PersistGS: Differentiable Physics for Object Permanence in 4D Gaussian Splatting

Adrian Ramlal[1] and John S. Zelek[1]

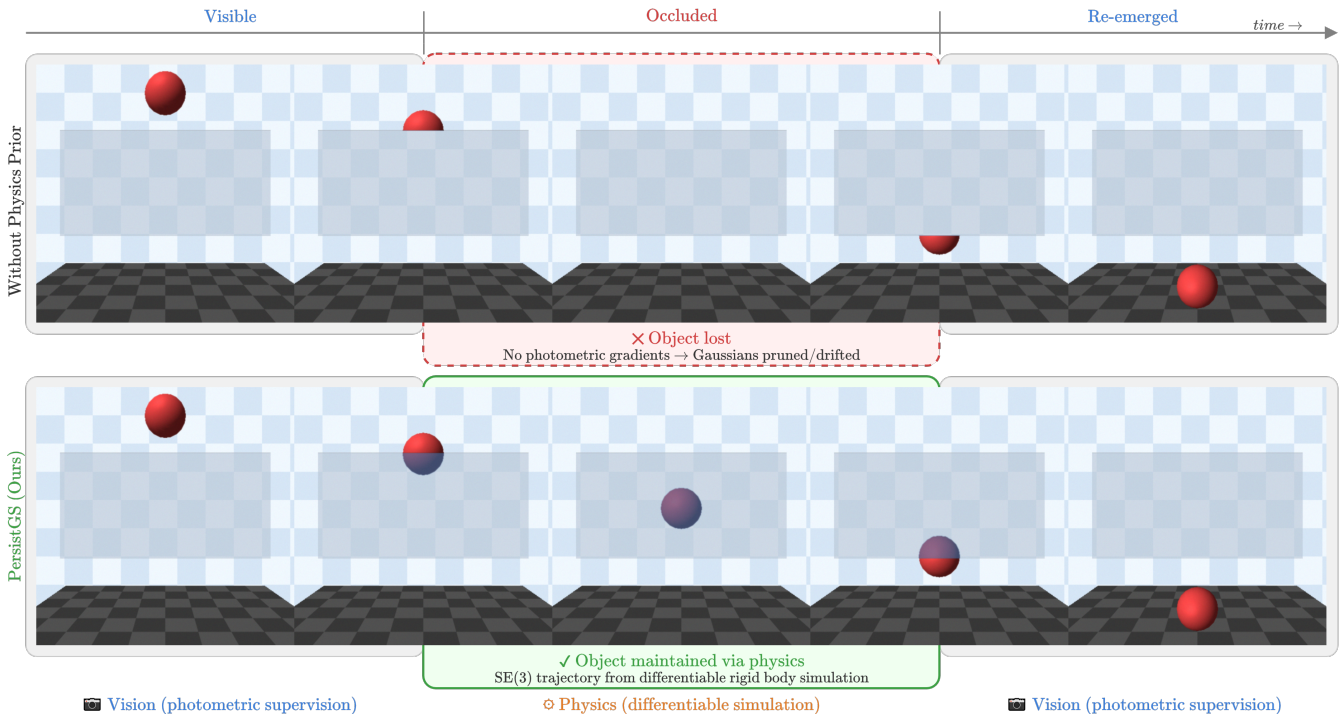

Fig. 1: Object permanence through physics. A ball falls past an occluder (opaque, but rendered translucent here for visualization). Without a physics prior (top), the object's Gaussians receive no photometric gradients during occlusion and are lost. PersistGS (bottom) maintains the object throughout by positioning its Gaussians along an SE(3) trajectory predicted from pre-occlusion observations via differentiable rigid body simulation.

***Abstract*— Dynamic 3D Gaussian Splatting (3DGS) methods reconstruct time-varying scenes from synchronized multi-camera video using photometric supervision. When a moving object becomes fully occluded from all training cameras, this supervision vanishes: the Gaussians representing it receive no gradient signal and degrade. Existing approaches to incomplete observations in neural reconstruction rely on learned generative priors that prioritize visual plausibility over physical correctness.**

We propose PersistGS, a method that restores object permanence during occlusion by coupling differentiable rigid body simulation with 3D Gaussian Splatting. Our approach decomposes the scene into per-object Gaussians and collision meshes, estimates friction and velocity from the observed pre-occlusion trajectory via differentiable simulation, and uses the resulting SE(3) trajectory to position object Gaussians throughout the occlusion period. Because the predicted trajectory satisfies the governing equations of rigid body dynamics, it faithfully captures contact events (bounces, friction-based deceleration, direction changes) that kinematic extrapolation cannot model. We introduce a centroid silhouette loss that isolates positional gradients from appearance noise, yielding 40% lower trajectory error than photometric supervision. We evaluate using cameras withheld from training that observe the object during its occlusion. Experiments on synthetic scenes show that PersistGS outperforms constant velocity extrapolation by $+2.46$ dB PSNR and comes within 0.19 dB of a ground-truth trajectory upper bound.

