RateGS: Learning Motion Rates for 4D Gaussian Forecasting

Pengcheng Xu et al.
University of California, Irvine

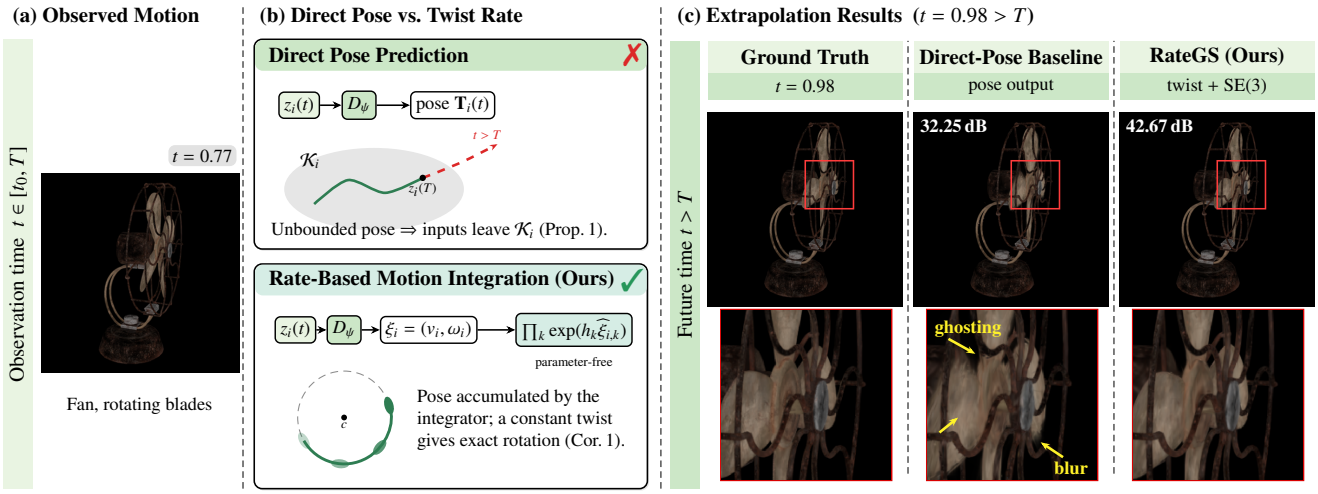

Figure 1. **Motion-rate learning for 4D Gaussian forecasting.** (a) Observed rotating-fan motion. (b) Unlike direct pose prediction, RateGS predicts instantaneous twists from latent dynamics and integrates them on $\mathrm{SE}(3)$ to recover rigid transformations. (c) Extrapolation at $t = 0.98$ beyond the training interval ($t \leq 0.75$). Compared with direct pose decoding (ParticleGS), RateGS better preserves blade geometry and produces sharper predictions.

Abstract

The ability to extrapolate dynamic 3D scenes beyond the observed timeframe is fundamental to physical world understanding and predictive modeling. Existing latent-dynamics methods learn continuous-time motion evolution but face two limitations in long-term extrapolation: accumulated-pose decoding requires out-of-distribution inputs for sustained unbounded motion, while observed-frame supervision does not directly constrain future motion-error propagation. To address these limitations, we propose RateGS, a latent-dynamics framework that separates instantaneous motion prediction from accumulated pose estimation. RateGS comprises three key components: 1) an autonomous latent evolver based on Neural Ordinary Differential Equations (Neural ODEs) that learns continuous-time dynamics without explicit time input; 2) a twist-rate decoder that predicts per-Gaussian rigid velocities (twists), which a parameter-free $\mathrm{SE}(3)$ *integrator accumulates into rigid transformations; and 3) a training-only Flow/Pose-Constrained (FC) correction that propagates observation-driven latent perturbations into twist space and regulates their temporal variation and pose sensitivity. Through this design, RateGS can represent sustained constant-twist motion at a latent equilibrium without out-of-distribution network inputs and introduces an auxiliary observation-based training signal for learned motion rates. Experiments on four benchmarks under a unified protocol evaluate its extrapolation accuracy, rendering quality, and computational cost.*

1. Introduction

Learning the dynamics of the physical world and predicting its future evolution are fundamental to predictive world modeling for embodied intelligence and autonomous systems. Given multi-view observations over $[t_0, T]$, dynamic 3D motion extrapolation aims to predict scene states at unseen future times $t > T$. Recent dynamic 3D Gaussian Splatting (3DGS) methods [10, 15, 41, 44] achieve high-

fidelity reconstruction by learning time-conditioned deformations $D(P, t)$ of canonical Gaussians P. However, reconstructing observed motion does not necessarily establish how that motion will evolve. By directly parameterizing scene configurations as functions of time, these methods rely on extrapolating learned deformation mappings beyond the observed interval, where their predictions may become unreliable [19, 34].

To enable motion extrapolation, recent approaches have introduced explicit models of motion evolution. Physics-based methods [24, 37, 42] predict future states using governing equations or physical simulation, but depend on assumptions about the underlying physical processes. Velocity-based methods instead learn motion rates and integrate them over time. NVFi [19] learns physics-regularized velocity fields, FreeGave [21] constructs velocities from a divergence-free basis, and TRACE [20] models particle velocities and accelerations. More recently, latent-dynamics approaches [34, 39] have employed Neural Ordinary Differential Equations (Neural ODEs) to model continuous-time state evolution. In particular, ParticleGS [34] evolves shared latent dynamic fields and decodes their states into accumulated Gaussian poses, achieving strong motion extrapolation performance.

Despite these advances, learning continuous-time dynamics does not necessarily resolve the limitations of the underlying motion representation. In accumulated-pose decoding, sustained translation or screw motion with nonzero axial translation produces displacement that grows without bound. Because a continuous decoder is bounded over the compact set of inputs encountered during the observation interval, reproducing such motion indefinitely requires decoder inputs outside that set, as formalized in Prop. 1. This reveals a representational mismatch between directly decoding accumulated poses and extrapolating sustained motion. A complementary challenge arises when motion is represented through instantaneous rates: even small rate errors can accumulate into substantial pose errors over long prediction horizons, while supervision limited to observed frames cannot directly constrain future error propagation. These challenges motivate separating learned motion rates from pose accumulation while introducing an observation-driven training signal.

To address these challenges, we propose RateGS, a latent-dynamics framework for 4D Gaussian forecasting that decouples motion evolution, motion-rate prediction, and geometric pose accumulation. Building on the scene encoder of ParticleGS, RateGS evolves shared latent states through an autonomous Neural ODE and decodes instantaneous per-Gaussian rigid-motion rates. Rather than directly regressing accumulated poses, it integrates predicted twists through ordered $\mathrm{SE}(3)$ exponentials to obtain rigid transformations. This formulation admits a latent-equilibrium representation of sustained constant-twist motion without requiring the decoder to leave its observation-window input set, but does not guarantee that optimization finds such an equilibrium. RateGS further introduces a training-only Flow/Pose-Constrained (FC) Twist Correction. FC derives initial latent-state perturbations from observed-frame photometric loss and propagates their first-order effect into twist space using Jacobian-vector products. Corrections are scaled by their temporal variation and pose sensitivity, checked against the observed-frame loss, and incorporated into an auxiliary training objective when accepted. FC requires no future frames or ground-truth motion rates and introduces no inference-time correction.

Our main contributions are summarized as follows:

- We identify and formalize a representational limitation of accumulated-pose decoding for sustained rigid motion with unbounded translation, and establish a conditional in-domain constant-twist representation under autonomous latent dynamics and $\mathrm{SE}(3)$ integration.
- We develop RateGS, combining autonomous latent evolution, per-Gaussian twist-rate readout, and geometric pose accumulation, together with an observation-driven FC auxiliary training mechanism.
- We evaluate motion extrapolation under a unified four-benchmark protocol and analyze the effects of latent evolution, twist readout, $\mathrm{SE}(3)$ integration, and FC correction through controlled ablations.

2. Related Work

Time-Conditioned Dynamic 3D Reconstruction. Novel view synthesis for dynamic scenes has been significantly advanced by NeRF [28] and 3D Gaussian Splatting [15]. The dominant strategy learns a deformation $D(P, t)$ from a canonical representation to the scene at time t, and methods differ mainly in how this map is parameterized: plane decompositions [4], time-aware voxels [7] or Gaussian deformation models [10, 26, 41, 43, 44]. A further line constrains the *output* of the map to rigid motions on $\mathrm{SE}(3)$, through motion bases [40], scaffolds [18], splines [31], per-primitive twists [45] or transformation fields [32], which improves the geometric validity of the reconstructed motion. These representations are designed for reconstruction within the observed interval: they constrain *what* motion is output, but not *how* it continues after T. In RateGS, $\mathrm{SE}(3)$ is not the space in which poses are regressed but the space on which learned rates are integrated.

Physical Priors and Velocity Fields for Extrapolation. Extrapolation requires a rule that determines motion after T, and existing methods differ in where this rule comes from. One line prescribes it: simulator-based methods [3, 13, 22–24, 42, 47] identify material parameters or initial states of

a physical model, and LagrangeGS [37] casts Gaussian trajectories as a non-conservative Lagrangian system. They extrapolate reliably when the assumed constitutive model, boundary conditions and forces match the scene, but the rule is only as correct as the prior. Another line learns motion *rates* from video and integrates them, so that an unbounded displacement is produced from bounded network outputs. The rate is regularized by physics in NVFi [19] and VeloGauss [25], produced by a time-conditioned network over a divergence-free basis in FreeGave [21], and given by velocities and accelerations of rigid particles in TRACE [20]. Other forecasting methods impose structural priors such as key-point graphs [46], Gaussian groups [17], Gaussian-process priors [16], future-aware fields [6] or rigid object representatives [9, 35]. RateGS shares the rate output of the second line, but obtains the rates from a latent state evolved by a learned autonomous law and integrates them on $\mathrm{SE}(3)$.

Latent Dynamics and Neural ODEs. Neural ODEs [5, 36] learn the derivative of a state and integrate it with a numerical solver, and have been applied to 4D shape dynamics [12, 29]. For dynamic Gaussians, ParticleGS [34] factorizes the scene into per-particle static features and shared dynamic fields evolved by a time-conditioned Neural ODE, ODE-GS [39] trains a Transformer latent ODE on trajectories of a frozen interpolation model, EvoGS [1] integrates a time-conditioned per-Gaussian velocity, and NeHaD [33] governs a deformation field by Hamiltonian mechanics; a concurrent study [38] reports that velocity-field ODEs extrapolate rigid structure better than articulated bodies. Beyond the expressiveness of the vector field, two design choices decide whether these models stay within their training inputs after T: whether the field depends on time, which reintroduces the time stamp as a network input, and which quantity the decoder outputs, which decides whether decoder inputs must keep changing to reproduce sustained motion. ParticleGS uses a time-dependent field and a pose output. RateGS keeps its shared latent state, removes the time input, and decodes twists integrated by a Lie-group integrator [2, 11].

Regularizing Learned Dynamics. Observations on a finite window do not identify a learned vector field outside the region its trajectories visit. Existing regularizers address this either uniformly or structurally. Kinetic-energy and Jacobian penalties [8] and penalties on higher-order solution derivatives [14] favor simple fields along the whole trajectory, independently of which observations are poorly fitted; Lyapunov-constrained models [27] guarantee stability by construction at the cost of restricting the admissible dynamics. Sensitivity analysis, used in adjoint form to train Neural ODEs [5], describes how a perturbation of the initial state propagates along a trajectory. FC uses its forward form differently: it derives the perturbation from the current observation error, restricts it to the initial state so that its effect is propagated by the learned dynamics, and scales it with a trust-region rule [30]. It is a training signal on observed frames, not a stability guarantee.

3. Methodology

We first formulate the Gaussian motion representation and establish the limitation of direct accumulated-pose decoding (Sec. 3.1). We then describe the autonomous twist-rate backbone and its geometric integrator (Sec. 3.2), followed by the training-only Flow/Pose-Constrained (FC) correction (Sec. 3.3).

3.1. Problem Formulation and Representational Analysis

Dynamic 3D Gaussian representation. Let $P = \{p_i\}_{i=1}^N$ denote canonical Gaussians, with $p_i = (x_i, q_i, s_i, \alpha_i, sh_i)$, where $x_i \in \mathbb{R}^3$ is the centre, q_i is a unit quaternion, $s_i \in \mathbb{R}^3$ denotes the scale parameters, α_i is the opacity, and sh_i contains spherical-harmonic coefficients. A differentiable renderer $\mathcal{R}$ produces

$$\hat{I}_t^v = \mathcal{R}(P_t, v), \tag{1}$$

where v specifies the camera. We observe calibrated multi-view images $\{I_t^v\}$ at times $t \in [t_0, T]$ and predict P_t for $t > T$. Time-conditioned methods learn a map $P_t = D(P, t)$ directly from the observation interval; its evaluation at $t > T$ requires extrapolation in the time coordinate.

Latent dynamics and rigid motion. Following ParticleGS [34], an encoder $E_\varphi(P)$ produces static per-Gaussian features $L = \{l_i\}_{i=1}^N$ with $l_i \in \mathbb{R}^S$ and an initial shared dynamic state $g_0 \in \mathbb{R}^{F\times G}$. The evolved state $g(t)$ forms the decoder input

$$z_i(t) = \big[l_i, \operatorname{vec} g(t)\big] \in \mathbb{R}^{S+FG}. \tag{2}$$

We express the rigid part of each Gaussian deformation by

$$\mathbf{T}_i(t) = \begin{bmatrix} Q_i(t) & b_i(t) \\ 0 & 1 \end{bmatrix} \in \mathrm{SE}(3), \qquad \mathbf{T}_i(t_0) = I_4, \tag{3}$$

so that $x_i(t) = Q_i(t)x_i + b_i(t)$. The decoder also predicts residual orientation and scale, denoted by $h_i(t) = (\delta q_i(t), \delta s_i(t))$; these heads follow ParticleGS and are distinct from the rigid-motion readout. The complete deformed Gaussian set is denoted $P_t = \mathcal{T}(P, \{\mathbf{T}_i(t), h_i(t)\}_{i=1}^N)$.

Definition 1 (Accumulated-pose and twist-rate readouts)**.** For a spatial twist $\xi = (v, \omega) \in \mathbb{R}^6$, define $[\omega]_\times y = \omega \times y$

and $\widehat{\xi} = \left[\begin{smallmatrix} [\omega]_\times & v \\ 0 & 0 \end{smallmatrix}\right] \in \mathfrak{se}(3)$. An accumulated-pose decoder returns

$$\begin{aligned} (b_i(t), \phi_i(t), h_i(t)) &= D_\psi(z_i(t)), \\ Q_i(t) &= \mathrm{Rod}(\phi_i(t)), \end{aligned} \tag{4}$$

where Rod converts a rotation vector into a rotation matrix. In contrast, a twist-rate decoder returns

$$\begin{aligned} (\xi_i(t), h_i(t)) &= D_\psi(z_i(t)), \\ \dot{\mathbf{T}}_i(t) &= \widehat{\xi_i(t)}\,\mathbf{T}_i(t), \quad \mathbf{T}_i(t_0) = I_4. \end{aligned} \tag{5}$$

Here ξ_i is a spatial (right-trivialized) twist: a point $y(t)$ rigidly attached to the Gaussian obeys $\dot{y}(t) = \omega_i(t) \times y(t) + v_i(t)$.

Limitation of accumulated-pose decoding. For each Gaussian, define the set of decoder inputs visited during the observed interval as $\mathcal{K}_i = \{z_i(t) : t \in [t_0, T]\}$. Assume the latent trajectory is continuous, so $\mathcal{K}_i$ is compact.

Proposition 1 (Out-of-set inputs for unbounded displacement)**.** *Let D_ψ be continuous. Suppose the desired rigid motion is a constant-twist trajectory $\mathbf{T}^*(t) = \exp((t-t_0)\widehat{\bar{\xi}})$ with $\bar{\xi} = (\bar{v}, \bar{\omega})$, where either (i) $\bar{\omega} = 0$ and $\bar{v} \neq 0$, or (ii) $\bar{\omega} \neq 0$ and $\bar{\omega}^\top \bar{v} \neq 0$ (screw motion). If the accumulated-pose readout in Eq. (4) reproduces $\mathbf{T}^*(t)$ for all $t \geq t_0$, then there exists a finite $t > T$ for which $z_i(t) \notin \mathcal{K}_i$.*

Proof. Continuity of D_ψ on the compact set $\mathcal{K}_i$ implies the existence of $M_i < \infty$ such that $\|b_i\| \leq M_i$ whenever $z_i \in \mathcal{K}_i$. Write the target pose as $\mathbf{T}^* = (Q^*, b^*)$. Its translation satisfies $\dot{b}^* = \bar{\omega} \times b^* + \bar{v}$ with $b^*(t_0) = 0$. In case (i), $\|b^*(t)\| = (t - t_0)\|\bar{v}\|$. In case (ii), let $u = \bar{\omega}/\|\bar{\omega}\|$; then $u^\top \dot{b}^* = u^\top \bar{v}$ and hence $\|b^*(t)\| \geq (t - t_0)|u^\top \bar{v}|$. Thus $\|b^*(t)\| \geq c(t - t_0)$ for some $c > 0$. Choose $t > \max\{T, t_0 + M_i/c\}$. The target translation exceeds every output attainable by D_ψ on $\mathcal{K}_i$, so exact reproduction requires $z_i(t) \notin \mathcal{K}_i$. □

Proposition 1 is a necessary condition for exact representation of the specified unbounded trajectories, not a claim that leaving $\mathcal{K}_i$ necessarily causes prediction failure. It does not apply to pure rotations with bounded accumulated translation, and an unbounded target twist can similarly challenge rate decoding. A complementary issue persists for rate representations: for pure translation, a constant rate error Δv yields $\Delta x(t) = (t - t_0)\Delta v$. Thus geometric integration does not itself control future error accumulation.

Framework overview. RateGS retains the encoder of ParticleGS and combines (i) autonomous latent evolution, (ii) per-Gaussian twist-rate decoding followed by SE(3) integration, and (iii) an observation-driven FC auxiliary training objective. The first two components define the inference backbone; FC is used only during training.

3.2. Autonomous Twist-Rate Dynamics

Autonomous latent evolver. ParticleGS evolves its latent state using a time-dependent vector field $\dot{g} = f_\theta(g, t)$. RateGS removes explicit time dependence:

$$\dot{g}(\tau) = f_\theta(g(\tau)), \qquad g(t_0) = g_0. \tag{6}$$

Assume f_θ is locally Lipschitz and its solution exists on the prediction interval under consideration. Then the flow is determined by the current state, without a timestamp supplied to the network. Equilibria g^* satisfying $f_\theta(g^*) = 0$ are admissible but are not enforced by this architecture. For a query time t, RK4 advances g on a grid $t_0 = \tau_0 < \cdots < \tau_K = t$ with $\Delta\tau_k = \tau_{k+1} - \tau_k \leq \Delta\tau_{\max}$. Let g_k denote the RK4 state at τ_k.

Twist-rate decoder. For each step, the decoder uses the arithmetic average of consecutive RK4 states, $\bar{g}_k = (g_k + g_{k+1})/2$, to predict

$$\xi_{i,k} = (v_{i,k}, \omega_{i,k}) = D_\psi^{\mathrm{twist}}([l_i, \mathrm{vec}\,\bar{g}_k])\,. \tag{7}$$

The two other decoder heads predict the residuals $h_i(t)$ at the requested frame, using the evolved state $g(t)$ as in ParticleGS. Let $\Xi_t = \{\xi_{i,k} : i \leq N,\ 0 \leq k < K\}$ denote the twist sequence up to t. No ground-truth twists are required.

Structure-preserving SE(3) **integration.** Holding $\xi_{i,k}$ constant on $[\tau_k, \tau_{k+1}]$ gives

$$\begin{aligned} \mathbf{T}_i(\tau_{k+1}) &= \exp\left(\Delta\tau_k \widehat{\xi_{i,k}}\right) \mathbf{T}_i(\tau_k), \\ \mathbf{T}_i(t_0) &= I_4. \end{aligned} \tag{8}$$

This is a Lie-group exponential midpoint-type discretization of Eq. (5) [2, 11]; it uses the averaged endpoint latent state to approximate the midpoint twist. Writing $\Delta\tau_k \xi_{i,k} = (\nu, \varpi)$ and $\theta = \|\varpi\|$, the matrix exponential is

$$\exp(\widehat{(\nu, \varpi)}) = \begin{bmatrix} R & V\nu \\ 0 & 1 \end{bmatrix}, \tag{9}$$

$$\begin{aligned} R &= I + \frac{\sin\theta}{\theta}[\varpi]_\times + \frac{1 - \cos\theta}{\theta^2}[\varpi]_\times^2, \\ V &= I + \frac{1 - \cos\theta}{\theta^2}[\varpi]_\times + \frac{\theta - \sin\theta}{\theta^3}[\varpi]_\times^2. \end{aligned} \tag{10}$$

The coefficients are evaluated by their continuous Taylor limits near $\theta = 0$. Each factor belongs to SE(3), so the numerical pose remains a rigid transformation up to floating-point error, without learned projection or re-orthonormalization.

