \tag{73}$$

The first and last terms promote temporally consistent regime dynamics, while the remaining terms regularize event intensity, jump magnitude, and drift magnitude. Thus,

$$\lambda_{\text{dyn}}\mathcal{L}_{\text{dyn}} = 0.01\,\mathcal{L}_{\text{reg-smooth}} + 0.001\,\mathcal{L}_{\text{event-sparse}} + 0.001\,\mathcal{L}_{\text{jump}} + 0.0005\,\mathcal{L}_{\text{drift}} + 0.01\,\mathcal{L}_{\text{transition}}. \tag{74}$$

Local temporal consistency. We encourage neighboring state updates to vary smoothly:

$$\mathcal{L}_{\text{vel}} = \mathbb{E}_i\left[\left\|(\mathbf{z}_{i,t}^* - \mathbf{z}_{i,t}) - (\mathbf{z}_{i,t-1}^* - \mathbf{z}_{i,t-1})\right\|_2^2\right], \tag{75}$$

with $\lambda_{\text{temp}}\mathcal{L}_{\text{temp}} = 0.01\,\mathcal{L}_{\text{vel}}$.

TVGE regularization. To encourage temporally coherent Gaussian participation, following the temporal-smoothness principle commonly used in dynamic neural representations (Fridovich-Keil et al., 2023), we define

$$\mathcal{L}_{\text{exist-smooth}} = \mathbb{E}_i\left[(\pi_i(t) - \pi_i(t-1))^2\right], \qquad \mathcal{L}_{\text{exist-conf}} = \mathbb{E}_i[\pi_i(t)(1 - \pi_i(t))]. \tag{76}$$

The second term is the Bernoulli variance of the participation variable and serves as a simple low-uncertainty surrogate related to entropy-minimization regularization (Grandvalet & Bengio, 2004), discouraging ambiguous participation probabilities near 0.5. Their combined contribution is

$$\lambda_{\text{exist}}\mathcal{L}_{\text{exist}} = 0.01\,\mathcal{L}_{\text{exist-smooth}} + 0.01\,\mathcal{L}_{\text{exist-conf}}. \tag{77}$$

FMint consistency. FMint uses the correction loss in Eq. 8, weighted as

$$\lambda_{\text{FM}}\mathcal{L}_{\text{FM}} = 0.1\,\mathcal{L}_{\text{FM}}. \tag{78}$$

The two-half-step reference is detached during optimization, as described in Sec. 3.3.

Inherited deformation regularization. Following STAG4D (Zeng et al., 2024), we retain time-smoothness, plane-TV, and temporal-plane ℓ_1 regularization with weights 5, 0.05, and 0.05, respectively. These terms are included in $\mathcal{L}_{\text{gen}}$ in the main paper.

D Training and Evaluation Protocol

We provide additional details on the optimization and evaluation protocols. Quantitative evaluation follows the Consistent4D benchmark (Jiang et al., 2024), while additional Internet videos released with STAG4D (Zeng et al., 2024) are used for qualitative evaluation on more diverse dynamic content.

Optimization protocol. Each scene is optimized independently using Adam with $\epsilon = 10^{-15}$ at a resolution of 512×512. Optimization runs for 8,000 iterations, with the first 1,000 iterations used for coarse Gaussian optimization before enabling the proposed dynamic modules. All experiments are conducted on a single NVIDIA L40S GPU. Unless otherwise stated, the remaining Gaussian optimization and view-conditioned guidance settings follow STAG4D (Zeng et al., 2024).

Quantitative evaluation. Following Consistent4D, we evaluate each optimized 4D representation from four viewpoints. Rendered sequences are compared with the corresponding ground-truth videos using CLIP similarity (CLIP), LPIPS, and Fr'echet Video Distance (FVD), which measure semantic consistency, perceptual similarity, and spatiotemporal fidelity, respectively.

CLIP. Predicted and ground-truth frames are resized to 512×512, with transparent regions composited onto a white background when needed. Image features are extracted using `openai/clip-vit-base-patch32`, and cosine similarity is computed between corresponding frames. Scores are averaged over timestamps, viewpoints, and finally objects.