[1]The authors are with Department of Systems Design Engineering, University of Waterloo, Canada {adrian.ramlal, jzelek}@uwaterloo.ca

I. Introduction

3D Gaussian Splatting (3DGS) [1] has become a leading representation for photorealistic scene reconstruction from multi-view images. Several extensions reconstruct dynamic scenes from synchronized multi-camera video by learning per-Gaussian deformation fields [2], [3], sparse control-point trajectories [4], or per-object SE(3) poses [5], [6]. These methods produce high-fidelity reconstructions when all scene elements are continuously observed across training views.

A fundamental challenge arises when observations are temporally incomplete. When a moving object passes behind a static occluder and becomes invisible from all training cameras, the Gaussians representing it receive no photometric gradients. Without gradients, these Gaussians are pruned,

drift, or collapse, and the reconstruction must recreate the object from scratch upon re-emergence. The representation fails to maintain *object permanence* [7]: the principle that objects continue to exist when unobserved.

Recent work addresses incomplete observations through generative priors. Diffusion models hallucinate plausible content for unobserved regions [8], and generative object priors complete multi-object 4D reconstructions through occlusion [9], [10]. While these approaches produce visually plausible results, they do not guarantee physical correctness. For a dynamic object moving behind an occluder, the question is not what the object might look like from an unobserved angle, but *where the object is* during the unobserved interval, a problem that requires dynamical reasoning rather than appearance hallucination.

Physics provides a natural prior for this problem. Given the object's trajectory before occlusion, a rigid body simulator can predict its trajectory through the occlusion period, including contact interactions with the environment. Unlike kinematic extrapolation, physics correctly handles bounces, friction-based deceleration, and direction changes at contact. Unlike generative priors, the resulting trajectory is physically correct by construction: it satisfies Newton's laws and the contact constraints of the scene geometry.

We propose PersistGS, a method that integrates differentiable rigid body simulation with 3DGS to achieve faithful 4D reconstruction through temporal occlusion. Our approach rests on a natural decomposition: vision provides *appearance* from observed frames, while physics provides *position* during unobserved frames. We decompose the scene into per-object Gaussians and collision meshes, estimate physical parameters through a differentiable simulation and rendering loop, and apply the resulting SE(3) trajectory to position the object's Gaussians throughout the occlusion period. Upon re-emergence, photometric supervision seamlessly resumes.

Our contributions are:

1) A method that uses differentiable rigid body simulation as a physics prior for 4D Gaussian Splatting, estimating physical parameters from observed frames and applying the resulting SE(3) trajectory to maintain object Gaussians through temporal occlusion.
2) A centroid silhouette loss that decouples positional supervision from appearance quality, yielding 40% lower trajectory error than photometric supervision, combined with an observability-aware curriculum for joint friction and velocity estimation.
3) Experiments demonstrating that physics-based reconstruction through occlusion outperforms kinematic baselines by $+2.46$ dB PSNR and approaches ground-truth quality (0.19 dB gap), with ablations characterizing the interplay between estimation accuracy, occlusion duration, and reconstruction fidelity.

II. Related Work

A. Dynamic Gaussian Splatting

3DGS [1] represents scenes as anisotropic Gaussians optimized through differentiable rasterization. Extensions to dynamic scenes learn per-Gaussian deformations via spatiotemporal features [2], continuous deformation fields [3], or sparse control points with SE(3) transforms [4]. Dynamic 3D Gaussians [6] tracks per-Gaussian positions with rigidity constraints, and Shape of Motion [11] decomposes monocular video into SE(3) motion bases, the formalism our method adopts. For rigid objects, Street Gaussians [5] applies per-timestep SE(3) poses to per-object Gaussian sets, and Gaussian Grouping [12] enables object-level decomposition.

Several methods explicitly separate static and dynamic scene components, a design central to our pipeline. Ex4DGS [13] separates static and dynamic Gaussians with keyframe-based motion interpolation, SP-GS [14] clusters Gaussians into superpoints as group-level motion primitives, and DrivingGaussian [15] models each moving object individually via a composite dynamic Gaussian graph. These decomposed architectures share our motivation of avoiding deformation field learning from limited views, but none addresses what happens when an object leaves the observable field entirely.

B. Reconstruction Under Incomplete Observation

When observations are spatially sparse, 3DGS suffers from elongation artifacts and overfitting. Mip-Splatting [16], Drop-Gaussian [17], and DNGaussian [18] address this through anti-aliasing, dropout regularization, and monocular depth priors respectively. We compose these techniques to handle our clustered camera distribution.