Given $\mathbf{T}_i(t) = (Q_i(t), b_i(t))$, the Gaussian parameters obey

$$\begin{aligned} x_i(t) &= Q_i(t)x_i + b_i(t), \\ q_i(t) &= \mathrm{quat}(Q_i(t)) \circ \delta q_i(t) \circ q_i, \\ s_i(t) &= s_i + \delta s_i(t), \end{aligned} \tag{11}$$

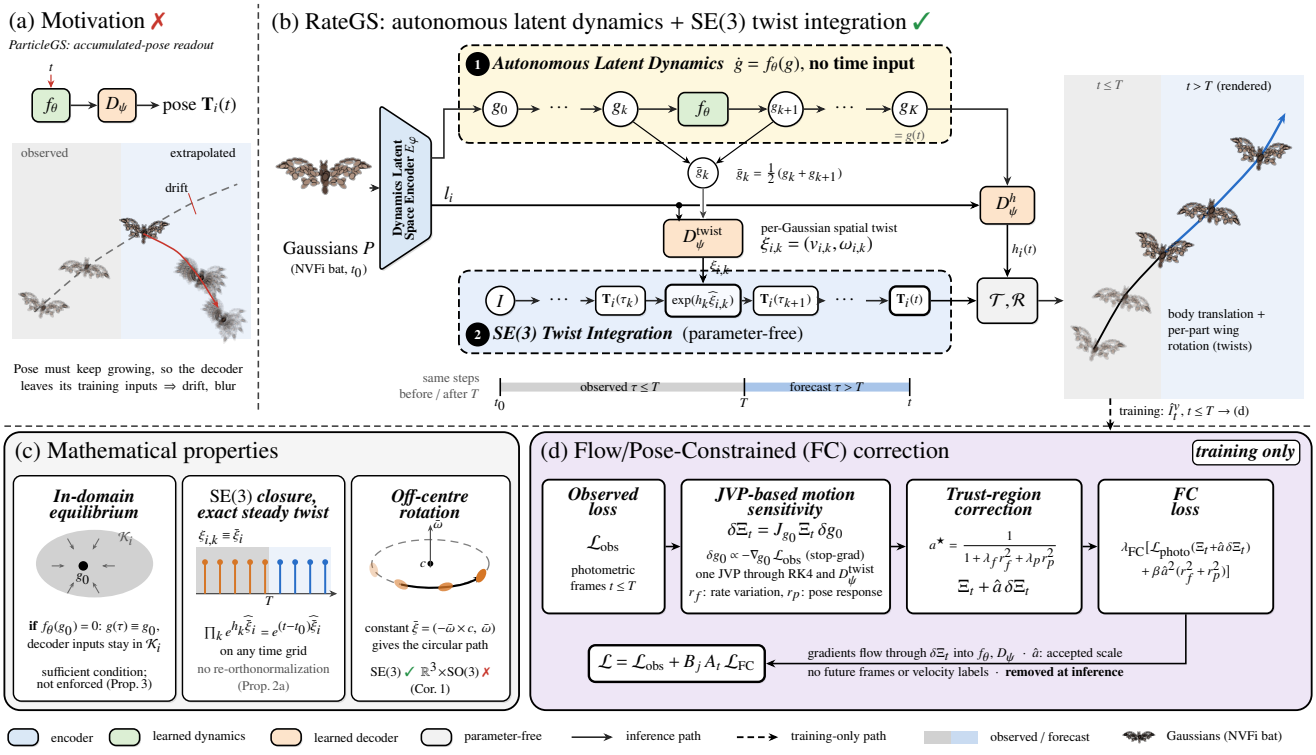

Figure 2. **Overview of RateGS.** (a) Conventional accumulated-pose decoding directly predicts rigid transformations from latent states, making sustained motion extrapolation susceptible to out-of-distribution decoder inputs. (b) RateGS evolves shared latent states through an autonomous Neural ODE, predicts instantaneous per-Gaussian twist rates, and integrates them on $\mathrm{SE}(3)$ to obtain future rigid transformations. (c) Mathematical properties of the proposed formulation, including in-domain representation of constant-twist motion under latent equilibrium, preservation of $\mathrm{SE}(3)$ structure, and coupled translation–rotation dynamics. (d) The training-only Flow/Pose-Constrained (FC) correction propagates observation-driven latent perturbations into twist space via Jacobian-vector products, scales corrections according to flow and pose sensitivities, and introduces an auxiliary objective using observed frames only.

with opacity and spherical-harmonic features handled as in ParticleGS. With $\mathbf{T}_i(t)$ obtained from Ξ_t by Eq. (8), the rendered frame is

$$\hat{I}_t^v(\Xi_t, h_t) = \mathcal{R}(\mathcal{T}(P, \Xi_t, h_t), v)\,, \qquad h_t = \{h_i(t)\}_{i=1}^N. \tag{12}$$

Here h_t is held fixed when FC tests a twist-only correction; it is not itself corrected by FC.

Proposition 2 (Exact steady twist and finite-horizon accuracy)**.** *(a) If $\xi_{i,k} = \bar{\xi}_i$ on every interval, then Eq. (8) yields exactly, in exact arithmetic, $\mathbf{T}_i(t) = \exp((t-t_0)\widehat{\bar{\xi}_i})$ on any time grid. (b) On a fixed finite time interval, suppose the latent trajectory, twist decoder and their required derivatives are sufficiently smooth, all numerical trajectories remain in a compact regularity region, and the time step tends to zero. Then the method in Eqs. (6) to (8) has global pose error $O(\Delta\tau_{\max}^2)$.*

Proof. For (a), all step generators are scalar multiples of the same $\widehat{\bar{\xi}_i}$ and therefore commute. Their product is $\exp((\sum_k \Delta\tau_k)\widehat{\bar{\xi}_i})$. For (b), RK4 has local state error $O(\Delta\tau_k^5)$. The average of exact endpoint states differs from the exact midpoint state by $O(\Delta\tau_k^2)$; using the RK4 endpoints preserves this order. Smoothness of the decoder therefore makes the midpoint twist error $O(\Delta\tau_k^2)$. The first term of the Magnus expansion of the exact group flow is the midpoint generator times $\Delta\tau_k$ up to $O(\Delta\tau_k^3)$, while higher commutator terms start at $O(\Delta\tau_k^3)$. Thus the one-step group error is $O(\Delta\tau_k^3)$. Standard stability on the fixed compact time interval gives the claimed global $O(\Delta\tau_{\max}^2)$ bound. □

Proposition 3 (In-domain equilibrium representation)**.** *Assume f_θ is locally Lipschitz and $f_\theta(g_0) = 0$. Then $g(\tau) = g_0$ for every $\tau \ge t_0$, and every RK4 stage and averaged state equals g_0. Consequently, $\xi_{i,k} = D_\psi^{\text{twist}}([l_i, \mathrm{vec}\, g_0])$ is constant and Eq. (8) represents the corresponding constant-twist trajectory for arbitrarily long finite horizons without changing any input of the evolver or twist decoder.*

Proof. The constant function $g(\tau) = g_0$ solves Eq. (6), and local uniqueness identifies it as the solution. Every RK4 stage evaluates the zero vector field at g_0, hence $g_k = \bar{g}_k = g_0$. The decoder output is constant and the first part of Prop. 2 applies. □

Why the coupled $\mathrm{SE}(3)$ exponential? A decoupled $\mathbb{R}^3 \times \mathrm{SO}(3)$ scheme can instead prescribe $\dot{x}_i = v_i$ and $\dot{Q}_i = [\omega_i]_\times Q_i$. It preserves individual rotation matrices but treats centre translation and rotation separately. The two schemes differ for rotation about an axis not passing through a Gaussian centre.

Corollary 1 (Off-centre rigid rotation)**.** *Let the desired motion be a rigid rotation about an axis through $c \in \mathbb{R}^3$ with $\bar{\omega} \neq 0$, and consider a Gaussian centre x_i not on that axis. It is represented by the constant spatial twist $\bar{\xi} = (-\bar{\omega} \times c, \bar{\omega})$ in Eq. (5). In contrast, a decoupled $\mathbb{R}^3 \times \mathrm{SO}(3)$ model with a constant centre velocity cannot reproduce the same circular centre trajectory.*

Proof. For the spatial twist, $\dot{y} = \bar{\omega} \times (y - c)$, the desired rotation equation. The resulting centre trajectory is non-constant and periodic in its component perpendicular to the axis, whereas a constant v_i in the decoupled scheme gives the affine trajectory $x_i(t) = x_i + (t - t_0)v_i$. The latter cannot equal the circular trajectory for all times. □

These propositions establish representational possibilities and geometric properties; they do not guarantee that optimization finds an equilibrium, that a learned latent trajectory remains bounded, or that extrapolated images are accurate.

3.3. Flow/Pose-Constrained Twist Correction and Optimization

Observation-driven correction. The observed-frame photometric objective is

$$\begin{aligned} \mathcal{L}_{\mathrm{obs}} &= \mathcal{L}_{\mathrm{photo}}\Big(\hat{I}_t^v(\Xi_t, h_t), I_t^v\Big), \quad t \le T, \\ \mathcal{L}_{\mathrm{photo}} &= \mathcal{L}_1 + \mathcal{L}_{\text{D-SSIM}}. \end{aligned} \tag{13}$$

Let $\gamma = \nabla_{g_0} \mathcal{L}_{\mathrm{obs}}$ be the gradient with respect to the encoder output g_0, back-propagated through the evolver, decoder, Lie-group integrator and renderer. Using $\eta > 0$ and $\varepsilon_g > 0$, define

$$\begin{aligned} \delta g_0 &= \mathrm{sg}\left(-\eta \frac{\max(\|g_0\|_F, 1)}{\max(\|\gamma\|_F, \varepsilon_g)} \gamma\right), \\ \eta &= 0.01, \quad \varepsilon_g = 10^{-12}, \end{aligned} \tag{14}$$

where sg denotes stop-gradient. This is a candidate perturbation direction, not an update applied to the latent state.

Discrete tangent propagation. For the continuous model, the tangent response $u(\tau) = \partial g(\tau)/\partial g_0 \, \delta g_0$ satisfies

$$\dot{u}(\tau) = J_{f_\theta}(g(\tau))u(\tau), \qquad u(t_0) = \delta g_0. \tag{15}$$

For the implemented RK4 solver, we obtain u_k by differentiating its discrete stages with a forward-mode Jacobian-vector product (JVP). This produces the exact directional derivative of the discrete RK4 map, up to numerical arithmetic, and is consistent with the continuous variational equation at RK4 order. At each interval,

$$\begin{aligned} \bar{u}_k &= \tfrac{1}{2}(u_k + u_{k+1}), \\ \delta\xi_{i,k} &= J_g D_\psi^{\mathrm{twist}}\left([l_i, \operatorname{vec} \bar{g}_k]\right) \operatorname{vec} \bar{u}_k. \end{aligned} \tag{16}$$

Thus $\delta\Xi_t = J_{g_0}\Xi_t \, \delta g_0$ is the first-order perturbation of the *discrete* twist sequence; the trial $\Xi_t + a\delta\Xi_t$ is not an independently estimated physical velocity field. The unperturbed latent trajectory and residual heads h_t are kept fixed during trial evaluation.

Dimensionless flow and pose sensitivities. Because linear and angular velocities have different physical units, we define a characteristic scene length $\ell > 0$ and the time-unit scale $\tau_{\mathrm{ref}} > 0$ (set $\tau_{\mathrm{ref}} = 1$ for normalized time). For a twist $\xi = (v, \omega)$, use the dimensionless representation

$$\widetilde{\xi} = \big(\tau_{\mathrm{ref}} v/\ell, \, \tau_{\mathrm{ref}}\omega\big), \quad \widetilde{\delta\xi} = \big(\tau_{\mathrm{ref}}\delta v/\ell, \, \tau_{\mathrm{ref}}\delta\omega\big). \tag{17}$$

Let $m_k = (\tau_k + \tau_{k+1})/2$ denote the interval centre and set $\Delta_m \widetilde{\delta\xi}_{i,k} = \tau_{\mathrm{ref}}(\widetilde{\delta\xi}_{i,k+1} - \widetilde{\delta\xi}_{i,k})/(m_{k+1} - m_k)$ for $k < K - 1$. Define the dimensionless pose descriptor $\widetilde{\mathcal{P}}(\Xi_t) = \{(x_i(t)/\ell, \operatorname{vec} Q_i(t)/\sqrt{2})\}_{i=1}^N$, where $(Q_i(t), x_i(t))$ is obtained by integrating Ξ_t with Eq. (8). The two sensitivity measures are

$$\begin{aligned} r_f &= \frac{\mathrm{RMS}\left(\Delta_m \widetilde{\delta\Xi}\right)}{\max\{\mathrm{RMS}(\widetilde{\Xi}_t), \varepsilon_f\}}, \\ r_p &= \frac{\mathrm{RMS}\Big(\partial_a \widetilde{\mathcal{P}}(\Xi_t + a\delta\Xi_t)\Big|_{a=0}\Big)}{\max\{\mathrm{RMS}(\widetilde{\mathcal{P}}(\Xi_t)), \varepsilon_p\}}, \end{aligned} \tag{18}$$

with $\varepsilon_f = \varepsilon_p = 10^{-3}$. Here RMS is taken over the available Gaussian, interval and component indices, and $r_f = 0$ when $K < 2$. These are normalized numerical sensitivities, not guaranteed bounds on future trajectory errors or physically invariant distances.

Constrained scale and observed-frame acceptance. Given nonnegative weights λ_f, λ_p, choose a scalar amplitude by the convex problem

$$\begin{aligned} a^* &= \operatorname*{arg\,min}_{0 \le a \le 1} \tfrac{1}{2}(a - 1)^2 + \tfrac{1}{2}(\lambda_f r_f^2 + \lambda_p r_p^2)a^2 \\ &= \frac{1}{1 + \lambda_f r_f^2 + \lambda_p r_p^2}, \quad \lambda_f = \lambda_p = 1. \end{aligned} \tag{19}$$

This rule dampens candidate changes as their sensitivities increase, but does not impose a hard upper bound on either sensitivity and does not by itself guarantee a meaningful reduction of the correction amplitude. For $\alpha \in \mathcal{A} = \{1, 1/2, 1/4, 1/8\}$, calculate

$$L_\alpha = \mathcal{L}_{\text{photo}}\Big(\hat{I}_t^v(\Xi_t + \alpha a^* \delta\Xi_t, h_t), I_t^v\Big). \tag{20}$$

Let $\alpha_{\min} \in \arg\min_{\alpha\in\mathcal{A}} L_\alpha$ (with a fixed tie-breaking rule). A correction is accepted iff $\mathcal{L}_{\text{obs}} - L_{\alpha_{\min}} > \varepsilon_{\text{acc}}$ for a specified nonnegative acceptance threshold ε_{acc}. Define the Boolean acceptance indicator A_t and the accepted scale

$$\begin{aligned} A_t &= \mathbf{1}[\mathcal{L}_{\text{obs}} - L_{\alpha_{\min}} > \varepsilon_{\text{acc}}], \\ \hat{a} &= A_t\, \alpha_{\min} a^*. \end{aligned} \tag{21}$$

This test guarantees only a reduction of the selected training-frame photometric loss at the parameters on which the test was performed. It gives no formal guarantee about future-frame loss.

Auxiliary training objective. For training iteration j, let $B_j = \mathbf{1}[j \bmod 200 = 0]$ and set

$$\mathcal{L} = \mathcal{L}_{\text{obs}} + B_j A_t \mathcal{L}_{\text{FC}}, \tag{22}$$

where, for accepted candidates,

$$\begin{aligned} \mathcal{L}_{\text{FC}} = \lambda_{\text{FC}}\big[&\mathcal{L}_{\text{photo}}\big(\hat{I}_t^v(\Xi_t + \hat{a}\delta\Xi_t, h_t), I_t^v\big) \\ &+ \beta\hat{a}^2(r_f^2 + r_p^2)\big], \end{aligned} \tag{23}$$

with $\lambda_{\text{FC}} = 0.2$ and $\beta = 0.01$. The discrete selection $A_t, \alpha_{\min}$ and the constructed perturbation δg_0 are treated as stop-gradient in this objective. The scale a^*, and hence $\hat{a}$, is also held fixed during the parameter-gradient step, while the JVP-generated $\delta\Xi_t$, base twists, and sensitivity penalty retain their parameter dependence. Therefore the auxiliary objective can differentiate through how the discrete learned dynamics transmit initial-state perturbations into the decoded rates; it is not direct supervision against ground-truth velocities. Multiplication by A_t ensures that no auxiliary photometric term is added when a correction is rejected.

Training and inference. Following ParticleGS [34], we retain geometry warm-up, a gradually expanded dynamics-training window, joint optimization, densification/pruning and online neighborhood regularization. We use $\Delta\tau_{\max} = 0.025$ for latent and pose integration. FC uses only images acquired at $t \le T$, with no future frames or motion labels; no perturbation, sensitivity evaluation, line search, or auxiliary loss is used at inference. The forward prediction is fully specified by Eqs. (6) to (8) and (11).

4. Experiments

4.1. Experimental Setup

Benchmarks and evaluation. We evaluate future-frame extrapolation on 20 scenes from four benchmarks: Dynamic Object (6) and Dynamic Indoor Scene (4) [19], Dynamic Multipart (4) [20], and FreeGave-GoPro (6) [21]. Following established protocols, the first 75% of each sequence is used for training, while the remaining 25% is reserved for extrapolation. Reconstruction is evaluated on held-out views within the observed temporal interval. We report PSNR, SSIM, and LPIPS at the original image resolution, with temporal extrapolation as the primary evaluation objective.

Baselines and implementation protocol. We compare against reconstruction-based methods (HexPlane, TiNeuVox, DeformGS, Grid4D), physics- or structure-informed methods (NVFi, GaussianPrediction), and forecasting approaches (TRACE, FreeGave, ParticleGS) [34]. Published reference results in Tabs. 1 and 2 are distinguished from our unified-protocol comparisons in Tab. 3. The latter use matched temporal splits, camera configurations, image resolutions, and evaluation metrics. Models are evaluated at their final checkpoints without test-set model selection; single-seed results are explicitly identified. Hyperparameters are selected on development data without accessing future test frames. Experiments use two NVIDIA H100 SXM GPUs, each equipped with 80 GB HBM3 memory.

4.2. Results and Analysis

Reconstruction and extrapolation. Tables 1 and 2 compare RateGS with published reference results across synthetic and real-world benchmarks. RateGS achieves the highest reconstruction PSNR on all four benchmarks. For extrapolation, it obtains 35.90, 39.92, and 28.20 dB on Indoor, Multipart, and GoPro, exceeding ParticleGS by 4.80, 3.78, and 1.41 dB, respectively. RateGS also achieves the best extrapolation SSIM and LPIPS on Indoor and GoPro, while ParticleGS retains the lowest extrapolation LPIPS on Multipart. On Dynamic Object, RateGS obtains 34.78 dB, below ParticleGS's 36.47 dB. This includes an anomalous falling-ball run (22.33 dB); excluding it yields 37.27 dB over the remaining five scenes. These cross-paper results indicate promising performance but do not establish superiority under a matched protocol.

Unified-protocol comparison and efficiency. Table 3 reports extrapolation performance under a common evaluation protocol. RateGS achieves the highest PSNR on all four benchmarks, exceeding the retrained ParticleGS by 0.15–3.48 dB and RateGS-Base by 0.53–0.60 dB. These gains entail additional computational costs: relative to

Table 1. **Quantitative comparison on Dynamic Object and Dynamic Indoor Scene at the original resolution.** RateGS achieves the highest reconstruction PSNR on both benchmarks and the best extrapolation PSNR, SSIM, and LPIPS on Indoor. Bold values indicate the best results, including ties. * denotes results affected by an anomalous falling-ball run.