LPIPS. LPIPS is computed between corresponding predicted and ground-truth frames at 512×512 resolution using the VGG backbone implemented in TorchMetrics. The same white-background compositing is applied to transparent regions. Scores are averaged over timestamps, viewpoints, and objects.

FVD. For each viewpoint, 32 temporally ordered frames are uniformly sampled, composited onto a white background when necessary, resized to 224×224, normalized to $[-1, 1]$, and encoded into one I3D clip feature. Each object-level FVD therefore uses four predicted and four ground-truth clip features, one per viewpoint. The final score is obtained by averaging the object-level FVD values over the seven evaluated objects.

Additional qualitative evaluation. We further evaluate the method on the Internet video sequences used by STAG4D, which contain more diverse object appearances and motion patterns. These sequences are used only for qualitative analysis and are excluded from the benchmark averages reported in the main quantitative results.

E Internal Ablation of SAHyD

The main paper ablates SAHyD, FMint, and TVGE at the module level. Here, we further examine the internal design of SAHyD by individually removing latent motion-regime modeling, event-conditioned state evolution, and state–event feature coupling. FMint and TVGE are disabled in all variants, ensuring that the comparison isolates the contribution of each SAHyD mechanism.

Table 3: **Internal ablation of State-Aware Hybrid Dynamics (SAHyD).** Each row removes one SAHyD component while keeping all other training and evaluation settings fixed. FMint and TVGE are excluded from this ablation.

Variant	CLIP ↑	LPIPS ↓	FVD ↓
SAHyD w/o Latent Motion Regime	0.916	0.130	936.42
SAHyD w/o Event-Conditioned Evolution	0.914	0.132	951.73
SAHyD w/o State-Event Feature Coupling	0.917	0.129	924.86
Full SAHyD	**0.920**	**0.127**	**897.88**

As shown in Table 3, removing any of the three components consistently degrades performance relative to the full SAHyD model. The largest drop occurs without event-conditioned evolution, where FVD increases from 897.88 to 951.73, CLIP decreases from 0.920 to 0.914, and LPIPS rises from 0.127 to 0.132. This highlights the importance of event-conditioned state updates for temporal modeling.

Removing the latent motion-regime module also degrades performance, increasing FVD to 936.42. Complementary regime-cardinality experiments in Appendix F further show that retaining the pathway with a single shared regime remains inferior to the default multi-regime configuration. Removing state-event feature coupling yields a smaller but consistent decline, with FVD increasing to 924.86, indicating that feature-level conditioning complements the explicit state update.

Overall, the full SAHyD model performs best across CLIP, LPIPS, and FVD, confirming the complementary contributions of regime inference, event-conditioned evolution, and state-event feature coupling.

F Additional Controlled Diagnostics

We conduct additional controlled experiments to examine state handling, model capacity, regime cardinality, regime inference, and temporal opacity modeling. All experiments in this section follow the same seven-object, 32-frame, four-view evaluation protocol as the main quantitative evaluation. The full reference configuration corresponds to the best-FVD model reported in Table 1, with an FVD of 793.10.

State propagation and evaluation. We first examine whether the performance gain requires persistent full-sequence motion-state propagation. We compare a fully non-recurrent variant, recurrent training with strict-reset evaluation, persistent full-sequence rollout, and the hybrid evaluation used by the full model. In the hybrid setting, the motion state is reinitialized at each timestamp while the preceding regime distribution may be reused as a cached prior.

Table 4: **Controlled comparison of state handling.** All variants use the same evaluation protocol.

Variant	Training	Evaluation	CLIP ↑	LPIPS ↓	FVD ↓
Non-recurrent	Independent	Reset	0.923	0.118	834.09
Recurrent + Reset	Windowed	Reset	0.925	0.118	813.61
Recurrent + Persistent	Windowed	Persistent	0.919	0.118	832.22
Hybrid reference	Windowed	Timestamp-init + cache	0.924	0.120	793.10

Windowed recurrent training reduces FVD from 834.09 to 813.61 even under strict-reset evaluation, showing that local recurrent training improves the learned dynamics without requiring persistent full-sequence motion-state propagation. Persistent rollout does not provide a further improvement and yields an FVD of 832.22. The hybrid evaluation used by the full model achieves the lowest FVD of 793.10, although its LPIPS is slightly higher than those of the strict-reset and persistent variants. These results indicate that the temporal-quality gain is not attributable to persistent full-sequence motion-state memory.