For temporally incomplete dynamic scenes, recent methods employ learned priors. 4DGS in the Wild [19] applies diffusion and depth priors to uncertain regions in monocular 4DGS. GenMOJO [9] uses object-centric diffusion priors with occlusion-aware splatting, and DreamScene4D [10] lifts monocular video to 4D via amodal completion. A parallel line of work addresses the spatial variant: Amodal3R [20] reconstructs complete 3D objects from partially visible observations, and pix2gestalt [21] uses diffusion models for amodal segmentation. These generative approaches reason about *what* the object looks like from unobserved angles; our work reasons about *where* the object is during unobserved time intervals.

For transient occlusion in static scenes, RobustNeRF [22] and SpotLessSplats [8] suppress distractors that appear inconsistently across views. Our problem is the inverse: maintaining a *persistent* object that becomes *temporarily unobserved*.

C. Physics-Informed Neural Reconstruction

Differentiable simulation has been coupled with neural rendering for system identification and dynamics generation. GradSim [23] introduced end-to-end parameter estimation through differentiable physics and rendering. PAC-NeRF [24] estimates material properties via differentiable MPM simulation, and DANO [25] estimates rigid body properties from NeRF density fields with differentiable contact modeling. PhyRecon [26] demonstrates that physics priors improve neural surface reconstruction even for static scenes.

Within the Gaussian framework, most existing work couples continuum simulation with 3DGS for deformable dynamics. PhysGaussian [27] established the SE(3) Gaussian transform recipe we adopt. Subsequent methods extend this paradigm: OmniPhysGS [28] introduces per-Gaussian learnable constitutive models, GIC [29] and Spring-Gaus [30] integrate spring-mass and continuum formulations into Gaussian kernels, NeuMA [31] combines a physics prior with learned residual corrections for material identification, Feature Splatting [32] uses language queries to assign material properties for simulation, and PhysTwin [33] reconstructs and simulates deformable objects from real-world video. These methods all target *deformable* materials via continuum mechanics.

Fewer methods address rigid body dynamics. SDF-Sim [34] learns rigid body simulation over implicit shapes from visual observations. Vid2Sim [35] reconstructs simulation-ready Gaussians with physical properties from video. Embodied Gaussians [36] and Splatting Physical Scenes [37] build physics-rendering world models from robot data, and POGS [38] maintains persistent object Gaussians for tracking via feature matching. All of these methods use physics for forward dynamics, material estimation, or state tracking. We use physics for a distinct purpose: as a *reconstruction* prior that fills temporal observation gaps with dynamically consistent trajectories.

D. Object Permanence in Vision

Object permanence, the understanding that objects continue to exist when unobserved, has been studied primarily in tracking. Tokmakov et al. [7] introduced it as an explicit inductive prior for multi-object tracking, using recurrent networks to predict trajectories of fully occluded objects. Vysics [39] fuses vision with contact-rich physics for shape reconstruction under occlusion, though it uses physics to resolve geometric ambiguity rather than to predict temporal trajectories.

Our work extends object permanence from tracking to reconstruction: rather than maintaining a 2D bounding box through occlusion, we maintain a complete 3D Gaussian representation positioned by a physics-predicted SE(3) trajectory. To our knowledge, PersistGS is the first method to combine differentiable rigid body simulation with Gaussian Splatting for object permanence through temporal occlusion.

III. Method

Given synchronized multi-camera video of a dynamic scene, we consider the setting where a rigid object becomes fully occluded from all training cameras and later re-emerges. Our goal is a 4D reconstruction in which the object persists through occlusion with correct geometry and preserved appearance. Our pipeline has three stages: scene decomposition (§III-A), physics parameter estimation (§III-B), and physics-guided reconstruction (§III-C). Fig. 2 provides an overview.

A. Scene Decomposition

We use a decomposed representation: static background Gaussians ($\sim$25K) and a separate per-object Gaussian model ($\sim$511K from MV-SAM3D), composed in a single rasterization pass with automatic depth ordering. We evaluated this against a joint 4D Gaussian Splatting approach [2] and found the decomposed architecture outperforms by $+6.9$ dB, because 4DGS severely overfits with only 5 training cameras. The decomposed design avoids learning a deformation field from sparse views by keeping the background static and applying rigid SE(3) transforms to the object.

We reconstruct the scene at a reference frame (5 frames before the first occlusion) where all objects are visible.

Object Gaussians. The dynamic object is reconstructed using MV-SAM3D, which produces both a Gaussian representation and a collision mesh from posed multi-view images. The mesh provides collision geometry for the simulator.

Background Gaussians. The static environment is reconstructed using standard 3DGS with an inverse-mask weighted loss that excludes the object region.