	Dynamic Object Dataset (Synthetic)						Dynamic Indoor Scene Dataset (Synthetic)					
	Reconstruction			Extrapolation			Reconstruction			Extrapolation		
Method	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
HexPlane [4]	21.59	0.910	0.125	22.42	0.934	0.078	17.98	0.482	0.626	23.19	0.671	0.474
TiNeuVox [7]	20.28	0.927	0.104	19.51	0.935	0.080	23.71	0.681	0.373	21.11	0.714	0.333
DeformGS [44]	37.38	0.964	0.034	26.16	0.946	0.038	20.02	0.595	0.445	21.98	0.769	0.233
Grid4D [43]	37.58	0.961	0.034	27.67	0.949	0.080	20.55	0.590	0.442	20.63	0.736	0.284
NVFi [19]	29.39	0.969	0.047	26.82	0.968	0.047	17.12	0.494	0.669	23.72	0.672	0.502
GSPrediction [46]	36.41	0.978	0.033	23.15	0.932	0.081	16.15	0.585	0.600	20.02	0.689	0.343
TRACE [20]	38.01	0.991	0.011	33.36	0.983	0.013	22.85	0.642	0.377	29.48	0.849	0.205
FreeGave [21]	38.64	**0.992**	0.011	33.63	0.983	**0.012**	19.68	0.589	0.453	28.98	0.854	0.212
ParticleGS [34]	39.78	**0.992**	**0.009**	**36.47**	**0.984**	**0.012**	25.50	0.748	0.291	31.10	0.892	0.133
RateGS (ours)	**40.15**	0.991	0.011	34.78*	0.957*	0.037*	**27.30**	**0.805**	**0.251**	**35.90**	**0.959**	**0.083**

Table 2. **Quantitative comparison on Dynamic Multipart and FreeGave-GoPro at the original resolution.** RateGS achieves the highest PSNR and SSIM on both benchmarks and the lowest LPIPS on FreeGave-GoPro. On Multipart, reconstruction LPIPS is tied, while ParticleGS achieves the lowest extrapolation LPIPS. Bold values indicate the best results, including ties.

	Dynamic Multipart Dataset (Synthetic)						FreeGave-GoPro Dataset (Real World)					
	Reconstruction			Extrapolation			Reconstruction			Extrapolation		
Method	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
DeformGS [44]	38.43	0.988	**0.019**	27.99	0.942	0.045	20.11	0.832	0.212	21.67	0.858	0.195
TRACE [20]	36.27	0.990	0.021	33.46	0.984	0.015	20.10	0.838	0.205	25.92	0.892	0.163
FreeGave [21]	36.14	0.990	0.020	33.53	0.981	0.019	19.95	0.838	0.206	26.51	0.896	0.160
ParticleGS [34]	38.61	0.990	**0.019**	36.14	0.985	**0.014**	20.12	0.839	0.205	26.79	0.903	0.150
RateGS (ours)	**39.05**	**0.993**	**0.019**	**39.92**	**0.987**	0.019	**20.80**	**0.848**	**0.192**	**28.20**	**0.916**	**0.137**

Table 3. **Unified-protocol comparison of extrapolation accuracy and computational efficiency.** We report PSNR, per-scene training time, peak GPU memory, and inference throughput. Bold values indicate the best results, including ties. † indicates single-seed results; § includes three failed-fit runs.

	Extrapolation PSNR ↑				Computational Efficiency		
Method	Obj.	Indoor	Multi.	GoPro	h ↓	GB ↓	FPS ↑
ParticleGS [34]	35.80	33.08	36.62§	26.60	**1.8**	**12.0**	**45.0**
TRACE [20]	33.10	32.80†	33.20	25.80	2.4	16.0	28.0
FreeGave [21]	33.50	33.73†	33.60	26.40	2.1	14.0	32.0
RateGS-Base (w/o FC)	35.37	35.57	39.55	27.60	3.4	30.0	30.5
RateGS (ours)	**35.95**	**36.10**	**40.10**	**28.20**	3.2	37.0	30.5

ParticleGS, per-scene training time increases from 1.8 to 3.2 h, peak GPU memory from 12 to 37 GB, and inference throughput decreases from 45.0 to 30.5 FPS. FC is applied only during training and introduces no separate inference-time correction module.

Component analysis and limitations. A development-set ablation shows that autonomous twist-rate dynamics with $\mathrm{SE}(3)$ integration improve mean PSNR from 35.95 to 38.20 dB over accumulated-pose decoding. The unified-protocol results further indicate improvements from FC across all four benchmarks. Nevertheless, motion-rate errors may accumulate during long-horizon extrapolation, particularly under accelerated or non-stationary motion, while sequential integration introduces additional computational overhead.

5. Conclusion

We introduced RateGS, which separates motion-rate prediction from pose accumulation through autonomous latent dynamics and $\mathrm{SE}(3)$ integration. This formulation preserves rigid-motion geometry and supports sustained constant-twist motion without out-of-domain decoder inputs under latent equilibrium. Observation-driven FC further regularizes motion learning without future supervision.

LLM Usage Statement

We used large language models (LLMs) as assistive tools during the preparation of this manuscript. Specifically, LLMs were used to refine author-written text, improve clarity and organization, assist with related work and citations, refine code and implementations, support experimental design and result analysis, improve figures and visual materials, and refine the presentation of research ideas and conclusions. No LLMs were used for synthetic data generation, dataset creation, or annotation. All research contributions, scientific claims, experimental results, and conclusions were developed, reviewed, and verified by the authors, who take full responsibility for the content of this work.

References

[1] Arnold Caleb Asiimwe and Carl Vondrick. 4D gaussian splatting as a learned dynamical system. *arXiv preprint arXiv:2512.19648*, 2025. 3

[2] Sergio Blanes, Fernando Casas, José A. Oteo, and José Ros. The Magnus expansion and some of its applications. *Physics Reports*, 470(5–6):151–238, 2009. 3, 4

[3] Junhao Cai, Yuji Yang, Weihao Yuan, Yisheng He, Zilong Dong, Liefeng Bo, Hui Cheng, and Qifeng Chen. GIC: Gaussian-informed continuum for physical property identification and simulation. In *NeurIPS*, pages 75035–75063, 2024. 2

[4] Ang Cao and Justin Johnson. HexPlane: A fast representation for dynamic scenes. In *CVPR*, pages 130–141, 2023. 2, 8

[5] Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. In *NeurIPS*, 2018. 3

[6] Mingyang Ding, Zhan Wang, Jiachen Wang, Tingting Han, Xinyuan Hu, Jiajun Ding, Min Tan, and Zhenzhong Kuang. FutureGS: Structured gaussian fields for future-aware dynamic scene modeling. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 8322–8331, 2025. 3

[7] 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*, pages 1–9, 2022. 2, 8

[8] Chris Finlay, Jörn-Henrik Jacobsen, Levon Nurbekyan, and Adam M. Oberman. How to train your neural ODE: the world of Jacobian and kinetic regularization. In *International Conference on Machine Learning*, pages 3154–3164, 2020. 3

[9] Denis Gridusov, Maxim Popov, and Sergey Kolyubin. R5DGS: Semantic-aware 4D gaussian splatting with rigid body constraints for efficient dynamic scene reconstruction. *arXiv preprint arXiv:2605.25909*, 2026. 3

[10] Yi-Hua Huang, Yang-Tian Sun, Ziyi Yang, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. SC-GS: Sparse-controlled gaussian splatting for editable dynamic scenes. In *CVPR*, pages 4220–4230, 2024. 1, 2

[11] Arieh Iserles, Hans Z. Munthe-Kaas, Syvert P. Nørsett, and Antonella Zanna. Lie-group methods. *Acta Numerica*, 9: 215–365, 2000. 3, 4

[12] Boyan Jiang, Yinda Zhang, Xingkui Wei, Xiangyang Xue, and Yanwei Fu. Learning compositional representation for 4D captures with neural ODE. In *CVPR*, pages 5340–5350, 2021. 3

[13] Hanxiao Jiang, Hao-Yu Hsu, Kaifeng Zhang, Hsin-Ni Yu, Shenlong Wang, and Yunzhu Li. PhysTwin: Physics-informed reconstruction and simulation of deformable objects from videos. In *ICCV*, pages 7219–7230, 2025. 2

[14] Jacob Kelly, Jesse Bettencourt, Matthew James Johnson, and David Duvenaud. Learning differential equations that are easy to solve. In *Advances in Neural Information Processing Systems*, pages 4370–4380, 2020. 3

[15] 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. 1, 2

[16] Mijeong Kim, Jungtaek Kim, and Bohyung Han. GP-4DGS: Probabilistic 4D gaussian splatting from monocular video via variational gaussian processes. In *CVPR*, pages 33394–33403, 2026. 3

[17] Junmyeong Lee, Hoseung Choi, and Minsu Cho. Space-time forecasting of dynamic scenes with motion-aware gaussian grouping. In *CVPR*, 2026. 3

[18] Jiahui Lei, Yijia Weng, Adam Harley, Leonidas Guibas, and Kostas Daniilidis. MoSca: Dynamic gaussian fusion from casual videos via 4D motion scaffolds. In *CVPR*, pages 6165–6177, 2025. 2

[19] Jinxi Li, Ziyang Song, and Bo Yang. NVFi: Neural velocity fields for 3D physics learning from dynamic videos. In *NeurIPS*, pages 34723–34751, 2023. 2, 3, 7, 8

[20] Jinxi Li, Ziyang Song, and Bo Yang. TRACE: Learning 3D gaussian physical dynamics from multi-view videos. In *ICCV*, pages 8820–8829, 2025. 2, 3, 7, 8

[21] Jinxi Li, Ziyang Song, Siyuan Zhou, and Bo Yang. FreeGave: 3D physics learning from dynamic videos by gaussian velocity. In *CVPR*, pages 12433–12443, 2025. 2, 3, 7, 8

[22] Xuan Li, Yi-Ling Qiao, Peter Yichen Chen, Krishna Murthy Jatavallabhula, Ming Lin, Chenfanfu Jiang, and Chuang Gan. PAC-NeRF: Physics augmented continuum neural radiance fields for geometry-agnostic system identification. In *ICLR*, 2023. 2

[23] Yuchen Lin, Chenguo Lin, Jianjin Xu, and Yadong Mu. OmniPhysGS: 3D constitutive gaussians for general physics-based dynamics generation. In *ICLR*, 2025.

[24] Nengbo Lu and Minghua Pan. CausalGS: Learning physical causality of 3D dynamic scenes with gaussian representations. In *ICMR*, 2026. 2

[25] Nengbo Lu and Bin Zhao. VeloGauss: Learning physically consistent gaussian velocity fields from videos. *arXiv preprint arXiv:2605.10567*, 2026. 3

[26] 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)*, pages 800–809, 2024. 2

[27] Gaurav Manek and J. Zico Kolter. Learning stable deep dynamics models. In *NeurIPS*, 2019. 3

[28] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2

[29] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Occupancy flow: 4D reconstruction by learning particle dynamics. In *ICCV*, pages 5379–5389, 2019. 3

[30] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, 2 edition, 2006. 3

[31] Jongmin Park, Minh-Quan Viet Bui, Juan Luis Gonzalez Bello, Jaeho Moon, Jihyong Oh, and Munchurl Kim. SplineGS: Robust motion-adaptive spline for real-time dynamic 3D gaussians from monocular video. In *CVPR*, pages 26866–26875, 2025. 2

[32] Weidong Qiao, Wangmeng Zuo, and Hui Li. Lie flow: Video dynamic fields modeling and predicting with Lie algebra as geometric physics principle. *arXiv preprint arXiv:2602.21645*, 2026. 2

[33] Hai-Long Qin, Sixian Wang, Guo Lu, and Jincheng Dai. Neural Hamiltonian deformation fields for dynamic scene rendering. In *SIGGRAPH Asia 2025 Conference Papers*, 2025. 3

[34] Jinsheng Quan, Qiaowei Miao, Yichao Xu, Zizhuo Lin, Ying Li, Wei Yang, Zhihui Li, and Yawei Luo. ParticleGS: Learning neural gaussian particle dynamics from videos for prior-free physical motion extrapolation. In *CVPR*, 2026. 2, 3, 7, 8

[35] Adrian Ramlal and John S. Zelek. PersistGS: Differentiable physics for object permanence in 4D gaussian splatting. *arXiv preprint arXiv:2606.03479*, 2026. 3

[36] Yulia Rubanova, Ricky T. Q. Chen, and David Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. In *NeurIPS*, 2019. 3

[37] Shogo Sato, Takuhiro Kaneko, Shoichiro Takeda, Tomoyasu Shimada, and Riku Inoue. LagrangeGS: Non-conservative Lagrangian system on dynamic 3D gaussian splatting. *arXiv preprint arXiv:2608.22773*, 2026. 2, 3

[38] Ahmed Soltani and Oumayma Bounouh. The rigid–articulated boundary: When neural-ODE extrapolation works in 3D gaussian splatting, and why it fails. In *NeurIPS Workshop on Continual Learning in the Era of Foundation Models and Embodied Agents*, 2026. 3

[39] Daniel Wang, Patrick Rim, Tian Tian, Dong Lao, Alex Wong, and Ganesh Sundaramoorthi. ODE-GS: Latent ODEs for dynamic scene extrapolation with 3D gaussian splatting. In *ICLR*, pages 131975–131997, 2026. 2, 3

[40] Qianqian Wang, Vickie Ye, Hang Gao, Weijia Zeng, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of motion: 4D reconstruction from a single video. In *ICCV*, pages 9660–9672, 2025. 2

[41] 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 *CVPR*, pages 20310–20320, 2024. 1, 2

[42] Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. PhysGaussian: Physics-integrated 3D gaussians for generative dynamics. In *CVPR*, pages 4389–4398, 2024. 2

[43] Jiawei Xu, Zexin Fan, Jian Yang, and Jin Xie. Grid4D: 4D decomposed hash encoding for high-fidelity dynamic gaussian splatting. In *NeurIPS*, pages 123787–123811, 2024. 2, 8

[44] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3D gaussians for high-fidelity monocular dynamic scene reconstruction. In *CVPR*, pages 20331–20341, 2024. 1, 2, 8

[45] Suwoong Yeom, Joonsik Nam, Seunggyu Choi, Lucas Yunkyu Lee, Sangmin Kim, Jaesik Park, Joonsoo Kim, Kugjin Yun, Kyeongbo Kong, and Sukju Kang. TRiGS: Temporal rigid-body motion for scalable 4D gaussian splatting. *arXiv preprint arXiv:2604.00538*, 2026. 2

[46] Boming Zhao, Yuan Li, Ziyu Sun, Lin Zeng, Yujun Shen, Rui Ma, Yinda Zhang, Hujun Bao, and Zhaopeng Cui. GaussianPrediction: Dynamic 3D gaussian prediction for motion extrapolation and free view synthesis. In *ACM SIGGRAPH Conference Papers*, 2024. 3, 8

[47] Licheng Zhong, Hong-Xing Yu, Jiajun Wu, and Yunzhu Li. Reconstruction and simulation of elastic objects with spring-mass 3D gaussians. In *ECCV*, pages 407–423, 2024. 2

RateGS: Learning Motion Rates for 4D Gaussian Forecasting

Supplementary Material

Pengcheng Xu et al.
University of California, Irvine

This supplement establishes the mathematical properties of RateGS and evaluates their empirical implications. The central distinction is between learning the instantaneous motion generator and accumulating geometric displacement: autonomous latent dynamics determine spatial twists, while $\mathrm{SE}(3)$ integration reconstructs motion. During training, flow–pose sensitivity regulation controls the directional response of the decoded rates. We specify the formulation and evaluation protocol, quantify component contributions, and establish the equilibrium, convergence, and finite-horizon error properties of the resulting rigid-motion model.

A. Experimental Protocol and Reproducibility

Evaluation protocol. Given calibrated observations $(I_j, \mathcal{C}_j, t_j)$ from the observed temporal prefix, the task is to predict the Gaussian scene $\mathcal{G}(t)$ and render $\widehat{I}(t, \mathcal{C}) = \mathcal{R}(\mathcal{G}(t), \mathcal{C})$ for $t > t_{\mathrm{obs}}$, where $\mathcal{R}$ is the Gaussian renderer. We evaluate 20 scenes across Dynamic Object (6), Dynamic Indoor (4), Dynamic Multipart (4), and FreeGave-GoPro (6), with the first 75% of each sequence observed and the remaining 25% held out for forecasting. Reconstruction uses held-out views within the observed interval. Unified comparisons with ParticleGS [2] use the same temporal partitions, camera parameters, image resolutions, and PSNR, SSIM and LPIPS definitions. Hyperparameter selection follows the development protocol of the main paper.

Model configurations. RateGS-Base combines autonomous latent dynamics, twist-rate decoding, and $\mathrm{SE}(3)$ integration. RateGS additionally incorporates flow–pose-constrained training. Both configurations share the same inference architecture.

Latent dynamics and rate readout. Following latent dynamical modeling [1, 3], let $g(t) \in \mathbb{R}^{F\times G}$ be the shared state and $l_i \in \mathbb{R}^S$ the static feature of canonical Gaussian i, $i = 1, \ldots, N$. An encoder initializes g_0 and $\{l_i\}$. RateGS separates autonomous state evolution from spatial motion-rate readout:

$$\begin{aligned} \dot{g}(t) &= f_\theta(g(t)), \\ g(t_0) &= g_0, \\ \xi_i(t) &= D_\psi^{\mathrm{twist}}(l_i, g(t)) \\ &= (v_i(t), \omega_i(t)) \in \mathbb{R}^6. \end{aligned} \tag{A.1}$$

Here f_θ maps the latent space to itself, and $v_i, \omega_i \in \mathbb{R}^3$ are the linear and angular components of a spatial twist. The rigid-motion component satisfies

$$\dot{T}_i = \widehat{\xi}_i T_i, \qquad T_i(t_0) = I_4, \tag{A.2}$$

where $T_i = (Q_i, b_i) \in \mathrm{SE}(3)$, and the hat operator is defined in Eq. (F.2). With RK4 latent integration on nodes τ_k, the readout uses $\bar{g}_k = (g_k + g_{k+1})/2$, followed by the left-multiplicative exponential update in Eq. (F.5). This separates instantaneous rate prediction from accumulated displacement.

Gaussian deformation. For canonical center $x_i \in \mathbb{R}^3$, write $x_i^{\mathrm{rig}}(t) = Q_i(t)x_i + b_i(t)$. The correction rendering path additionally obtains local residual displacement $d_i(t) \in \mathbb{R}^3$ and rotation vector $\rho_i(t) \in \mathbb{R}^3$ from the nominal terminal latent state and canonical features, using zero residuals when the residual head is absent. Its rendered center and returned displacement are

$$\begin{aligned} x_i^{\mathrm{def}}(t) &= Q_i(t)(x_i + d_i(t)) + b_i(t), \\ \Delta x_i(t) &= x_i^{\mathrm{def}}(t) - x_i. \end{aligned} \tag{A.3}$$

Define $q_{\mathrm{res}}(\rho) = (\cos(\|\rho\|_2/2), \frac{1}{2}\,\mathrm{sinc}(\|\rho\|_2/2)\rho)$, with $\mathrm{sinc}(z) = \sin(z)/z$ and $\mathrm{sinc}(0) = 1$. For terminal orientation and scale readouts $r_i \in \mathbb{R}^4$ and $s_i^{\mathrm{out}} \in \mathbb{R}^3$, the deformation outputs in the correction path are

$$\begin{aligned} \Delta q_i &= \mathrm{quat}(Q_i) \circ q_{\mathrm{res}}(\rho_i) \circ r_i, \\ \mathcal{D}_i &= (\Delta x_i, \Delta q_i, s_i^{\mathrm{out}}), \end{aligned} \tag{A.4}$$

where $\circ$ denotes quaternion multiplication. These outputs are passed to the native renderer in its own attribute parameterization. In the continuous model, for differentiable d_i,

$$\dot{x}_i^{\mathrm{def}} = \omega_i \times x_i^{\mathrm{def}} + v_i + Q_i \dot{d}_i. \tag{A.5}$$

The group-preservation and rate-integration guarantees in Sec. F concern T_i and its rigid-motion component; residual deformation is accounted for separately.

B. Benchmark Forecasting Results

Cross-benchmark PSNR. Tab. B.1 summarizes extrapolation PSNR under the unified protocol of the main paper. RateGS achieves higher mean PSNR than ParticleGS on all four reported benchmarks, with 35.95 dB on Dynamic Object, 36.10 dB on Dynamic Indoor, 40.10 dB on Dynamic Multipart, and 28.20 dB on FreeGave-GoPro.

Table B.1. Forecasting PSNR (dB, ↑) across four dynamic-scene benchmarks. Values reproduce the unified-protocol PSNR in the main paper. [§] includes all runs, including three runs with unsuccessful fitting. Bold indicates the highest mean in each row.

Benchmark	ParticleGS	RateGS
Dyn. Object	35.80	**35.95**
Dyn. Indoor	33.08	**36.10**
Dyn. Multipart	36.62[§]	**40.10**
FreeGave-GoPro	26.60	**28.20**

Magnitude and distribution of improvements. Following the benchmark order in Tab. B.1, RateGS achieves PSNR gains of $+0.15$, $+3.02$, $+3.48$, and $+1.60$ dB over ParticleGS. Dynamic Multipart and Dynamic Indoor exhibit the largest absolute improvements, highlighting the empirical effectiveness of the proposed framework across structured dynamic scenes.