Table 5: **Capacity-controlled comparison of dynamic modeling.** The residual baselines are parameter-matched to the structured SAHyD–FMint dynamics stack.

Dynamics	Params	ΔParams	CLIP ↑	LPIPS ↓	FVD ↓
Generic residual MLP	26,991	$+0.279\%$	0.922	0.118	893.25
Gated residual MLP	26,802	-0.424%	0.924	0.118	885.13
Structured dynamics	26,916	Ref.	0.924	0.120	793.10

Capacity-controlled dynamics. We next test whether the improvement can be explained by additional model capacity. We replace the structured state-dynamics stack with parameter-matched residual predictors while keeping the remaining training and evaluation settings fixed. The direct residual MLP predicts a center-state residual, while the gated variant modulates the residual with a learned sigmoid gate.

The parameter-matched alternatives differ from the structured dynamics by less than 0.5% in parameter count. Introducing a conventional residual gate improves FVD from 893.25 to 885.13, but the result remains substantially above the 793.10 obtained by the structured dynamics. At the reported precision, the gated and structured models have the same CLIP score of 0.924, while the residual predictors obtain slightly lower LPIPS. The pronounced FVD difference therefore cannot be explained by additional parameter capacity or conventional residual gating alone.

Number of latent motion regimes. We further examine whether multiple latent motion regimes provide benefits beyond a single shared regime. We compare the default $K = 6$ configuration with a $K = 1$ variant while keeping all other components, training settings, and evaluation protocols unchanged. For $K = 1$, the regime-conditioning pathway is retained but degenerates to a single shared latent regime. Table 6 summarizes the comparison.

Table 6: Effect of the number of latent motion regimes.

K	CLIP ↑	LPIPS ↓	FVD ↓
1	0.923	0.128	830.22
6	**0.924**	**0.120**	**793.10**

Compared with $K = 1$, the $K = 6$ configuration reduces FVD from 830.22 to 793.10 and LPIPS from 0.128 to 0.120, while slightly improving CLIP from 0.923 to 0.924. Since the single-regime variant retains the same conditioning pathway, the comparison indicates that the gain is not solely due to introducing an additional regime feature. Rather, allowing multiple latent regimes provides useful flexibility for representing heterogeneous motion configurations.

Soft versus hard regime inference. The main implementation uses soft regime embeddings during training and the top-1 regime embedding during inference. We therefore compare the two inference choices using the same recurrently trained model and the same persistent diagnostic rollout.

Table 7: **Soft versus hard regime inference.** Both variants use the same trained model and state-propagation setting.

Regime inference	CLIP ↑	LPIPS ↓	FVD ↓
Hard	0.919	0.118	832.22
Soft	0.919	0.118	831.84

Soft and hard inference produce nearly identical results. Their CLIP and LPIPS scores are identical at the reported precision, while soft inference reduces FVD only slightly from 832.22 to 831.84, a difference of 0.38. This indicates that the model is empirically robust to the regime-embedding choice under this controlled setting. The comparison does not imply continuity of hard regime selection or assign predefined semantic meanings to individual latent regimes.

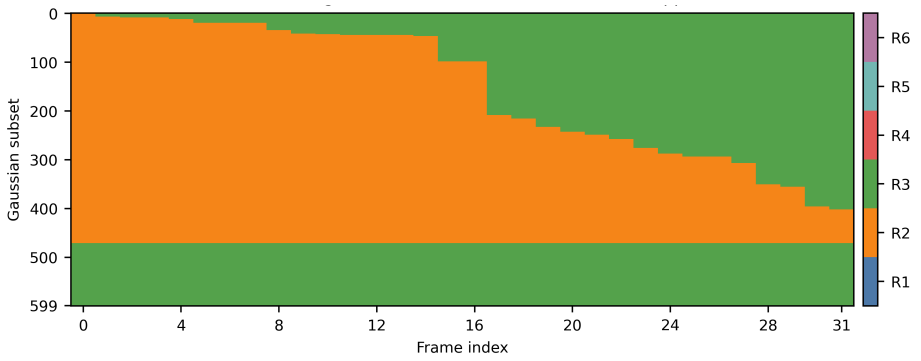


Figure 6: **Temporal variation of latent regime indices.** Top-1 regime indices for a motion-ranked subset of Gaussian primitives in the Guppie sequence. Each entry denotes $\arg\max_k p^s_{i,t,k}$.

Capacity-controlled temporal opacity modeling. Finally, we compare TVGE against a parameter-matched non-recurrent time-conditioned opacity head. The baseline directly predicts participation as $\sigma(O(\boldsymbol{\mu}_i^0, \mathbf{c}_t, t))$ and does not use a recurrent existence state, event features, or regime features.

Table 8: **Capacity-controlled comparison of temporal opacity modeling.**

Opacity model	Params	ΔParams	CLIP ↑	LPIPS ↓	FVD ↓
Time-conditioned head	8,501	+0.556%	0.920	0.120	846.47
TVGE	8,454	Ref.	0.924	0.120	793.10

Despite nearly identical parameter counts, replacing TVGE with the plain time-conditioned opacity head increases FVD from 793.10 to 846.47 and decreases CLIP from 0.924 to 0.920, while LPIPS remains unchanged at the reported precision. This comparison indicates that the benefit of TVGE is not explained by adding an opacity predictor alone and supports its stateful temporal-participation formulation as a combined design.

G Analysis of Learned Dynamics

We examine how the learned dynamic variables vary across timestamps and Gaussian primitives. The analysis covers latent motion regimes, TVGE participation probabilities, and event intensity. These variables characterize internal dynamics without assigning predefined semantic labels to regimes or event types.

Latent motion-regime evolution. Figure 6 visualizes the top-1 latent regime index for a motion-ranked subset of Gaussian primitives in the Guppie sequence. Each entry denotes $\arg\max_k p^s_{i,t,k}$ at a particular timestamp.

The top-1 regime indices vary across both primitives and timestamps: some Gaussians change their dominant index over time, whereas others remain comparatively stable. Since the visualization reports only the most probable regime index, it is used descriptively and should not be interpreted as semantic motion-regime segmentation or as evidence of high-confidence discrete assignments.

Time-varying Gaussian participation. Figure 7 summarizes the absolute TVGE participation probabilities on the Guppie sequence. We show the median over all displayed primitives, the medians of the high-motion and low-motion subsets, and the 10th–90th percentile (P10–P90) range over the Gaussian population.

High-motion Gaussians maintain higher participation probabilities than low-motion Gaussians throughout the sequence, with the overall median lying between the two. Participation also changes

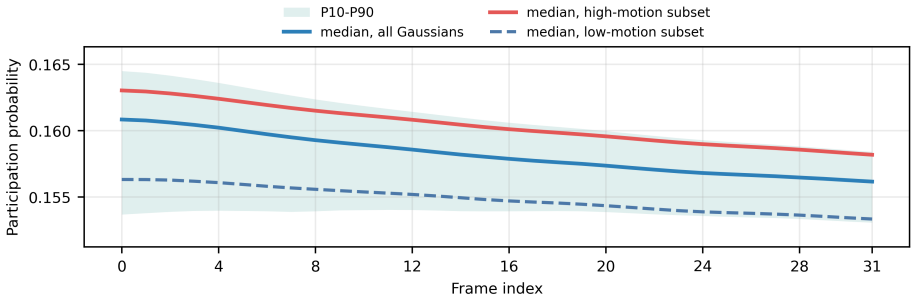


Figure 7: **Absolute temporal Gaussian participation.** TVGE participation probabilities on the Guppie sequence. Curves show the median over all displayed primitives and the medians of the high-motion and low-motion subsets, together with the P10–P90 range. High-motion Gaussians consistently have higher participation probabilities than low-motion Gaussians.

gradually over time. The persistent separation between subsets shows that TVGE learns differentiated primitive contributions through soft temporal opacity modulation.