Connection to representation choice. Continuous motion-rate prediction links instantaneous dynamics to accumulated transformations. The controlled comparisons in Sec. C examine the contributions of autonomous latent evolution and coupled integration.

Scene-level statistical formulation. Let $q_{s,k}^{(m)}$ denote the mean held-out PSNR of method m evaluated on scene s under random seed k. The paired scene-level improvement relative to baseline b is defined as

$$\Delta_s^{(m,b)} = \frac{1}{K_s} \sum_{k=1}^{K_s} \left(q_{s,k}^{(m)} - q_{s,k}^{(b)} \right), \tag{B.1}$$

with benchmark-level scene-macro improvement given by $\Delta^{(m,b)} = S^{-1} \sum_{s=1}^{S} \Delta_s^{(m,b)}$. Here, K_s denotes the number of paired seeds and S the number of evaluated scenes, ensuring uniform scene-level weighting across configurations.

B.1. Per-Scene Forecasting Results

Table B.2 reports the forecasting accuracy of RateGS across 20 scenes from four benchmarks, following the published-reference comparison protocol in the main paper. We quantify rendering fidelity using PSNR, SSIM, and LPIPS and report benchmark-level averages with equal weighting across scenes.

C. Factorized Dynamics and Geometric Readout

Architectural configurations. Tab. C.1 evaluates five configurations spanning latent state evolution, motion parameterization, and geometric integration. Autonomous twist dynamics with $\mathrm{SE}(3)$ integration achieves the highest forecasting PSNR of 38.20 dB among the evaluated configurations.

Role of autonomous dynamics. Autonomous latent evolution parameterizes temporal dynamics through a time-independent state-transition field, consistent with intrinsic dynamics learning [3]. Under direct pose readout, autonomy improves PSNR from 35.95 to 36.48 dB ($+0.53$ dB). With twist readout and $\mathrm{SE}(3)$ integration, the corresponding gain reaches $+1.08$ dB, from 37.12 to 38.20 dB.

Role of geometric integration. Under autonomous twist dynamics, coupled $\mathrm{SE}(3)$ integration improves PSNR from 37.58 to 38.20 dB ($+0.62$ dB) relative to the evaluated $\mathbb{R}^3 \times \mathrm{SO}(3)$ update. The spatial-twist exponential captures rotation–translation coupling through the group Jacobian, providing a geometrically consistent representation of offset-center rigid motion, as formalized in Sec. F.

Decoupled kinematic reference. The decoupled model defined in the main paper evolves Gaussian centers and rotations according to $\dot{x}_i = v_i$ and $\dot{Q}_i = [\omega_i]_\times Q_i$. For piecewise constant rates its corresponding step is

$$\begin{aligned} x_{i,k+1} &= x_{i,k} + \Delta\tau_k v_{i,k}, \\ Q_{i,k+1} &= \mathrm{Exp}(\Delta\tau_k [\omega_{i,k}]_\times) Q_{i,k}. \end{aligned} \tag{C.1}$$

Here $v_{i,k}$ denotes center velocity in the decoupled model, whereas in RateGS it is the linear component of a spatial twist, yielding $\dot{x}_i^{\mathrm{rig}} = \omega_i \times x_i^{\mathrm{rig}} + v_i$. The comparison concerns these specific parameterizations. This reference describes the idealized kinematic comparison; the residual attributes are outside this comparison.

D. Flow–Pose-Constrained Twist Correction

Observation-conditioned proposal. The correction acts on the decoded motion rates during training. Let L_0 be the photometric loss of the original rendered training frame, and let $\gamma = \nabla_{g_0} L_0$. The proposal is

$$\delta g_0 = \mathrm{sg}\left(-\eta \frac{\max(\|g_0\|_F, 1)}{\max(\|\gamma\|_F, 10^{-12})} \gamma \right), \tag{D.1}$$

where $\eta = 0.01$ and sg is the stop-gradient operator. The same configured photometric objective $\mathcal{L}_{\mathrm{photo}}$, combining weighted L1 and structural dissimilarity, is used for the original image and candidate renderings. Only observed training frames are used. The correction path integrates from $t_0 = -0.01$ in normalized time.

Table B.2. **Per-scene forecasting results of RateGS.** PSNR is reported in dB. Mean denotes the arithmetic average across scenes within each benchmark.

Dataset	Scene	PSNR ↑	SSIM ↑	LPIPS ↓
Dynamic Object	Fan	41.023	0.985899	0.014937
	Fallingball	22.333	0.815428	0.153685
	Bat	35.628	0.980046	0.011525
	Shark	37.650	0.985077	0.011883
	Telescope	36.805	0.987985	0.006703
	Whale	35.236	0.985240	0.018783
	Mean	**34.78**	**0.957**	**0.036**
Dynamic Indoor	Chessboard	34.785	0.960974	0.082936
	Dining	35.365	0.960899	0.070512
	Factory	33.406	0.939885	0.087675
	Darkroom	40.044	0.974242	0.090877
	Mean	**35.90**	**0.959**	**0.083**
Dynamic Multipart	Foldingchair	36.659	0.984300	0.014357
	Hyperbolic	43.056	0.991028	0.009771
	Satellite	33.138	0.980849	0.017521
	Stove	46.822	0.992096	0.024905
	Mean	**39.92**	**0.987**	**0.017**
FreeGave-GoPro	Box	30.830	0.941000	0.112000
	Collision	23.960	0.881000	0.179000
	Hammer	30.120	0.932000	0.119000
	Mat	25.640	0.895000	0.161000
	Pen1	29.530	0.924000	0.128000
	Pen2	29.120	0.923000	0.123000
	Mean	**28.20**	**0.916**	**0.137**

Table C.1. Architectural ablation of latent dynamics, motion representation, and geometric integration on the Bat development scene. PSNR is reported in dB (↑).

Dynamics	Readout	Integration	PSNR
Non-autonomous ODE	Pose	Direct	35.95
Autonomous ODE	Pose	Direct	36.48
Non-autonomous ODE	Twist	$\mathrm{SE}(3)$	37.12
Autonomous ODE	Twist	$\mathbb{R}^3 \times \mathrm{SO}(3)$	37.58
Autonomous ODE	Twist	$\mathrm{SE}(3)$	**38.20**

Tangent propagation through the rate decoder. Let $\Psi_{\theta,k}$ be the discrete RK4 transition on $[\tau_k, \tau_{k+1}]$ and $u_0 = \delta g_0$. Differentiating the numerical trajectory yields

$$\begin{aligned} g_{k+1} &= \Psi_{\theta,k}(g_k), \\ \mathrm{vec}(u_{k+1}) &= J_{\Psi_{\theta,k}}(g_k)\,\mathrm{vec}(u_k), \\ \bar{u}_k &= (u_k + u_{k+1})/2. \end{aligned} \tag{D.2}$$

For each Gaussian, the nominal twist and directional response are

$$\begin{aligned} \xi_{i,k} &= D_\psi^{\mathrm{twist}}(l_i, \bar{g}_k), \\ \delta\xi_{i,k} &= J_g D_\psi^{\mathrm{twist}}(l_i, \bar{g}_k)\,\mathrm{vec}(\bar{u}_k). \end{aligned} \tag{D.3}$$

The static features are held fixed in these directional derivatives. Collecting the rates across $K \geq 1$ integration intervals and N Gaussians gives $\Xi_t, \delta\Xi_t \in \mathbb{R}^{K\times N\times 6}$, with

$$\mathrm{vec}(\delta\Xi_t) = J_{g_0}\,\mathrm{vec}(\Xi_t)\,\mathrm{vec}(\delta g_0). \tag{D.4}$$

Thus a single observation-driven latent direction induces a distributed perturbation of the entire decoded rate sequence. The nominal state trajectory is retained; the correction changes the twist readout before geometric integration.

Coupled flow–pose sensitivity. For an array Y with M scalar entries, let $\mathrm{RMS}(Y) = \|\,\mathrm{vec}(Y)\|_2/\sqrt{M}$. Define the detached numerical reference scales

$$\begin{aligned} \sigma_\xi &= \max\{\mathrm{RMS}(\mathrm{sg}(\Xi_t)), 10^{-3}\}, \\ \sigma_x &= \max\{\mathrm{RMS}(\mathrm{sg}(X_0)), 10^{-3}\}, \end{aligned} \tag{D.5}$$

where $X_0 = (x_i)_{i=1}^N$ contains the canonical centers. The implemented temporal operator for $K \geq 2$ is

$$(\mathcal{D}_\tau Y)_{k,i} = \frac{Y_{k+1,i} - Y_{k,i}}{\tau_{k+2} - \tau_{k+1}}, \quad 0 \leq k < K-1. \tag{D.6}$$

Let $\mathcal{V}_\tau = \mathcal{D}_\tau$ for $K \geq 2$ and let $\mathcal{V}_\tau$ be the identity for $K = 1$.

The geometric response is measured before residual deformation. For the integrated centers $X^{\mathrm{rig}}(t;\Xi_t)$ and rotations $Q(t;\Xi_t)$, define

$$\mathcal{P}(\Xi_t) = \begin{bmatrix} \mathrm{vec}(X^{\mathrm{rig}}(t;\Xi_t)) \\ \mathrm{vec}(Q(t;\Xi_t)) \end{bmatrix} \in \mathbb{R}^{12N}. \tag{D.7}$$

The directional geometric response follows by differentiating the integrated transformation with respect to a scalar perturbation coefficient:

$$\delta\mathcal{P}_t = \frac{\mathrm{d}}{\mathrm{d}a}\mathcal{P}(\Xi_t + a\delta\Xi_t)\Big|_{a=0}. \tag{D.8}$$

Equivalently, $\delta\mathcal{P}_t = J_{\mathcal{P}}(\Xi_t)\,\mathrm{vec}(\delta\Xi_t)$, where the Jacobian acts on vectorized twists. The two sensitivities used by RateGS are

$$\begin{aligned} r_f &= \frac{\mathrm{RMS}(\mathcal{V}_\tau(\delta\Xi_t))}{\sigma_\xi}, \\ r_p &= \frac{\mathrm{RMS}(\delta\mathcal{P}_t)}{\sigma_x}. \end{aligned} \tag{D.9}$$

The first term measures temporal variation of the rate perturbation (or its magnitude for a single interval); the second measures its accumulated geometric response. Both use the numerical coordinates of the implemented model, with joint component weighting rather than a physically invariant metric.

Analytical regulation of correction amplitude. For nonnegative weights λ_f, λ_p, define $S = \lambda_f r_f^2 + \lambda_p r_p^2$. RateGS balances the unit proposal with its weighted first-order response via

$$a^* = \arg\min_{|a|\leq 1} \left\{ \frac{1}{2}(a-1)^2 + \frac{1}{2}a^2 S \right\}. \tag{D.10}$$

For fixed nominal trajectory and reference scales, the directional operators are linear, so their RMS responses scale with $|a|$. This yields the quadratic sensitivity term in the objective.

Proposition D.1 (Optimal regulation and bounded directional response)**.** *For finite $S \geq 0$, the unique minimizer of Eq. (D.10) is*

$$a^* = \frac{1}{1 + \lambda_f r_f^2 + \lambda_p r_p^2}. \tag{D.11}$$

It is non-increasing in either sensitivity. For any $\beta \in [0,1]$, the effective coefficient $\hat{a} = \beta a^*$ satisfies

$$\hat{a}^2 S \leq \frac{S}{(1+S)^2} \leq \frac{1}{4}. \tag{D.12}$$

Proof. The objective equals $(1+S)(a-a^*)^2/2 + S/[2(1+S)]$, and $a^* \in (0,1]$ is feasible. Its sensitivity derivatives are $\partial a^*/\partial r_j = -2\lambda_j r_j (a^*)^2$ for $j \in \{f,p\}$. The bound follows from $\beta^2 \leq 1$ and $(1+S)^2 \geq 4S$, with equality at $S = 1$ and $\beta = 1$. □

The bound concerns the weighted directional response at the nominal sequence, not the full nonlinear pose error. The implemented weights are $\lambda_f = \lambda_p = 1$.

Observation-consistent candidate selection. For ordered coefficients $(\alpha_1, \ldots, \alpha_4) = (1, 1/2, 1/4, 1/8)$, define

$$L_j = \mathcal{L}_{\mathrm{photo}}\big(\hat{I}_t^v(\Xi_t + \alpha_j a^* \delta\Xi_t, h_t), I_t^v\big), \tag{D.13}$$

where h_t denotes the nominal terminal residual attributes; the candidate twists are integrated and composed with these attributes. Candidate rendering follows the training-stage schedule, including zero rotation and scale deformation outputs before the configured warm-up plus 2000 iterations. Let $\epsilon_{\mathrm{acc}} = \max(10^{-6}, 10^{-4}|L_0|)$. Initialize $B_0 = L_0$ and $\beta_0 = 0$, and update for $j = 1, \ldots, 4$:

$$(B_j, \beta_j) = \begin{cases} (L_j, \alpha_j), & L_j < B_{j-1} - \epsilon_{\mathrm{acc}}, \\ (B_{j-1}, \beta_{j-1}), & \text{otherwise.} \end{cases} \tag{D.14}$$

The selected coefficient is $\beta = \beta_4$, giving

$$\hat{a} = \beta a^*, \qquad \widehat{\Xi}_t = \Xi_t + \hat{a}\delta\Xi_t. \tag{D.15}$$

Whenever $\beta > 0$, the selected loss is strictly below $L_0 - \epsilon_{\mathrm{acc}}$ under identical deterministic reevaluation. The bounded directional response in Eq. (D.12) holds for every selected coefficient, including $\beta = 0$. The selection compares candidate photometric objectives at fixed parameters and does not imply monotonic loss reduction across optimizer updates.

Sensitivity-regularized learning. Let B_n^{train} indicate an active correction iteration (every 200 iterations when the correction path is invoked). The training objective augments the original objective $\mathcal{L}_{\mathrm{base}}$ by

$$\begin{aligned} \mathcal{L} &= \mathcal{L}_{\mathrm{base}} + B_n^{\mathrm{train}} \mathcal{L}_{\mathrm{corr}}, \\ \mathcal{L}_{\mathrm{corr}} &= 0.2\big[L_{\mathrm{trial}} + 0.01\hat{a}^2(r_f^2 + r_p^2)\big], \end{aligned} \tag{D.16}$$

where $L_{\mathrm{trial}} = \mathcal{L}_{\mathrm{photo}}(\hat{I}_t^v(\widehat{\Xi}_t, h_t), I_t^v)$ is recomputed through the candidate renderer. When $\beta = 0$, the auxiliary photometric term remains active and the sensitivity penalty vanishes. With the adopted weights the regularizer contribution is bounded by $0.2 \times 0.01/4$.

Differentiable regulation. The initial perturbation, normalization reference tensors and discrete β are detached in backpropagation. The directional responses and a^* retain their parameter dependence. Let ϑ denote the trainable parameters, and $d_0 = \mathrm{vec}(\delta g_0)$. The total derivative of the JVP is

$$\mathrm{D}_\vartheta \,\mathrm{vec}(\delta\Xi_t) = \big(\mathrm{D}_\vartheta J_{g_0}\,\mathrm{vec}(\Xi_t)\big)d_0. \tag{D.17}$$

This includes dependence through the nominal latent state and static features whenever they depend on the parameters. With detached references, the amplitude derivative is

$$\begin{aligned}\mathrm{D}_\vartheta \hat{a} = -2\beta(a^*)^2\big(&\lambda_f r_f \mathrm{D}_\vartheta r_f \\ &+ \lambda_p r_p \mathrm{D}_\vartheta r_p\big).\end{aligned} \tag{D.18}$$

At differentiable points the corrected sequence therefore satisfies

$$\mathrm{D}_\vartheta \widehat{\Xi}_t = \mathrm{D}_\vartheta \Xi_t + \hat{a}\mathrm{D}_\vartheta \delta\Xi_t + \delta\Xi_t\, \mathrm{D}_\vartheta \hat{a}, \tag{D.19}$$

with arrays interpreted in vectorized form. This retains mixed second derivatives through the JVP and the response of the adaptive coefficient. Inference uses the learned autonomous dynamics and native geometric reconstruction; observation-conditioned correction is a training mechanism.

D.1. First-Order Perturbation Analysis

Let $z_0 = \mathrm{vec}(g_0)$ and fix a terminal integration time and grid, indexed here by H. The latent, rate and rigid-pose mappings are $F_H^g(z_0) = \mathrm{vec}(g_H)$, $F_H^\xi(z_0) = \mathrm{vec}(\Xi_H)$, and $F_H^p = \mathcal{P}\circ F_H^\xi$. Parameters, static features and canonical centers are held fixed. By the chain rule,

$$J_{F_H^p}(z_0) = J_{\mathcal{P}}(\Xi_H) J_{F_H^\xi}(z_0). \tag{D.20}$$

Hence the directional pose response equals $J_{\mathcal{P}}(\Xi_H)\,\mathrm{vec}(\delta\Xi_H)$, and its norm is bounded by the product of the two Jacobian operator norms and $\|d_0\|_2$.

Proposition D.2 (First-order perturbation bound)**.** *Let F_H be any of these mappings, twice continuously differentiable on a neighborhood of $\{z_0 + sd_0 : 0 \le s \le 1\}$, with second derivative bounded in bilinear operator norm by M_H. Then*

$$\|F_H(z_0+d_0) - F_H(z_0) - J_{F_H}(z_0)d_0\|_2 \le \tfrac{1}{2}M_H\|d_0\|_2^2. \tag{D.21}$$

Proof. Taylor's integral remainder is $\int_0^1 (1-s)\mathrm{D}^2F_H(z_0 + sd_0)[d_0,d_0]\,\mathrm{d}s$. Its norm is at most $M_H\|d_0\|_2^2/2$. □

For a fixed unit direction u and $d_0 = \varepsilon u$, define the absolute residual $E_H^{\mathrm{abs}} = \|F_H(z_0+\varepsilon u) - F_H(z_0) - \varepsilon J_{F_H}(z_0)u\|_2$. When $J_{F_H}(z_0)u \neq 0$, the relative residual satisfies

$$E_H^{\mathrm{rel}} = \frac{E_H^{\mathrm{abs}}}{\varepsilon\|J_{F_H}(z_0)u\|_2} \le \frac{M_H\varepsilon}{2\|J_{F_H}(z_0)u\|_2}. \tag{D.22}$$

Thus absolute and relative errors are respectively $\mathcal{O}(\varepsilon^2)$ and $\mathcal{O}(\varepsilon)$ for a fixed horizon. This justifies the local directional approximation under the stated regularity assumptions.

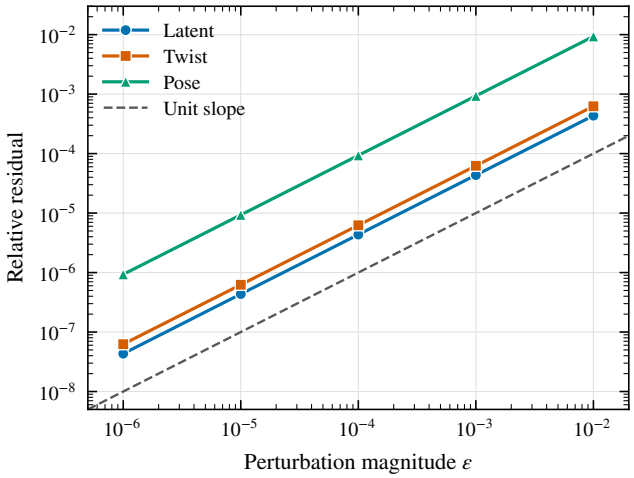

Figure D.1. **Numerical verification of first-order perturbation propagation.** Relative residuals of latent states, decoded twists, and integrated poses as functions of the initial perturbation magnitude ε at prediction horizon $H = 2.0$. The maximum integration step is 0.025. The dashed line represents the theoretical $\mathcal{O}(\varepsilon)$ scaling reference.

Numerical illustration. Figure D.1 illustrates relative linearization residuals for latent, rate and pose outputs in a controlled numerical evaluation at $H = 2.0$, maximum step 0.025, and perturbations between 10^{-6} and 10^{-2}. The curves assess the scaling of the directional approximation in the numerical test; they do not establish the magnitude of the response for all trained scenes.

D.2. Sensitivity-Constrained Correction Analysis

We compare constrained and unconstrained correction on Fan using two matched seeds and 210 extrapolation views per run. The mean paired improvement gives equal weight to each seed, taking constrained minus unconstrained PSNR and SSIM, and the reverse difference for LPIPS.

Across the two seed pairs, sensitivity regulation improves PSNR by 2.63 and 3.72 dB, SSIM by 0.0059 and 0.0089, and reduces LPIPS by 0.00236 and 0.00389, respectively (Tab. D.1). The mean PSNR gain is approximately 3.18 dB. These matched comparisons provide scene-specific evidence for the proposed sensitivity-dependent

Table D.1. **Effect of sensitivity-constrained correction on motion forecasting.** Paired evaluation on the Fan scene across two random seeds, with 210 extrapolation views per run. PSNR is expressed in dB. Mean values are computed across random seeds. Bold indicates the better result within each seed pair and across the corresponding means.

Seed	Correction	PSNR ↑	SSIM ↑	LPIPS ↓
101	Unconstrained	38.51	0.9803	0.02462
	Constrained	**41.14**	**0.9862**	**0.02226**
202	Unconstrained	36.16	0.9743	0.02712
	Constrained	**39.88**	**0.9832**	**0.02323**
Mean	Unconstrained	37.33	0.9773	0.02587
	Constrained	**40.51**	**0.9847**	**0.02275**

correction. In a separate paired configuration on Fan, constrained versus unconstrained correction yields 38.54 versus 38.00 dB, 0.98047 versus 0.97913 SSIM, and 0.02436 versus 0.02553 LPIPS. The configurations are reported separately.

E. Long-Horizon Forecasting Behavior

Forecasting protocol. We characterize the temporal evolution of learned motion dynamics through complementary observations in image, transformation, and latent spaces. Following the unified evaluation protocol, the first 75% of each temporal sequence constitutes the observed temporal prefix, and the remaining 25% defines the held-out forecasting interval $(t_{\text{obs}}, t_{\text{end}}]$.

For a forecasting timestamp $t > t_{\text{obs}}$, we define the prediction horizon as

$$H = t - t_{\text{obs}}. \tag{E.1}$$

Image-space forecasting fidelity is quantified by PSNR at timestamps with corresponding reference observations. The aggregate forecasting results under the unified evaluation protocol are reported in Tab. B.1.

Figure E.1 presents the temporal forecasting PSNR visualization, complementing the aggregate benchmark results.

Temporal forecasting fidelity. Figure E.2 characterizes the forecasting performance of RateGS-Base across increasing prediction horizons on Dynamic Indoor and Dynamic Multipart. Relative to ParticleGS, RateGS-Base maintains higher PSNR and SSIM throughout the evaluated interval, achieving final-timestamp PSNR gains of 2.7 and 2.9 dB, respectively.

Motion-rate preservation. Figure E.3 shows that RateGS maintains stable angular and linear motion rates on Fan and better preserves linear motion rates on Foldingchair during extended forecasting. These results support the central design principle of RateGS: separating motion-rate prediction from pose accumulation helps preserve learned motion dynamics beyond the observed temporal interval.

Geometric trajectory diagnostics. To characterize the evolution of rigid transformations, we define complementary translational and rotational deviations in the transformation space. Let

$$T(t) = \begin{bmatrix} R(t) & p(t) \\ 0_{1\times 3} & 1 \end{bmatrix} \in \mathrm{SE}(3) \tag{E.2}$$

denote the predicted transformation, and let $T_{\text{ref}}(t) \in \mathrm{SE}(3)$ denote its corresponding reference transformation, expressed in the same coordinate system and relative to the same canonical frame. Here, $R(t), R_{\text{ref}}(t) \in \mathrm{SO}(3)$ and $p(t), p_{\text{ref}}(t) \in \mathbb{R}^3$.

The translational deviation and intrinsic rotational deviation are defined as

$$\begin{aligned} E_p(t) &= \|p(t) - p_{\text{ref}}(t)\|_2, \\ E_R(t) &= \left\| \mathrm{Log}\big(R_{\text{ref}}(t)^\top R(t)\big)^\vee \right\|_2. \end{aligned} \tag{E.3}$$

The translational quantity $E_p(t)$ measures the Euclidean deviation between the translation components of the two rigid transformations. For a canonical Gaussian center x_i, the corresponding rigid-motion component of the center is

$$x_i^{\text{rig}}(t) = R_i(t)x_i + p_i(t). \tag{E.4}$$

Thus, the translation component and the transformed Gaussian center represent distinct geometric quantities.

The rotational quantity $E_R(t)$ corresponds to the minimum rotation angle relating the predicted and reference orientations. Here, Log denotes a minimum-norm Lie algebra logarithm on $\mathrm{SO}(3)$, and $(\cdot)^\vee : \mathfrak{so}(3) \to \mathbb{R}^3$ is the inverse of the rotational hat operator.

Defining the relative rotation as

$$R_{\text{rel}}(t) = R_{\text{ref}}(t)^\top R(t), \tag{E.5}$$

the equivalent angular representation is

$$E_R(t) = \arccos\left(\mathrm{clip}\left(\frac{\mathrm{tr}(R_{\text{rel}}(t)) - 1}{2}, -1, 1\right)\right), \tag{E.6}$$

where $\mathrm{clip}(a, -1, 1) = \min\{1, \max\{-1, a\}\}$. The rotational deviation takes values in $[0, \pi]$ and is expressed in radians. At a rotation angle of π, the logarithm admits equivalent axis representations with opposite signs, while the angular magnitude remains uniquely defined. The clipping operation ensures numerical consistency with the domain of the inverse cosine.

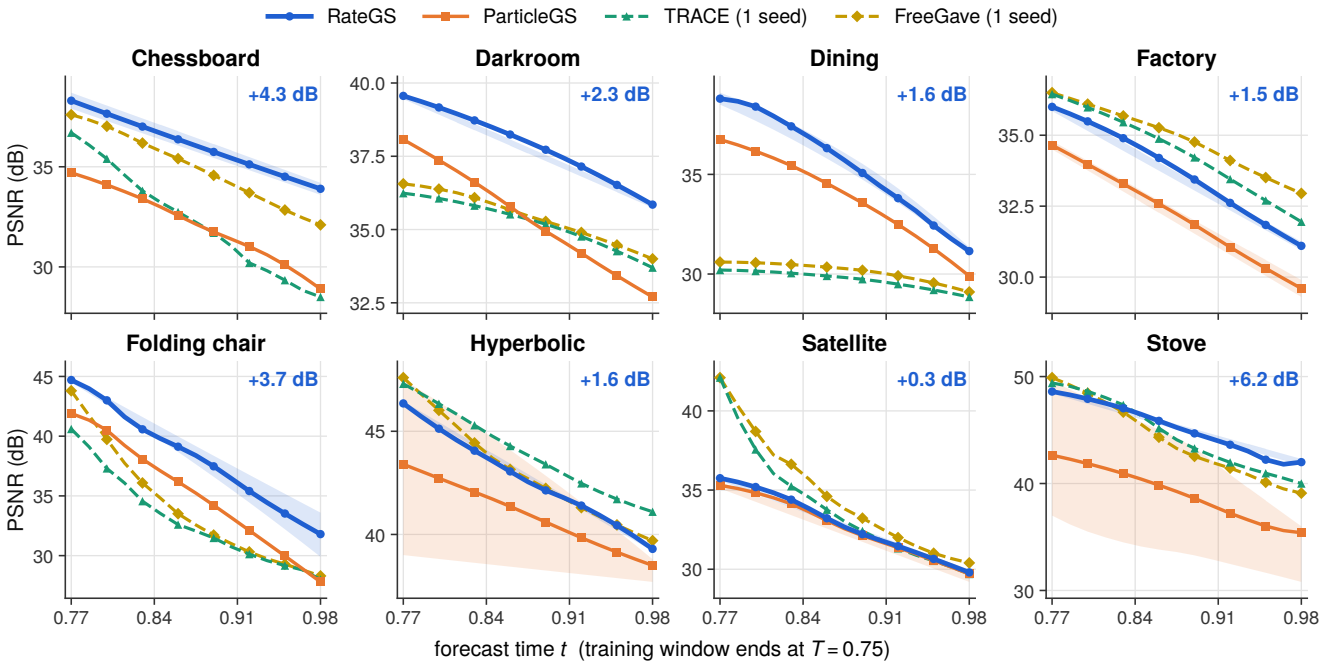

Figure E.1. **Forecasting fidelity over time.** Temporal visualization of forecasting PSNR over the evaluated prediction interval.

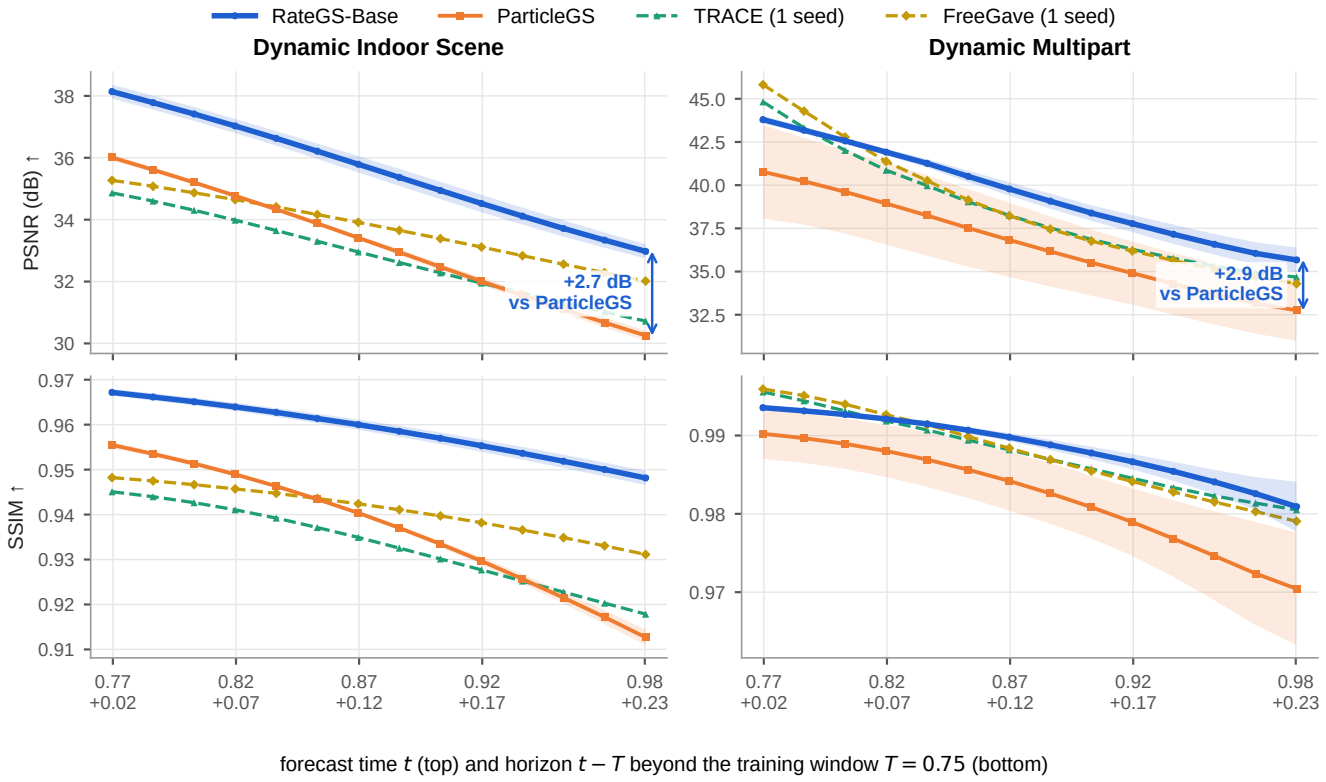

Figure E.2. **Temporal forecasting fidelity of RateGS-Base.** PSNR and SSIM as functions of forecast time on Dynamic Indoor and Dynamic Multipart, with the observation interval ending at $T = 0.75$. Annotated gains denote final-timestamp PSNR improvements relative to ParticleGS. TRACE and FreeGave are evaluated using one seed.

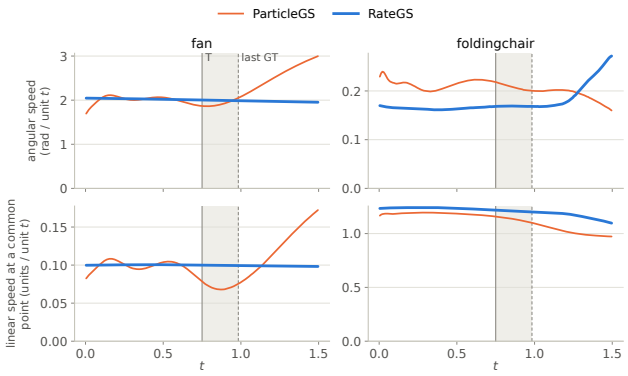

Figure E.3. **Motion-rate preservation during forecasting.** Estimated dominant angular and linear motion rates on Fan and Foldingchair using one matched random seed. Solid and dashed vertical lines indicate the observation boundary and the last reference timestamp, respectively.

Latent-state evolution. We further define a normalized displacement observable to characterize the evolution of the learned latent state relative to the end of the observation interval. Let $g(t) \in \mathbb{R}^{F\times G}$ denote the shared dynamic state governed by the autonomous differential equation

$$\dot{g}(t) = f_\theta(g(t)). \tag{E.7}$$

For a fixed regularization constant $\epsilon_0 > 0$, the normalized latent displacement is defined as

$$D_g(t) = \frac{\|g(t) - g(t_{\text{obs}})\|_F}{\max\{\|g(t_{\text{obs}})\|_F, \epsilon_0\}}. \tag{E.8}$$

This dimensionless observable quantifies the magnitude of latent-state evolution relative to a fixed reference state, with $D_g(t_{\text{obs}}) = 0$.

By integrating the autonomous latent dynamics, the displacement admits the representation

$$g(t) - g(t_{\text{obs}}) = \int_{t_{\text{obs}}}^{t} f_\theta(g(s))\,\mathrm{d}s. \tag{E.9}$$

Applying the triangle inequality yields

$$D_g(t) \le \frac{\displaystyle\int_{t_{\text{obs}}}^{t} \|f_\theta(g(s))\|_F\,\mathrm{d}s}{\max\{\|g(t_{\text{obs}})\|_F, \epsilon_0\}}. \tag{E.10}$$

This relationship connects normalized latent displacement to the accumulated magnitude of the latent vector field along the forecasting trajectory.

At a latent equilibrium $g_\star$ satisfying $f_\theta(g_\star) = 0$ and $g(t_{\text{obs}}) = g_\star$, the latent displacement satisfies

$$D_g(t) = 0, \qquad t \ge t_{\text{obs}}, \tag{E.11}$$

under the uniqueness condition established in Proposition F.1. The corresponding spatial twist remains constant, generating the exact rigid trajectory characterized by the same proposition.

Complementary diagnostic interpretation. These observables characterize distinct aspects of continuous-time forecasting. PSNR measures rendered image fidelity relative to reference observations. The geometric deviations E_p and E_R quantify translational and rotational agreement whenever corresponding reference transformations are available. The normalized latent displacement D_g characterizes internal state evolution within a fixed latent parameterization; it is not by itself an out-of-distribution or accuracy measure.

Together, these quantities provide a complementary description of forecasting behavior across observation, transformation, and latent spaces, connecting the rendered predictions to the underlying motion-rate representation and its geometric integration.

F. Mathematical Properties

F.1. Geometric Formulation of Motion on $\mathrm{SE}(3)$

We formulate motion as the continuous evolution of rigid transformations driven by spatial twists. Let

$$T(t) = \begin{bmatrix} R(t) & p(t) \\ 0_{1\times 3} & 1 \end{bmatrix} \in \mathrm{SE}(3), \tag{F.1}$$

where $R(t) \in \mathrm{SO}(3)$ and $p(t) \in \mathbb{R}^3$ denote the rotation and translation, respectively. A spatial twist $\xi(t) = (v(t), \omega(t)) \in \mathbb{R}^6$, expressed in the world frame, defines an element of the Lie algebra $\mathfrak{se}(3)$ through the hat operator:

$$\widehat{\xi}(t) = \begin{bmatrix} [\omega(t)]_\times & v(t) \\ 0_{1\times 3} & 0 \end{bmatrix}, \qquad \dot{T}(t) = \widehat{\xi}(t)T(t), \tag{F.2}$$

where $[\omega]_\times \in \mathfrak{so}(3)$ is the skew-symmetric matrix satisfying $[\omega]_\times x = \omega \times x$ for any $x \in \mathbb{R}^3$.

The corresponding rotational and translational kinematics follow directly from matrix multiplication:

$$\dot{R} = [\omega]_\times R, \qquad \dot{p} = [\omega]_\times p + v. \tag{F.3}$$

Here, ω represents the spatial angular velocity, whereas $v = \dot{p} - [\omega]_\times p$ denotes the translational component of the spatial twist. Together, these quantities characterize the instantaneous generator of rigid motion.

Let g_k and g_{k+1} denote consecutive latent states obtained through fourth-order Runge–Kutta integration at times τ_k and τ_{k+1}, respectively. For Gaussian i, we evaluate the twist readout at the arithmetic midpoint of these states:

$$\bar{g}_k = \frac{g_k + g_{k+1}}{2}, \qquad \xi_{i,k} = D_\psi^{\text{twist}}(l_i, \bar{g}_k), \tag{F.4}$$

where l_i is the fixed Gaussian-specific feature.

With the step size $h_k = \Delta\tau_k = \tau_{k+1} - \tau_k$, the rigid transformation evolves through the left-multiplicative exponential update

$$T_{k+1} = \mathrm{Exp}\Big(h_k \widehat{\xi}_k\Big)\, T_k, \tag{F.5}$$

where the Gaussian index is omitted for clarity. This construction maps the decoded motion rates from the Lie algebra to finite transformations on the Lie group.

For a constant spatial twist $\xi = (v, \omega)$ over an interval of duration h, the exact group increment is given by

$$\mathrm{Exp}(h\widehat{\xi}) = \begin{bmatrix} \mathrm{Exp}(h[\omega]_\times) & J(h\omega)(hv) \\ 0_{1\times 3} & 1 \end{bmatrix}, \tag{F.6}$$

where $J : \mathbb{R}^3 \to \mathbb{R}^{3\times 3}$ is the left Jacobian of $\mathrm{SO}(3)$,

$$J(\phi) = I_3 + \frac{1-\cos\theta}{\theta^2}[\phi]_\times + \frac{\theta - \sin\theta}{\theta^3}[\phi]_\times^2, \tag{F.7}$$

with $\theta = \|\phi\|_2$ and the continuous extension $J(0) = I_3$.

The translational component follows from integrating the rotation-dependent velocity contribution:

$$\int_0^h \mathrm{Exp}(s[\omega]_\times) v \, \mathrm{d}s = J(h\omega)(hv). \tag{F.8}$$

Consequently, the exponential map incorporates the rotation–translation coupling prescribed by the spatial kinematics within each group increment.

F.2. Equilibrium-Induced Motion

The rate-based formulation establishes a direct correspondence between equilibria in the latent dynamical system and constant-twist trajectories in the transformation space. This correspondence provides an exact characterization of persistent motion generated by stationary latent representations.

Proposition F.1 (Exact motion at latent equilibria)**.** *Consider the autonomous latent dynamical system*

$$\dot{g}(t) = f_\theta(g(t)), \tag{F.9}$$

where f_θ is locally Lipschitz continuous. Let $g_\star$ be an equilibrium satisfying $f_\theta(g_\star) = 0$, and suppose that $g(t_0) = g_\star$. For a fixed Gaussian-specific feature l_i and a time-independent twist readout, the decoded spatial twist remains constant:

$$\xi_i(t) = D_\psi^{\mathrm{twist}}(l_i, g_\star) := \xi_{i,\star}, \qquad t \geq t_0. \tag{F.10}$$

The resulting transformation satisfies

$$T_i(t) = \mathrm{Exp}\Big((t-t_0)\widehat{\xi}_{i,\star}\Big)\, T_i(t_0). \tag{F.11}$$

Moreover, RK4 integration of the latent dynamics combined with the exponential update in Eq. (F.5) reproduces this trajectory exactly at every integration node in exact arithmetic, for any strictly increasing time grid $t_0 = \tau_0 < \tau_1 < \cdots < \tau_n$.

Proof. Since $f_\theta(g_\star) = 0$, the constant function $g(t) = g_\star$ satisfies the latent differential equation and its initial condition. Local Lipschitz continuity ensures uniqueness of the solution, establishing $g(t) = g_\star$ for all $t \geq t_0$.