Temporal behavior of event intensity. Figure 8 shows the event intensity $\lambda_{i,t}$ for a motion-ranked subset of Gaussian primitives in the Crocodile sequence. The heatmap preserves individual Gaussian responses and therefore reveals variation across both primitives and timestamps. For a global summary of the temporal conditioning strength, the corresponding frame-wise mean event intensity is defined as

$$\bar{\lambda}_t = \frac{1}{N} \sum_{i=1}^{N} \lambda_{i,t}, \tag{79}$$

where N denotes the number of Gaussian primitives included in the analysis. While Eq. 79 summarizes the overall temporal tendency, the per-Gaussian visualization retains the heterogeneous responses that would otherwise be averaged out.

Event intensity varies smoothly over time, with different Gaussian subsets exhibiting distinct response strengths. Regions of higher and lower intensity shift throughout the sequence rather than remaining fixed across primitives. This variation is consistent with the role of $\lambda_{i,t}$ in Eq. 3, where it adaptively weights the expected latent-event response. Accordingly, the learned intensity is interpreted as an internal, state-dependent modulation of the transition dynamics rather than an indicator of externally observed discrete events.

These visualizations provide complementary views of the learned dynamics. The top-1 regime indices vary across primitives and timestamps, TVGE produces subset-dependent participation patterns, and the event intensity captures Gaussian-dependent temporal modulation. Together with Secs. E and F, these results support the use of adaptive dynamic variables to condition Gaussian deformation and temporal participation.

H Additional Text-to-4D Results

Although our main experiments focus on video-to-4D generation, we further evaluate the proposed dynamic modeling strategy in a text-to-4D setting. Figure 9 shows representative generations from three text prompts across multiple timestamps and viewpoints.

The generated objects preserve consistent appearance and geometry across viewpoints while exhibiting temporally evolving poses and local configurations. These results suggest that our dynamic modules can be readily extended to text-conditioned 4D generation without modifying the underlying state-evolution formulation.

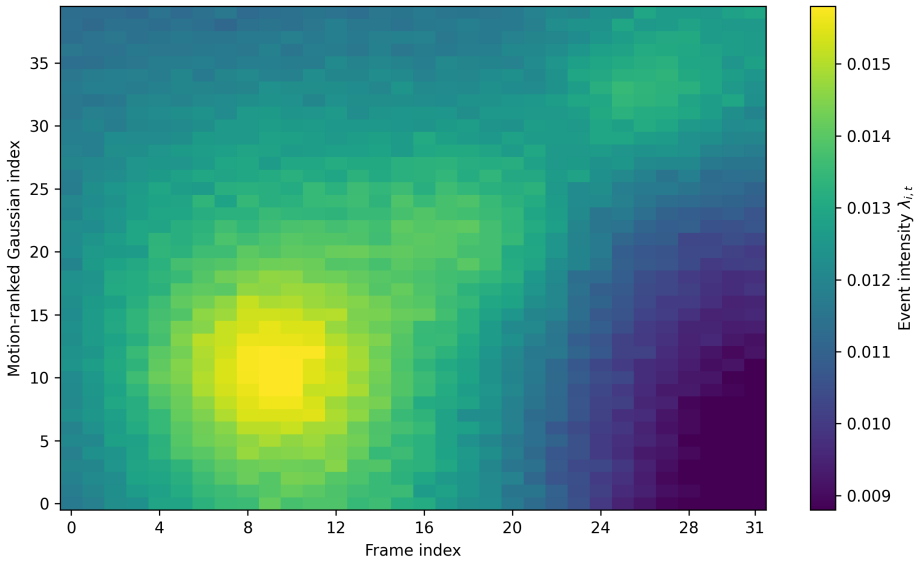


Figure 8: **Temporal variation of event intensity.** Per-Gaussian event intensity $\lambda_{i,t}$ for a motion-ranked subset of primitives in the Crocodile sequence. The heatmap shows variation in event-conditioning strength across both Gaussian primitives and timestamps.