The fixed feature l_i and time-independent readout therefore yield $\xi_i(t) = \xi_{i,\star}$. The transformation dynamics reduce to the constant-coefficient matrix differential equation

$$\dot{T}_i(t) = \widehat{\xi}_{i,\star} T_i(t), \tag{F.12}$$

whose solution is

$$T_i(t) = \mathrm{Exp}\Big((t-t_0)\widehat{\xi}_{i,\star}\Big)\, T_i(t_0). \tag{F.13}$$

For the discrete latent dynamics, every RK4 stage evaluated at $g_\star$ equals zero. Induction over the integration steps gives

$$g_k = \bar{g}_k = g_\star, \qquad \xi_{i,k} = \xi_{i,\star}. \tag{F.14}$$

Repeated application of Eq. (F.5) then yields

$$\begin{aligned} T_{i,n} &= \left(\prod_{k=n-1}^{0} \mathrm{Exp}(h_k \widehat{\xi}_{i,\star})\right) T_{i,0} \\ &= \mathrm{Exp}\left(\sum_{k=0}^{n-1} h_k \widehat{\xi}_{i,\star}\right) T_{i,0} \\ &= \mathrm{Exp}\Big((\tau_n - t_0)\widehat{\xi}_{i,\star}\Big)\, T_{i,0}. \end{aligned} \tag{F.15}$$

The second equality follows because all group increments are generated by the same Lie algebra element and therefore commute. The final equality follows from $\sum_{k=0}^{n-1} h_k = \tau_n - t_0$. Thus, the discrete trajectory coincides with the continuous solution at every integration node. □

This result identifies latent equilibria as a representation of persistent constant-twist motion. The stationary latent state determines the motion generator, while the exponential evolution accumulates the corresponding rigid displacement over time. A time-independent direct pose readout at the same latent equilibrium would instead be constant; rate readout can represent nonzero persistent motion with stationary latent input. This is a representational property, conditional on an equilibrium, rather than a claim that training necessarily reaches one.

F.3. Preservation of Rigid Geometry

Proposition F.2 (Preservation of rigid transformations)**.** *For $T_0 \in \mathrm{SE}(3)$ and finite spatial twists, the update in Eq. (F.5) gives $T_k \in \mathrm{SE}(3)$ at every step in exact arithmetic.*

Proof. Each $h_k\widehat{\xi}_k$ belongs to $\mathfrak{se}(3)$, so its exponential belongs to $\mathrm{SE}(3)$. Closure under multiplication proves the claim by induction. □

In particular, $R_k^\top R_k = I_3$ and $\det R_k = 1$. Each individual transform preserves distances between points acted upon by that same transform; this is a property of the rigid component, distinct from the full residual Gaussian deformation.

F.4. Second-Order Convergence of Geometric Integration

The temporal discretization combines fourth-order Runge–Kutta integration of the latent dynamics with exponential integration of the decoded spatial twists. We characterize the convergence of this coupled numerical scheme by examining the propagation of latent-state approximation errors through the twist readout and the geometric evolution on $\mathrm{SE}(3)$.

Proposition F.3 (Second-order geometric convergence)**.** *Consider the continuous dynamics*

$$\dot{g}(t) = f_\theta(g(t)), \qquad \dot{T}(t) = \widehat{\xi}(t)T(t), \tag{F.16}$$

where

$$\xi(t) = D_\psi^{\mathrm{twist}}(l_i, g(t)). \tag{F.17}$$

Assume that the exact solution exists on a fixed interval $[t_0, t_f]$, the latent vector field f_θ is five times continuously differentiable, and the twist readout is twice continuously differentiable on a neighborhood of the latent trajectory, with uniformly bounded derivatives.

Let $t_0 = \tau_0 < \tau_1 < \cdots < \tau_K = t_f$ be an integration grid, with step sizes $h_k = \tau_{k+1} - \tau_k$ and $h_{\max} = \max_k h_k$. Given exact initial conditions, define the numerical latent states using RK4 and the decoded midpoint twist as

$$\xi_k = D_\psi^{\mathrm{twist}}\left(l_i, \frac{g_k + g_{k+1}}{2}\right). \tag{F.18}$$

Then the exponential update

$$T_{k+1} = \mathrm{Exp}(h_k\widehat{\xi}_k)T_k \tag{F.19}$$

has local truncation error $\mathcal{O}(h_k^3)$ when initialized at the exact latent state and transformation. Moreover, there exists a constant $C > 0$, independent of the integration grid, such that

$$\max_{0\le k\le K} \|T_k - T(\tau_k)\| \le Ch_{\max}^2 \tag{F.20}$$

for sufficiently small $h_{\max}$, where $\|\cdot\|$ denotes an induced matrix norm.

Proof. We establish the convergence result by considering the latent-state approximation, the local group increment, and the accumulation of integration errors.

Latent-state and midpoint approximation. Under the stated regularity conditions, RK4 yields the uniform latent-state error

$$\max_{0\le k\le K} \|g_k - g(\tau_k)\|_F = \mathcal{O}(h_{\max}^4). \tag{F.21}$$

Let $m_k = (\tau_k + \tau_{k+1})/2$ denote the temporal midpoint of the k-th integration interval. Taylor expansion of the exact latent trajectory about m_k gives

$$\frac{g(\tau_k) + g(\tau_{k+1})}{2} = g(m_k) + \mathcal{O}(h_k^2). \tag{F.22}$$

Combining this approximation with the RK4 state error yields

$$\bar{g}_k = g(m_k) + \mathcal{O}(h_k^2 + h_{\max}^4), \tag{F.23}$$

where $\bar{g}_k = (g_k + g_{k+1})/2$.

The local Lipschitz continuity of the twist readout and the linearity of the hat operator therefore imply

$$\|\widehat{\xi}_k - \widehat{\xi}(m_k)\| = \mathcal{O}(h_k^2 + h_{\max}^4). \tag{F.24}$$

For a single step initialized at the exact latent state, the RK4 endpoint error is $\mathcal{O}(h_k^5)$, yielding a local midpoint twist approximation of order $\mathcal{O}(h_k^2)$.

Local accuracy of the group increment. Define the continuous motion generator as $A(t) = \widehat{\xi}(t) \in \mathfrak{se}(3)$. The exact propagator U_k over $[\tau_k, \tau_{k+1}]$ satisfies

$$T(\tau_{k+1}) = U_k T(\tau_k). \tag{F.25}$$

For sufficiently small h_k, the Magnus expansion expresses the propagator as

$$U_k = \mathrm{Exp}(\Omega_k), \tag{F.26}$$

where

$$\begin{aligned}\Omega_k = &\int_{\tau_k}^{\tau_{k+1}} A(s)\,\mathrm{d}s \\ &+ \frac{1}{2}\int_{\tau_k}^{\tau_{k+1}}\int_{\tau_k}^{s} [A(s), A(r)]\,\mathrm{d}r\,\mathrm{d}s \\ &+ \mathcal{O}(h_k^3).\end{aligned} \tag{F.27}$$

Here, $[A, B] = AB - BA$ denotes the matrix commutator.

The midpoint quadrature formula gives

$$\int_{\tau_k}^{\tau_{k+1}} A(s)\,\mathrm{d}s = h_k A(m_k) + \mathcal{O}(h_k^3). \tag{F.28}$$

Furthermore, boundedness of A and its first derivative implies

$$\|[A(s), A(r)]\| \le C_1|s - r|. \tag{F.29}$$

Consequently, the double-integral commutator contribution is $\mathcal{O}(h_k^3)$, and hence

$$\Omega_k = h_k A(m_k) + \mathcal{O}(h_k^3). \tag{F.30}$$

Let the numerical group increment be

$$V_k = \mathrm{Exp}(h_k \widehat{\xi}_k). \tag{F.31}$$

Using Eq. (F.24) and the local Lipschitz continuity of the matrix exponential, we obtain

$$\|V_k - U_k\| \le C_2(h_k^3 + h_k h_{\max}^4). \tag{F.32}$$

For a step initialized at the exact latent state, the corresponding group increment defect is $\mathcal{O}(h_k^3)$. Since the exact transformation is bounded on the fixed integration interval, the local transformation error is also $\mathcal{O}(h_k^3)$.

Global convergence. Define the transformation error at each integration node as

$$E_k = T_k - T(\tau_k). \tag{F.33}$$

The numerical and exact evolution equations give

$$E_{k+1} = V_k E_k + (V_k - U_k)T(\tau_k). \tag{F.34}$$

Uniform boundedness of the decoded generators implies the existence of $M > 0$ such that

$$\|V_k\| \le \exp(M h_k). \tag{F.35}$$

Combining this bound with Eq. (F.32) yields

$$\|E_{k+1}\| \le \exp(M h_k)\|E_k\| + C_3(h_k^3 + h_k h_{\max}^4). \tag{F.36}$$

Since $E_0 = 0$, recursive application of this inequality gives

$$\|E_n\| \le C_4 \sum_{k=0}^{n-1} (h_k^3 + h_k h_{\max}^4), \tag{F.37}$$

where C_4 is independent of the integration grid on the fixed interval $[t_0, t_f]$. Using

$$\begin{aligned} \sum_{k=0}^{n-1} h_k^3 &\le (t_f - t_0) h_{\max}^2, \\ \sum_{k=0}^{n-1} h_k h_{\max}^4 &\le (t_f - t_0) h_{\max}^4, \end{aligned} \tag{F.38}$$

we obtain

$$\max_{0 \le n \le K} \|E_n\| \le C_5(h_{\max}^2 + h_{\max}^4) = \mathcal{O}(h_{\max}^2). \tag{F.39}$$

This establishes the second-order global convergence of the coupled latent and geometric integration scheme. □

Together with Propositions F.1 and F.2, this result characterizes three complementary properties of the motion-rate formulation: exact integration of equilibrium-induced constant-twist trajectories, preservation of the rigid-transformation group, and second-order convergence for smooth time-varying spatial twists. These properties establish the geometric and numerical consistency of the learned motion evolution over fixed forecasting intervals. Figure F.1 provides a complementary visualization of the numerical integration analysis.

Geometric coupling in offset-center rotation. Consider a rigid motion about a fixed spatial axis passing through a point $c \in \mathbb{R}^3$ with constant angular velocity $\omega \in \mathbb{R}^3 \setminus \{0\}$. The corresponding spatial twist is defined by

$$\xi = \begin{bmatrix} -[\omega]_\times c \\ \omega \end{bmatrix}. \tag{F.40}$$

Substituting $v = -[\omega]_\times c$ into Eq. (F.2) yields the coupled translational and rotational dynamics

$$\begin{aligned} \dot{p}(t) &= [\omega]_\times (p(t) - c), \\ \dot{R}(t) &= [\omega]_\times R(t). \end{aligned} \tag{F.41}$$

Let $\Delta t = t - t_0$ and define the relative rotation $Q(t) = \mathrm{Exp}(\Delta t [\omega]_\times)$. The exact solution is

$$\begin{aligned} R(t) &= Q(t) R(t_0), \\ p(t) &= c + Q(t)(p(t_0) - c). \end{aligned} \tag{F.42}$$

Equivalently, the rigid transformation evolves as

$$T(t) = \begin{bmatrix} Q(t) & (I_3 - Q(t))c \\ 0_{1\times 3} & 1 \end{bmatrix} T(t_0). \tag{F.43}$$

This expression follows directly from the spatial-twist exponential in Eq. (F.6), with the translational increment satisfying

$$J(\Delta t\, \omega)(\Delta t\, v) = (I_3 - Q(t))c. \tag{F.44}$$

To characterize the geometry of the resulting trajectory, let $u = \omega / \|\omega\|_2$ denote the unit rotation axis and define the orthogonal projection onto the plane perpendicular to this axis as

$$P_\perp = I_3 - u u^\top. \tag{F.45}$$

The perpendicular distance to the rotation axis is preserved throughout the motion:

$$\|P_\perp(p(t) - c)\|_2 = \|P_\perp(p(t_0) - c)\|_2 := r_\perp. \tag{F.46}$$

Differentiating the translational dynamics with respect to time gives

$$\begin{aligned} \ddot{p}(t) &= [\omega]_\times^2 (p(t) - c) \\ &= -\|\omega\|_2^2 P_\perp (p(t) - c). \end{aligned} \tag{F.47}$$

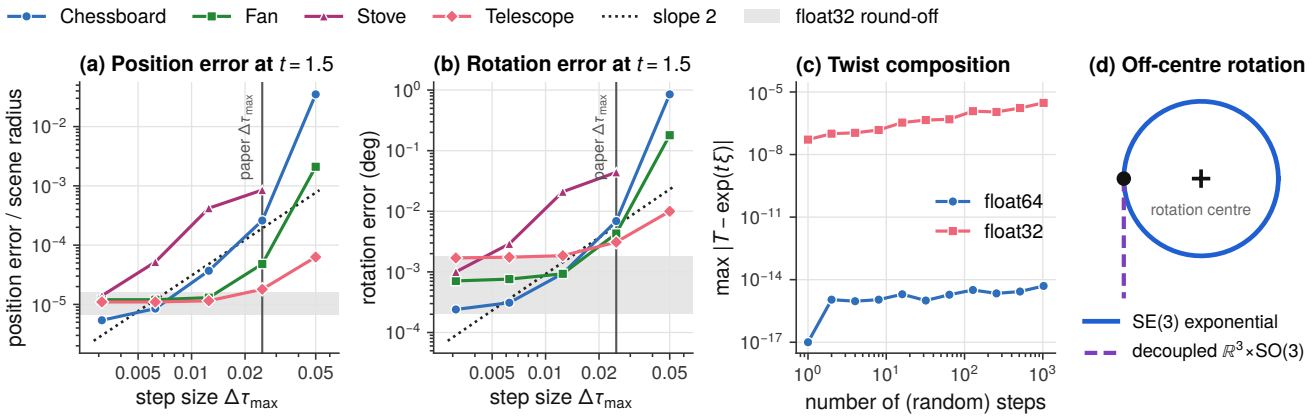

Figure F.1. **Numerical and geometric properties of motion-rate integration.** (a,b) Scene-normalized position and angular errors at $t = 1.5$ across integration step sizes. Dotted lines indicate second-order convergence references, vertical markers denote $\Delta\tau_{\max} = 0.025$, and shaded regions indicate float32 round-off. (c) Twist composition errors in float64 and float32 arithmetic. (d) Offset-center rotation under $\mathrm{SE}(3)$ and decoupled $\mathbb{R}^3 \times \mathrm{SO}(3)$ integration.

The corresponding centripetal acceleration has magnitude

$$\|\ddot{p}(t)\|_2 = \|\omega\|_2^2 r_\perp. \tag{F.48}$$

These identities establish the geometric relationship between angular velocity, translational curvature, and the spatial rotation axis. The $\mathrm{SE}(3)$ exponential represents their coupled evolution through a shared motion generator, yielding exact offset-center rotational trajectories under constant spatial twists.

Figure F.3 illustrates planar rigid-motion integration as a geometric example of coupled rotational and translational evolution.

F.5. Motion-Rate Error Propagation

The geometric integration of spatial twists establishes a direct relationship between instantaneous motion-rate deviations and accumulated transformation discrepancies. We characterize this relationship by exploiting the rotational and translational structure of $\mathrm{SE}(3)$.

Proposition F.4 (Finite-horizon geometric error propagation)**.** *Consider two continuous spatial-twist trajectories*

$$\begin{aligned} \xi(t) &= (v(t), \omega(t)), \\ \bar{\xi}(t) &= (\bar{v}(t), \bar{\omega}(t)), \end{aligned} \tag{F.49}$$

defined on a fixed interval $[t_0, t_f]$*. Their corresponding rigid transformations* $T(t) = (R(t), p(t)) \in \mathrm{SE}(3)$ *and* $\bar{T}(t) = (\bar{R}(t), \bar{p}(t)) \in \mathrm{SE}(3)$ *satisfy*

$$\dot{T}(t) = \widehat{\xi}(t) T(t), \qquad \dot{\bar{T}}(t) = \widehat{\bar{\xi}}(t) \bar{T}(t), \tag{F.50}$$

with the common initial condition $T(t_0) = \bar{T}(t_0) = T_0$.

Define the linear and angular rate deviations as

$$\begin{aligned} \Delta v(t) &= v(t) - \bar{v}(t), \\ \Delta\omega(t) &= \omega(t) - \bar{\omega}(t). \end{aligned} \tag{F.51}$$

For any fixed canonical point $y_0 \in \mathbb{R}^3$*, let*

$$\begin{aligned} y(t) &= R(t) y_0 + p(t), \\ \bar{y}(t) &= \bar{R}(t) y_0 + \bar{p}(t). \end{aligned} \tag{F.52}$$

Then the rotational deviation satisfies

$$\frac{\|R(t) - \bar{R}(t)\|_F}{\sqrt{2}} \le \int_{t_0}^{t} \|\Delta\omega(s)\|_2 \, \mathrm{d}s, \tag{F.53}$$

and the spatial trajectory deviation satisfies

$$\begin{aligned} \|y(t) - \bar{y}(t)\|_2 \le \int_{t_0}^{t} \big(&\|\Delta v(s)\|_2 \\ &+ \|\Delta\omega(s)\|_2 \|\bar{y}(s)\|_2 \big) \, \mathrm{d}s. \end{aligned} \tag{F.54}$$

Proof. The spatial-twist kinematics induce the rotational dynamics

$$\dot{R} = [\omega]_\times R, \qquad \dot{\bar{R}} = [\bar{\omega}]_\times \bar{R}. \tag{F.55}$$

Define the rotational difference as $\mathcal{E}_R(t) = R(t) - \bar{R}(t)$. Its evolution satisfies

$$\dot{E}_R = [\omega]_\times E_R + [\Delta\omega]_\times \bar{R}, \qquad E_R(t_0) = 0. \tag{F.56}$$

Let $U_\omega(t, s)$ denote the state-transition matrix generated by $[\omega(t)]_\times$,

$$\begin{aligned} \partial_t U_\omega(t, s) &= [\omega(t)]_\times U_\omega(t, s), \\ U_\omega(s, s) &= I_3. \end{aligned} \tag{F.57}$$

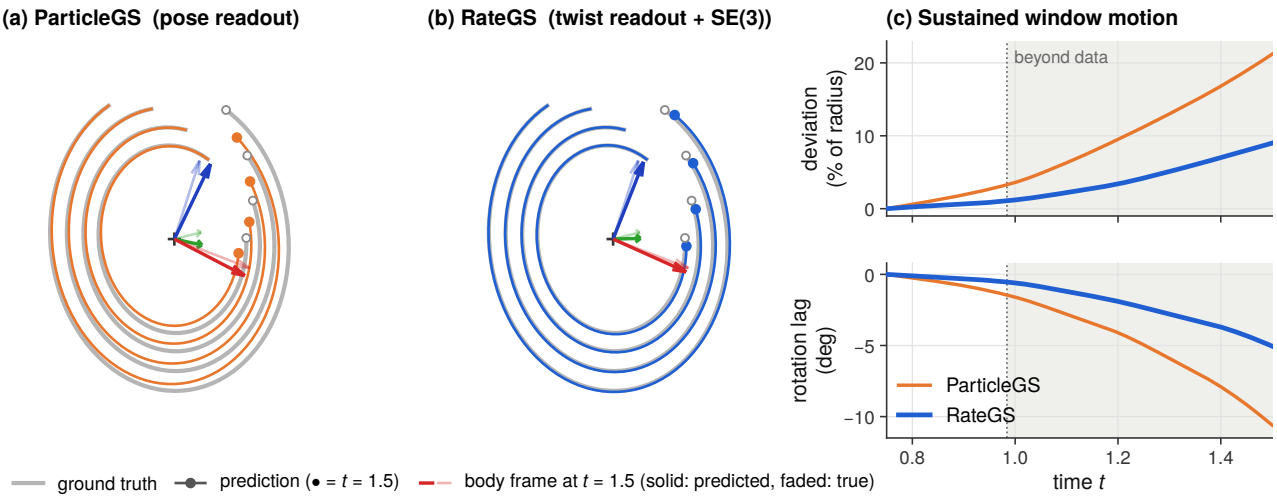

Figure F.2. **Sustained rigid-motion forecasting.** (a,b) Rotational trajectories predicted by ParticleGS and RateGS relative to the reference trajectories (grey). Local coordinate frames indicate the predicted and reference orientations at $t = 1.5$. (c) Temporal evolution of radial deviation and angular lag beyond the observed interval.

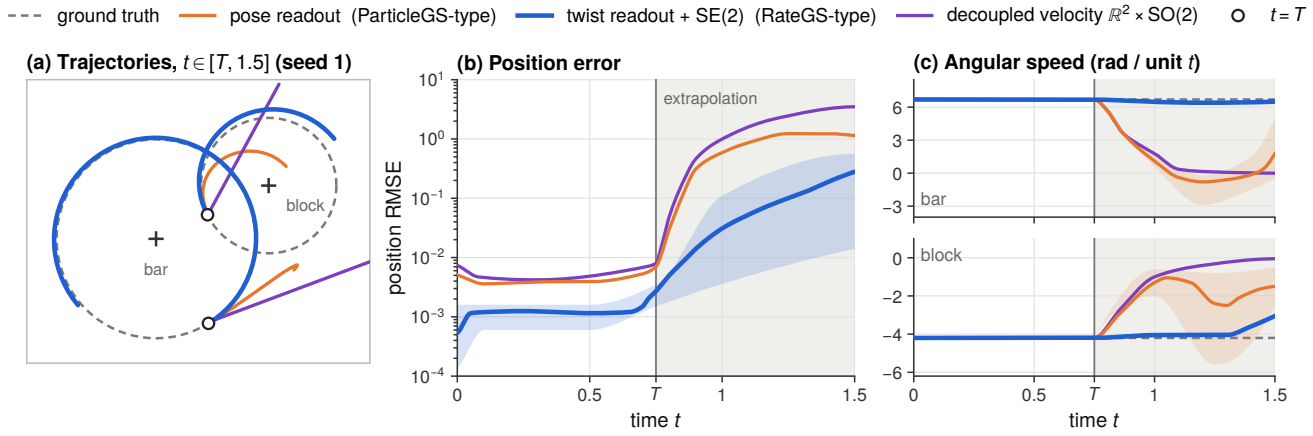

Figure F.3. **Controlled planar kinematics experiment.** (a) Forecast trajectories of the planar bar–block construction beyond the observation boundary, with dashed curves representing reference trajectories. (b) Temporal evolution of positional RMSE. (c) Signed angular-velocity trajectories. The planar construction provides a geometric illustration of rotation–translation coupling.

Since $[\omega(t)]_\times$ is skew-symmetric, $U_\omega(t,s) \in \mathrm{SO}(3)$. The variation-of-constants formula therefore gives

$$E_R(t) = \int_{t_0}^{t} U_\omega(t,s)[\Delta\omega(s)]_\times \bar{R}(s)\,\mathrm{d}s. \tag{F.58}$$

Using orthogonal invariance of the Frobenius norm and the identity

$$\|[a]_\times\|_F = \sqrt{2}\|a\|_2, \qquad a \in \mathbb{R}^3, \tag{F.59}$$

we obtain

$$\begin{aligned}\|E_R(t)\|_F &\le \int_{t_0}^{t} \|[\Delta\omega(s)]_\times\|_F\,\mathrm{d}s \\ &= \sqrt{2}\int_{t_0}^{t} \|\Delta\omega(s)\|_2\,\mathrm{d}s.\end{aligned} \tag{F.60}$$

This establishes Eq. (F.53).

For the transformed points, the spatial kinematics give

$$\begin{aligned}\dot{y} &= [\omega]_\times y + v, \\ \dot{\bar{y}} &= [\bar{\omega}]_\times \bar{y} + \bar{v}.\end{aligned} \tag{F.61}$$

Defining $e_y(t) = y(t) - \bar{y}(t)$ yields

$$\dot{e}_y = [\omega]_\times e_y + [\Delta\omega]_\times \bar{y} + \Delta v, \tag{F.62}$$

with $e_y(t_0) = 0$. Consequently,

$$e_y(t) = \int_{t_0}^{t} U_\omega(t,s)\big([\Delta\omega(s)]_\times \bar{y}(s) + \Delta v(s)\big)\,\mathrm{d}s. \tag{F.63}$$

Since $U_\omega(t,s)$ preserves Euclidean norms and

$$\|a \times b\|_2 \le \|a\|_2 \|b\|_2, \tag{F.64}$$

the spatial deviation satisfies

$$\|e_y(t)\|_2 \le \int_{t_0}^{t} \big(\|\Delta\omega(s)\|_2 \|\bar{y}(s)\|_2 + \|\Delta v(s)\|_2\big)\,\mathrm{d}s. \tag{F.65}$$

This establishes Eq. (F.54). □

Dependence on the integration horizon. Let $L = t - t_0$ denote the elapsed integration duration. Assume that the linear and angular rate deviations satisfy the uniform bounds

$$\sup_{s\in[t_0,t]} \|\Delta v(s)\|_2 \le \varepsilon_v, \quad \sup_{s\in[t_0,t]} \|\Delta\omega(s)\|_2 \le \varepsilon_\omega, \tag{F.66}$$

and that the reference linear twist component satisfies

$$\sup_{s\in[t_0,t]} \|\bar{v}(s)\|_2 \le V. \tag{F.67}$$

The reference trajectory admits the representation

$$\bar{y}(s) = U_{\bar{\omega}}(s,t_0)\bar{y}(t_0) + \int_{t_0}^{s} U_{\bar{\omega}}(s,r)\bar{v}(r)\,\mathrm{d}r. \tag{F.68}$$

Orthogonality of $U_{\bar{\omega}}$ implies

$$\|\bar{y}(s)\|_2 \le \|\bar{y}(t_0)\|_2 + V(s - t_0). \tag{F.69}$$

Substituting these bounds into Eqs. (F.53) and (F.54) gives

$$\frac{\|R(t) - \bar{R}(t)\|_F}{\sqrt{2}} \le L\varepsilon_\omega \tag{F.70}$$

and

$$\|y(t) - \bar{y}(t)\|_2 \le L\varepsilon_v + L\varepsilon_\omega \|\bar{y}(t_0)\|_2 + \frac{1}{2} V L^2 \varepsilon_\omega. \tag{F.71}$$

The resulting estimate separates the accumulated linear-rate deviation from the spatial contribution of angular-rate deviations. The latter depends on the reference trajectory's distance from the spatial coordinate origin.

For the forecasting horizon $H = t - t_{\mathrm{obs}}$ defined in Sec. E, the complete integration duration satisfies

$$L = (t_{\mathrm{obs}} - t_0) + H. \tag{F.72}$$

The bound thus accounts for motion-rate deviations accumulated over both the observed and forecasting intervals.

Intrinsic rotational deviation. The rotational discrepancy can equivalently be characterized by the geodesic angle $E_R(t) \in [0,\pi]$ defined in Eq. (E.3). For two rotations $R, \bar{R} \in \mathrm{SO}(3)$, the Frobenius distance satisfies

$$\frac{\|R - \bar{R}\|_F}{\sqrt{2}} = 2\sin\left(\frac{E_R}{2}\right). \tag{F.73}$$

A direct intrinsic bound follows by considering the relative rotation

$$C(t) = \bar{R}(t)^\top R(t). \tag{F.74}$$

Its dynamics satisfy

$$\dot{C}(t) = [\bar{R}(t)^\top \Delta\omega(t)]_\times C(t), \qquad C(t_0) = I_3. \tag{F.75}$$

The length of this rotation trajectory under the standard angular metric is

$$\mathcal{L}_C(t) = \int_{t_0}^{t} \|\bar{R}(s)^\top \Delta\omega(s)\|_2\,\mathrm{d}s = \int_{t_0}^{t} \|\Delta\omega(s)\|_2\,\mathrm{d}s. \tag{F.76}$$

Since the geodesic distance is bounded by the length of any connecting rotation trajectory,

$$E_R(t) \le \min\left\{\pi, \int_{t_0}^{t} \|\Delta\omega(s)\|_2\,\mathrm{d}s\right\}. \tag{F.77}$$

This establishes a direct relationship between accumulated angular-rate deviations and intrinsic rotational discrepancy.

Propagation to Gaussian geometry. For the rigid center $x_i^{\mathrm{rig}}(t) = Q_i(t)x_i + b_i(t)$ and its reference, Proposition F.4 applies with $y_0 = x_i$. For full deformed centers with respective local residuals $d_i, \bar{d}_i$, Eq. (A.3) yields

$$\|x_i^{\mathrm{def}} - \bar{x}_i^{\mathrm{def}}\|_2 \le \|x_i^{\mathrm{rig}} - \bar{x}_i^{\mathrm{rig}}\|_2 + \|d_i - \bar{d}_i\|_2 + \|Q_i - \bar{Q}_i\|_{\mathrm{op}} \|\bar{d}_i\|_2. \tag{F.78}$$

This follows by adding and subtracting $Q_i\bar{d}_i$ and using orthogonality. Thus a full-center bound also requires control of residual displacement and its rotation. No corresponding image-space error bound is inferred.

Table G.1. Computational profiling of per-scene training duration, peak GPU memory, and rendering throughput.

Method	Train (h) ↓	Peak (GB) ↓	FPS ↑
ParticleGS	1.8	12.0	45.0
RateGS	3.0	37.0	30.5

Exact accumulation under constant translation. Consider constant purely translational twists $\xi = (v, 0)$ and $\bar{\xi} = (\bar{v}, 0)$. Their spatial trajectories satisfy

$$\dot{y}(t) = v, \qquad \dot{\bar{y}}(t) = \bar{v}. \tag{F.79}$$

With a common initial position, direct integration gives

$$y(t) - \bar{y}(t) = L(v - \bar{v}), \tag{F.80}$$

and therefore

$$\|y(t) - \bar{y}(t)\|_2 = L\|v - \bar{v}\|_2. \tag{F.81}$$

In this setting, the spatial error bound is attained exactly, establishing the sharp linear dependence of accumulated translational deviation on the integration duration.

Together, these results characterize the propagation of motion-rate deviations through spatial-twist kinematics over finite temporal intervals. The rotational structure yields a direct integral bound on angular deviations, while the coupled translational dynamics establish the dependence of spatial discrepancies on both rate accuracy and trajectory geometry. These properties complement the numerical convergence of the geometric integrator and provide a mathematical foundation for the perturbation sensitivities introduced in Sec. D.

G. Computational Cost

Training and rendering cost. Tab. G.1 reports training duration, peak GPU memory, and rendering throughput. ParticleGS requires 1.8 h and 12.0 GB, achieving 45.0 FPS. RateGS requires 3.0 h and 37.0 GB while maintaining 30.5 FPS.

RateGS trades additional training time and memory for higher reported forecasting fidelity; Figs. G.1 and G.2 visualize these measurements.

H. Additional Visual Evidence and Mechanistic Interpretation

H.1. Rendered Forecasts and Image-Space Fidelity

Evaluation scope. We evaluate RateGS and RateGS-Base in terms of geometric fidelity, spatial alignment, and appearance consistency throughout the forecasting interval. Quantitative results for the complete RateGS framework are reported in Tab. B.1.

Qualitative forecasting comparison. Figure H.3 presents rendered forecasts across four scenes at three successive prediction timestamps. Compared with ParticleGS, TRACE, and FreeGave, RateGS-Base achieves higher PSNR across all displayed frames and better preserves object geometry and local appearance. These observations support the effectiveness of motion-rate learning and coupled geometric integration for temporal extrapolation.

Spatially resolved forecasting fidelity. Spatially resolved evaluation complements global accuracy metrics in learned dynamical prediction [1]. Let $I_t^v(u, c)$ and $\hat{I}_t^v(u, c)$ denote the reference and predicted RGB intensities, respectively, at pixel u, color channel $c \in \{1, 2, 3\}$, timestamp t, and camera view v. We define the channel-averaged absolute RGB error as

$$E_t^v(u) = \frac{1}{3} \sum_{c=1}^{3} \left| \hat{I}_t^v(u, c) - I_t^v(u, c) \right|. \tag{H.1}$$

Figure H.4 presents the spatial distribution of this quantity for ParticleGS and RateGS at successive forecasting timestamps. The resulting maps characterize local image-space agreement across object boundaries, structural regions, and appearance details, while PSNR quantifies global photometric fidelity through mean squared error. Together, these complementary measurements provide a spatially resolved and quantitative characterization of forecasting fidelity throughout the evaluated temporal interval.

Sustained rigid-window motion. Figure F.2 compares direct pose readout and spatial-twist readout with left-multiplicative $\mathrm{SE}(3)$ integration for an illustrated bounded rotational trajectory. Radial deviation and angular lag characterize the geometric evolution beyond the observation boundary. The displayed example exhibits closer reference alignment for the twist-based trajectory.

H.2. Forecasting Beyond the Benchmark Horizon

Extended-horizon evaluation. We compare ParticleGS and RateGS at normalized timestamps from approximately $t = 0.98$ to $t = 1.5$. Four of eight candidate scenes (Fan, Chessboard, Whale, and Darkroom) are selected by visual contrast; these examples are illustrative rather than a representative sample. The final recorded frame at approximately $t = 0.98$ provides visual context rather than time-matched ground truth; subsequent frames are assessed qualitatively for structural coherence and temporal continuity.

Geometric and temporal coherence. Figures H.5 to H.8 shows the evolution of rendered predictions. In the selected sequences, RateGS exhibits greater visual coherence in object geometry and appearance. Figure H.9 compares the

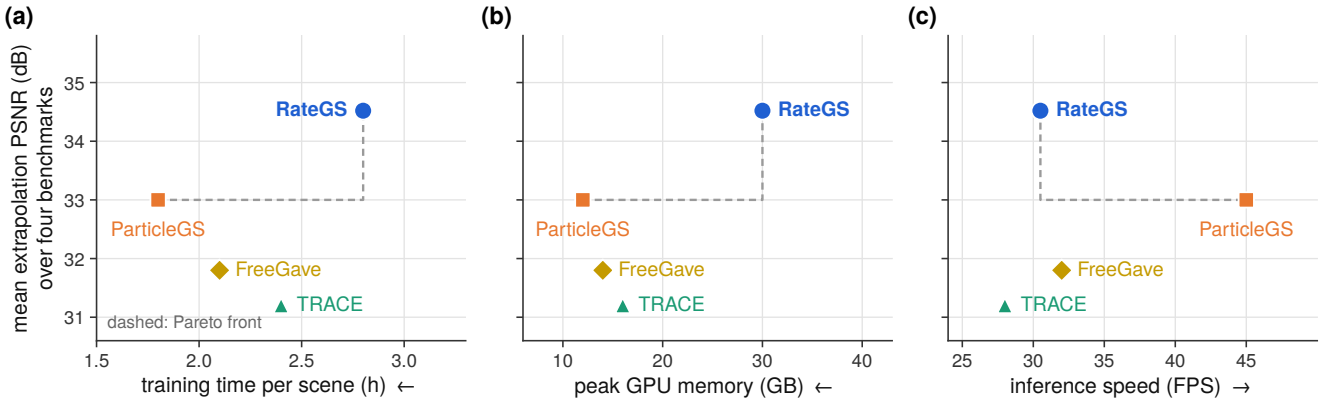

Figure G.1. **Forecasting accuracy and computational efficiency.** Forecasting PSNR in relation to training duration, peak GPU memory, and inference throughput.

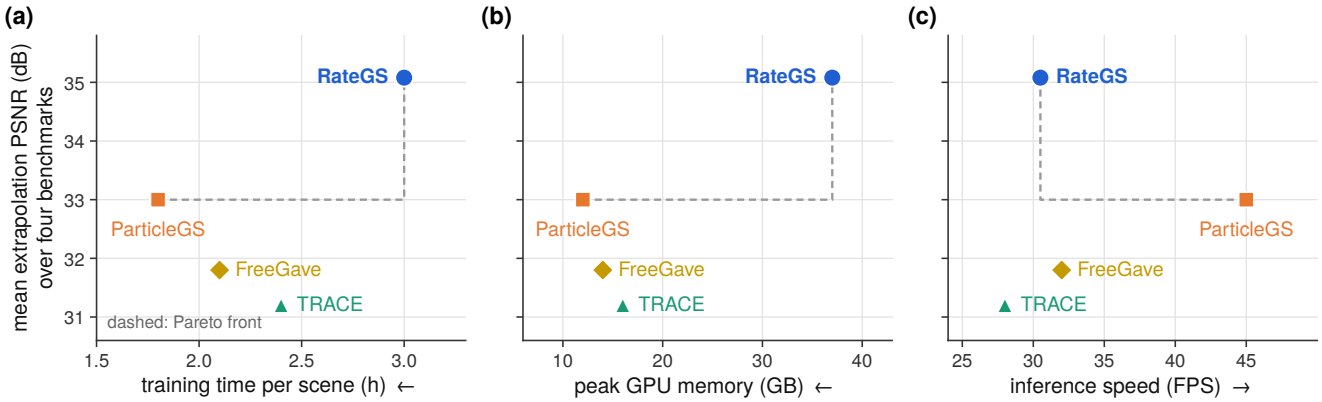

Figure G.2. **Comparative computational performance.** Complementary characterization of forecasting accuracy and computational efficiency.

methods at $t = 1.3$. These observations illustrate the behavior of motion-rate prediction and group-valued integration over extended horizons.

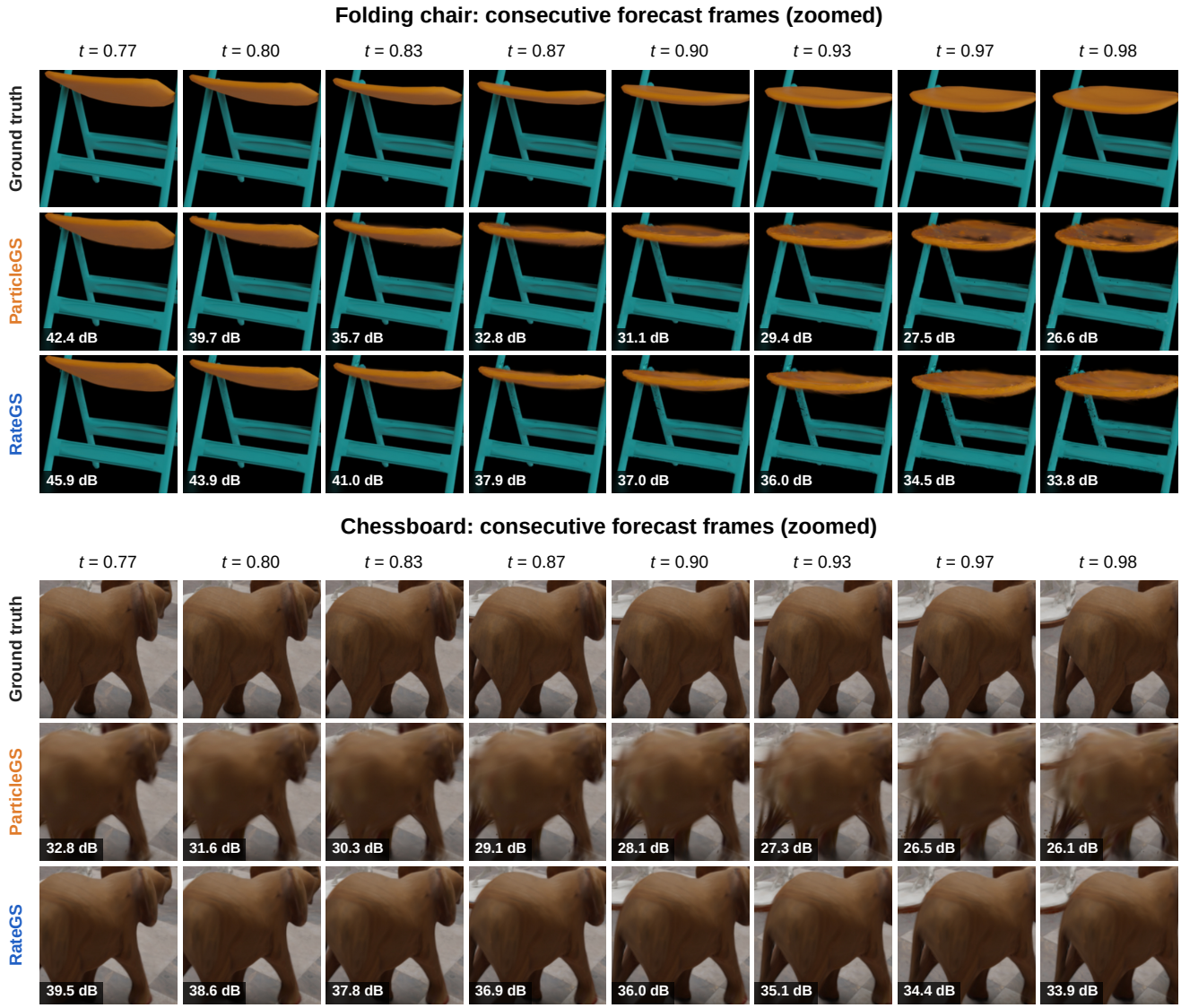

Figure H.1. **Temporal evolution of rendered forecasts.** Qualitative comparison of ParticleGS and RateGS across successive normalized forecasting timestamps. The rendered sequences illustrate the temporal evolution of object geometry, spatial configuration, and appearance consistency. Frame-specific PSNR values quantify photometric fidelity relative to the corresponding reference views.