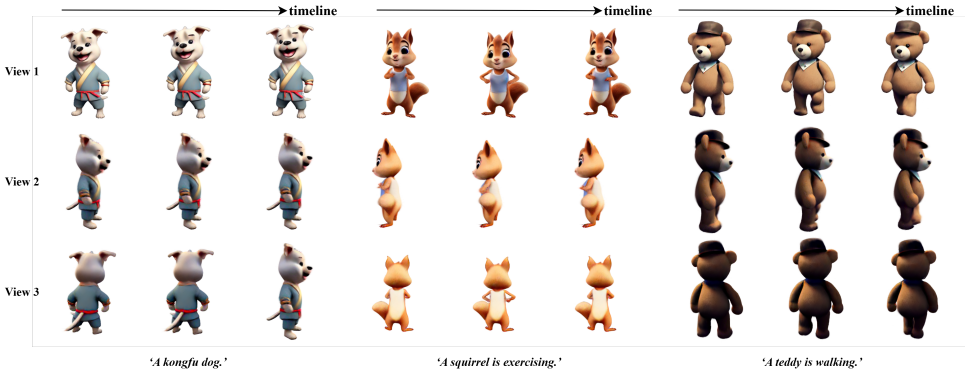


Figure 9: **Additional qualitative text-to-4D results.** Representative generations for three text prompts. Columns show different timestamps, and rows show different viewpoints. The results demonstrate coherent temporal dynamics and consistent object appearance across views.

FVD Evaluation Protocol

Our FVD evaluation follows the Consistent4D benchmark protocol used in the main comparison. For each object, we render four evaluation viewpoints, each providing one temporally ordered 32-frame clip that is encoded into a single I3D feature. Object-level FVD is computed from four predicted and four corresponding ground-truth clip features, and the final score is averaged over the seven benchmark objects. The same protocol is used for all compared methods.

Because this protocol provides relatively few clip-level samples for estimating Fréchet statistics, we interpret FVD primarily as a benchmark-compatible relative metric rather than a low-variance population estimate. We therefore report CLIP and LPIPS alongside FVD and complement the quantitative results with matched qualitative comparisons of local structural fidelity.

Structured Dynamics and Generic Residual Controls

To isolate the effect of structured dynamics from model capacity, we compare the proposed formulation with parameter-matched direct and sigmoid-gated residual MLPs under identical training and evaluation settings. Their parameter counts differ from the structured model by less than 0.5%, yet their FVD scores are 893.25 and 885.13, respectively, compared with 793.10 for the structured dynamics. These results indicate that the gain is not explained by additional capacity or conventional residual gating alone.

The event-conditioned transition is deterministic and can be written as the effective drift

$$b_\theta = f_\theta + \lambda_\theta \sum_m \rho_\theta^{(m)} J_\theta^{(m)}. \tag{80}$$

Accordingly, the proposed formulation is interpreted as a structured factorization of state-conditioned dynamics rather than a stochastic jump process. The capacity-controlled comparison provides empirical support for this structured inductive bias.

I Limitations

Our method inherits several limitations of optimization-based video-to-4D generation. First, full motion-state recurrence is restricted to finite training windows, while evaluation uses timestamp-conditioned initialization with a cached regime prior rather than persistent full-sequence state rollout, limiting explicit long-range recurrent modeling. Second, TVGE modulates the temporal participation of Gaussian primitives through soft opacity while keeping their cardinality fixed; primitive insertion and removal still rely on the inherited densification and pruning mechanisms. Reconstruction quality also depends on the monocular observations and view-conditioned guidance priors. Finally, following STAG4D, our framework requires scene-specific optimization rather than feed-forward generation.

J Computational Cost

We evaluate computational cost on the seven objects from the Consistent4D benchmark using NVIDIA L40S GPUs, with each object optimized independently on a single GPU. The STAG4D backbone requires approximately 0.50 hours per object and reaches a peak memory usage of 8.27 GiB. Enabling the proposed dynamic modules increases the average training time to 1.4 hours per object and peak memory usage to 21.89 GiB, while adding only 0.050M learnable parameters. The additional cost therefore arises primarily from temporal-state computation and intermediate activations rather than parameter storage.

This increased optimization cost does not imply persistent recurrent inference. During evaluation, the motion state is reinitialized for each queried timestamp and the deformation remains timestamp-conditioned, so full-sequence state rollout is not required. The controlled state-handling experiments further show that persistent rollout is not responsible for the observed improvement. Overall, the proposed framework incurs substantially higher optimization cost in exchange for improved temporal modeling while retaining timestamp-conditioned inference.