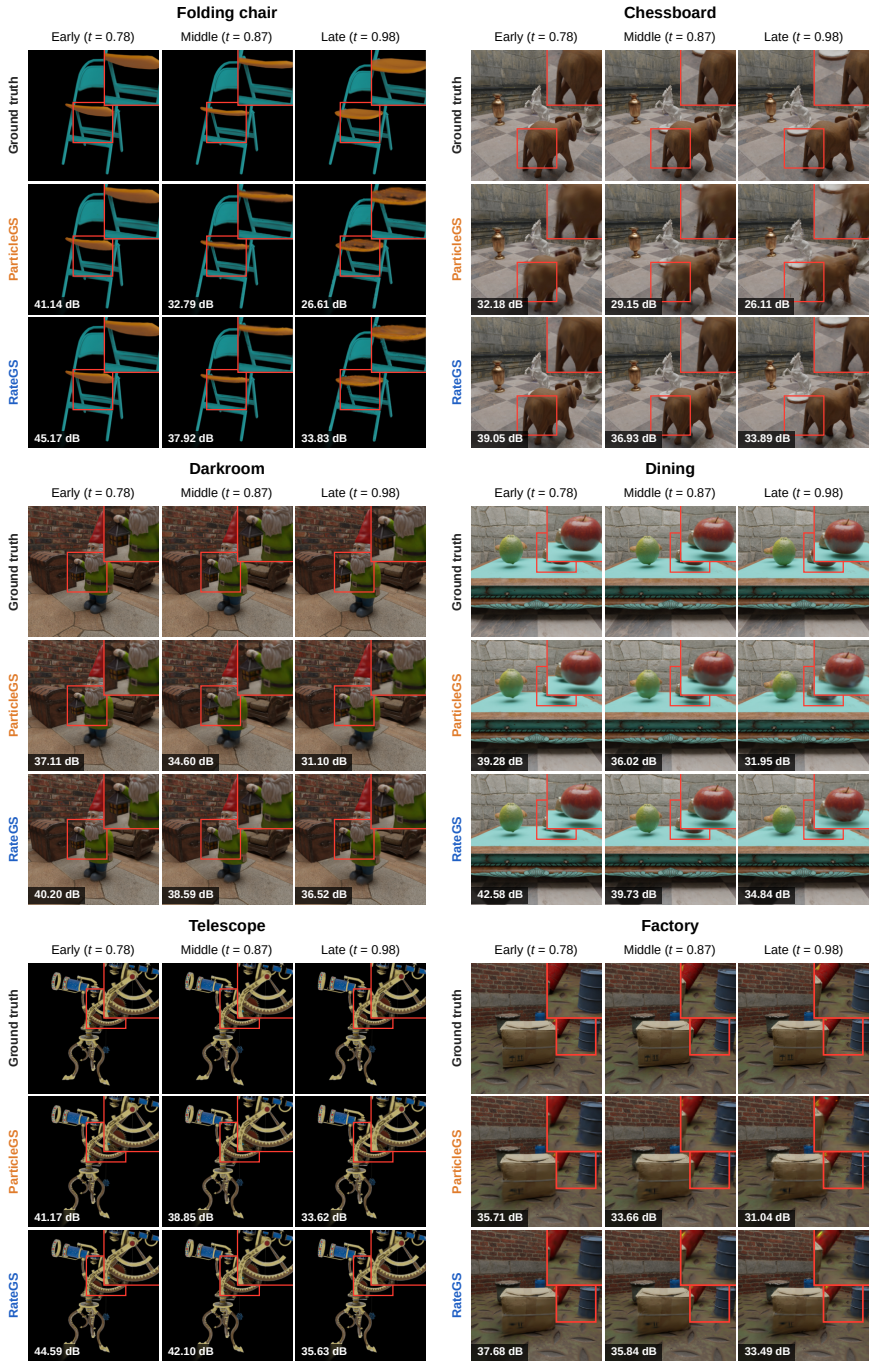

Figure H.2. **Qualitative comparison of future-view synthesis.** Rendered predictions of ParticleGS and RateGS across selected dynamic scenes and camera viewpoints. The visualizations illustrate the temporal evolution of object geometry, spatial alignment, and appearance consistency throughout the forecasting interval. The corresponding PSNR values quantify image-space forecasting fidelity at the evaluated timestamps.

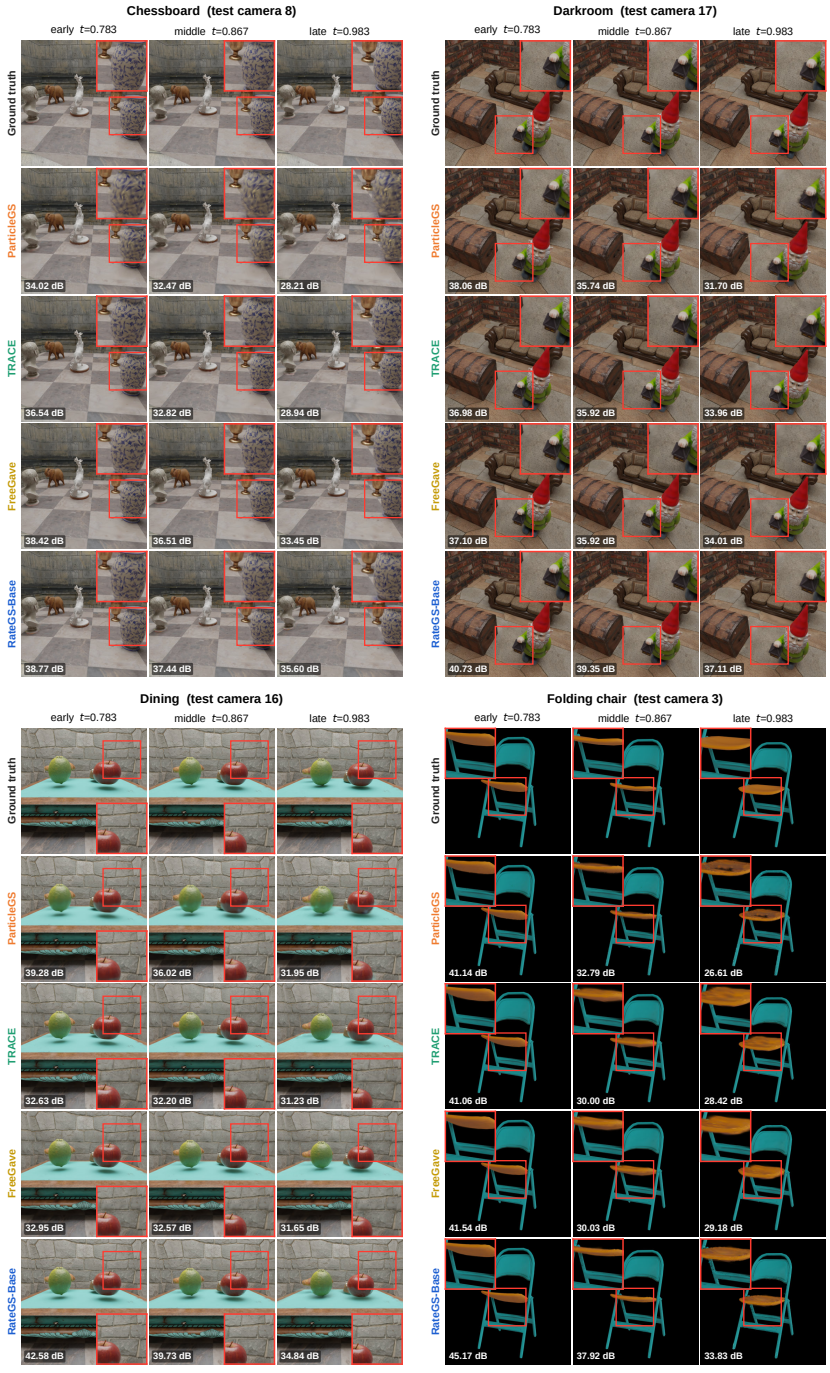

Figure H.3. **Qualitative comparison of motion forecasting.** Rendered predictions of ParticleGS, TRACE, FreeGave, and RateGS-Base on Chessboard, Darkroom, Dining, and Foldingchair at $t = 0.783$, 0.867, and 0.983. Red boxes highlight local geometric structures and appearance details. Frame-specific PSNR values (dB) quantify rendering fidelity relative to the corresponding reference images.

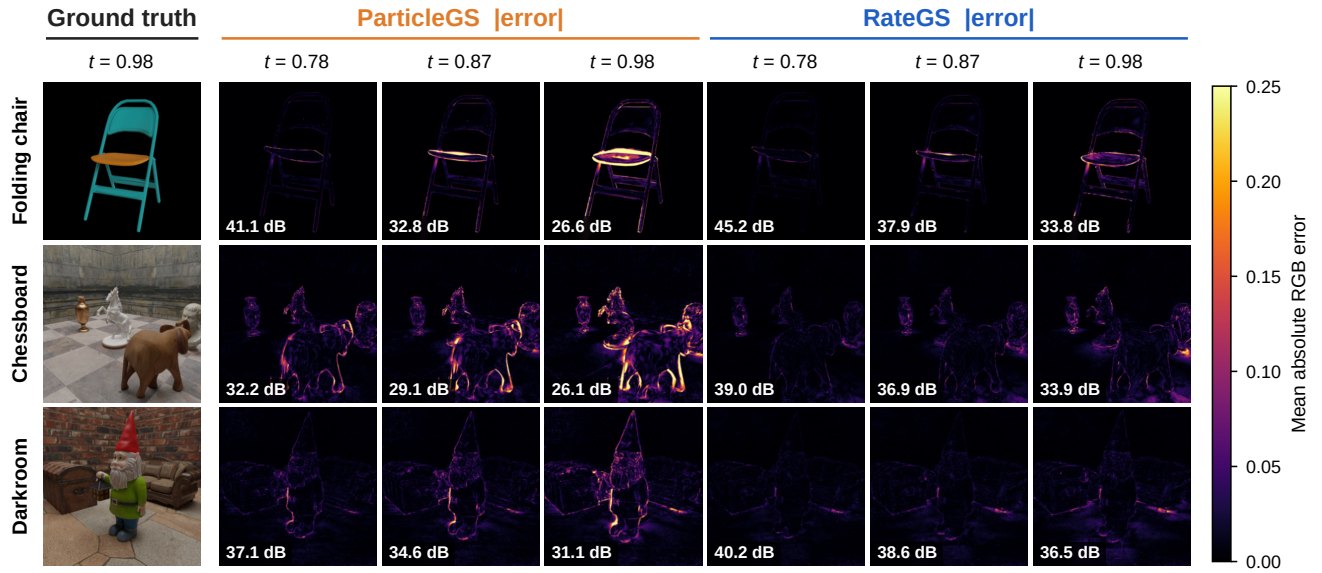

Figure H.4. **Spatially resolved forecasting fidelity.** Channel-averaged absolute RGB error maps for ParticleGS and RateGS at normalized timestamps $t = 0.78$, 0.87, and 0.98. Each map is computed using the corresponding reference image at the evaluated timestamp. The spatial distributions characterize local photometric agreement across object boundaries, geometric structures, and appearance regions. The displayed ground-truth column corresponds to $t = 0.98$.

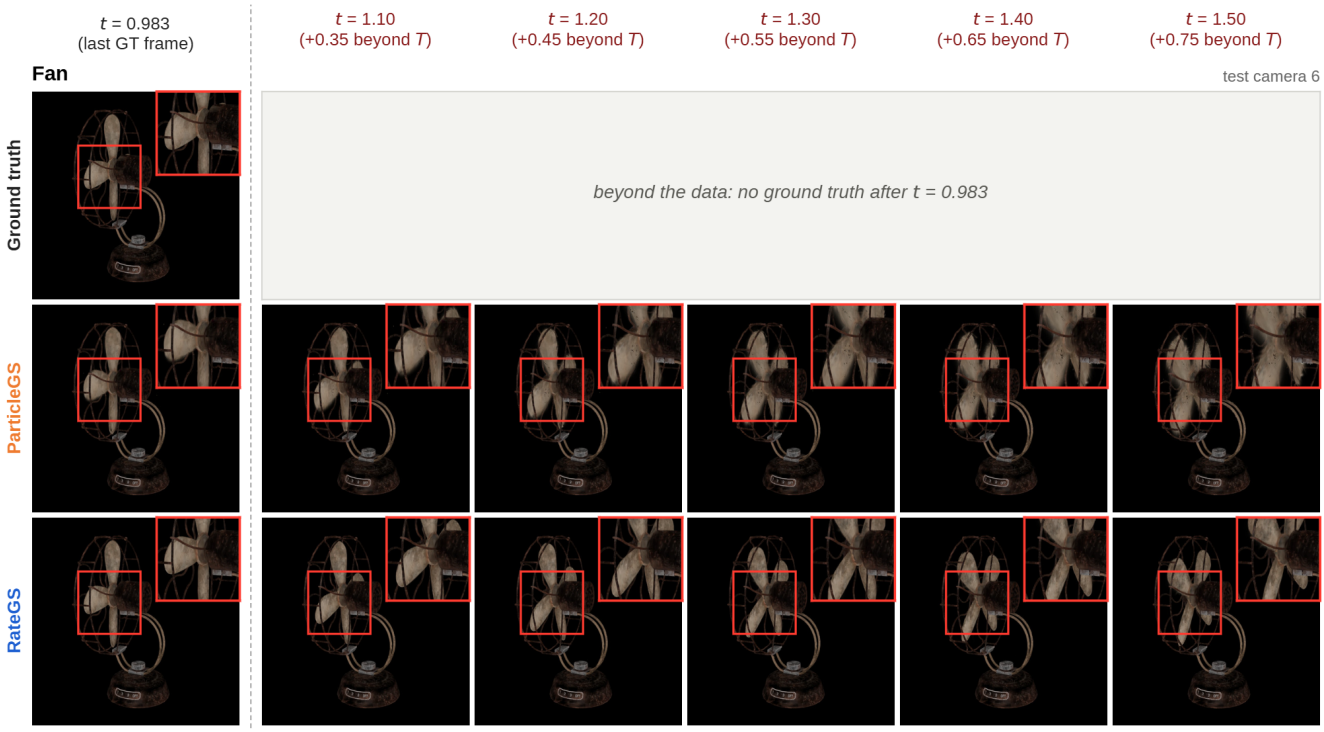

Figure H.5. **Extended-horizon rotational dynamics on Fan.** Qualitative comparison of ParticleGS and RateGS up to $t = 1.5$, illustrating the temporal evolution of blade geometry, rotational configuration, and structural coherence. The final recorded frame serves as the visual reference for extended temporal extrapolation.



Figure H.6. **Extended-horizon scene dynamics on Chessboard.** Qualitative comparison of ParticleGS and RateGS up to $t = 1.5$, illustrating the evolution of object geometry, relative spatial configuration, and rendered appearance. The final recorded frame provides the visual reference for assessing extended-horizon coherence.

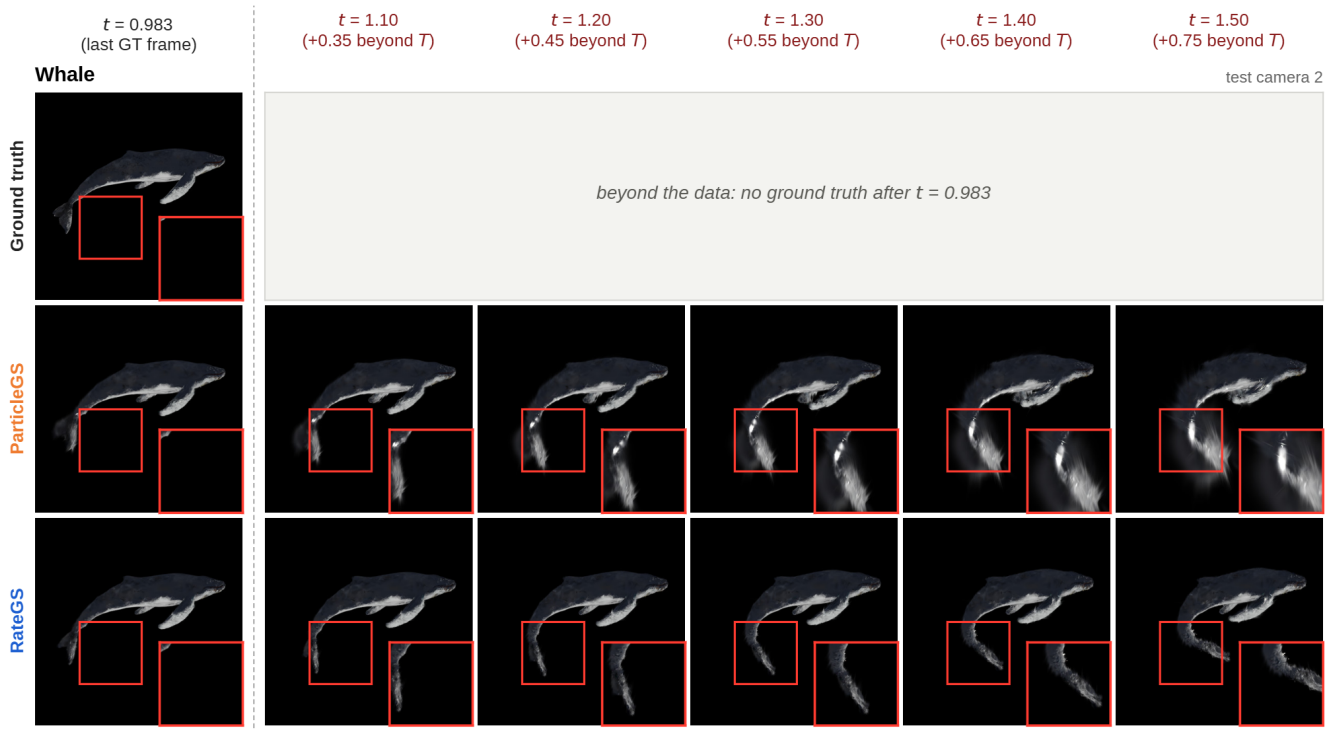

Figure H.7. **Extended-horizon motion dynamics on Whale.** Qualitative comparison of ParticleGS and RateGS up to $t = 1.5$, illustrating the temporal evolution of object morphology, boundary coherence, and spatial configuration. The final recorded frame provides the visual reference for assessing the continuity of object motion.



Figure H.8. **Extended-horizon scene dynamics on Darkroom.** Qualitative comparison of ParticleGS and RateGS up to $t = 1.5$, illustrating object-level geometric coherence, spatial relationships, and temporal appearance evolution. The final recorded frame serves as the visual reference for extended-horizon forecasting.

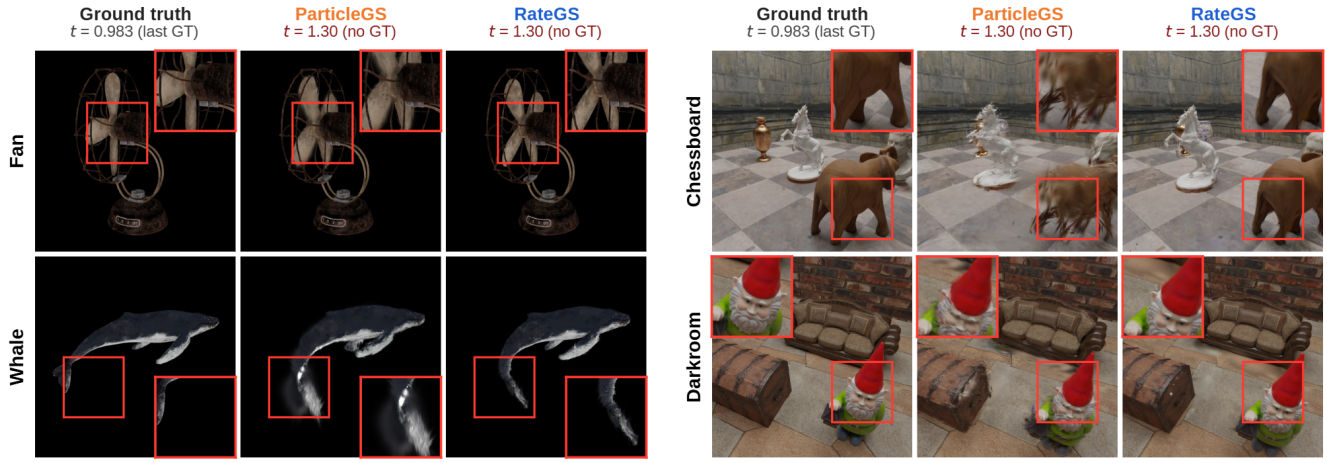

Figure H.9. **Cross-scene forecasting beyond the benchmark horizon.** Qualitative comparison of ParticleGS and RateGS at $t = 1.3$ across selected dynamic scenes. The rendered predictions illustrate geometric coherence, spatial organization, and appearance consistency across distinct motion patterns under extended temporal extrapolation.

References

[1] Katiana Kontolati, Somdatta Goswami, George Em Karniadakis, and Michael D. Shields. Learning nonlinear operators in latent spaces for real-time predictions of complex dynamics in physical systems. *Nature Communications*, 15:5101, 2024. 1, 15

[2] Jinsheng Quan, Qiaowei Miao, Yichao Xu, Zizhuo Lin, Ying Li, Wei Yang, Zhihui Li, and Yawei Luo. ParticleGS: Learning neural Gaussian particle dynamics from videos for prior-free physical motion extrapolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8331–8341, 2026. 1

[3] Francesco Regazzoni, Stefano Pagani, Matteo Salvador, Luca Dede', and Alfio Quarteroni. Learning the intrinsic dynamics of spatio-temporal processes through Latent Dynamics Networks. *Nature Communications*, 15:1834, 2024. 1, 2