Sparse-view regularization. Training cameras are clustered to produce consistent occlusion. We apply the Mip-Splatting 3D smoothing filter [16], DropGaussian dropout [17] (p=0.5), and depth supervision from Depth Anything V2 [40] (λ_{depth}=0.05).

B. Physics Parameter Estimation

We estimate friction and initial velocity from the visible trajectory using a differentiable simulation and rendering loop.

Parameterization. We optimize friction μ (in $\log_{10}$ space) and initial velocity $\mathbf{v}_0 = (v_x, v_y, v_z)$: four free parameters. The remaining contact parameters are fixed: mass m=1.0, contact stiffness k_e=10^5, friction stiffness k_f=10^3, and damping k_d=10^2. Visual trajectory observations constrain the net forces on the object but not the individual contact parameters that produce those forces. Specifically, k_e and mass enter the contact force equation only as a ratio ($a = k_e\delta/m$), making them strictly degenerate. The remaining stiffness parameters (k_f, k_d) interact with μ in governing friction buildup rate and energy dissipation during contact, making their joint estimation from trajectory data alone poorly conditioned. We therefore retain only the two parameters with the most direct and independent influence on the observable trajectory: μ (which sets the Coulomb friction limit) and $\mathbf{v}_0$ (which determines starting momentum).

Simulator. NVIDIA Newton [41], [42] simulates the object as a rigid body interacting with the scene's contact surfaces via a semi-implicit solver with penalty-based contact at 8 substeps per frame. The simulator outputs per-frame SE(3) poses (translation and quaternion). Warp's tape-based reverse-mode automatic differentiation provides gradients through the full trajectory, including contact events.

Enabling effective gradient flow through contact dynamics required two modifications to Newton's contact kernels. First, the default friction model uses a hard $\min(\cdot)$ clamp that creates zero-gradient regions when the friction force saturates at the Coulomb limit. We replace this with a smooth harmonic combination $f = (k_f v_t \cdot f_c)/(k_f v_t + f_c + \epsilon)$, where f_c is the Coulomb limit, ensuring non-zero gradients

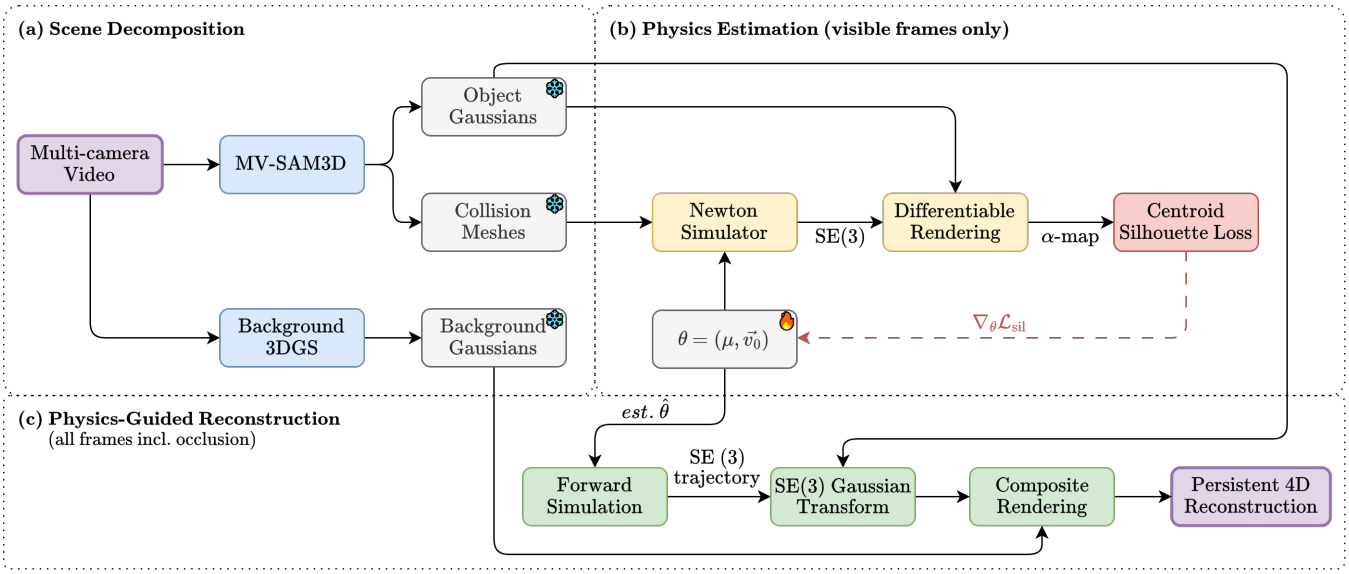

Fig. 2: PersistGS pipeline. **(a) Scene Decomposition** extracts per-object Gaussians and collision meshes via MV-SAM3D, and trains background Gaussians separately. All representations are frozen after this stage. **(b) Physics Estimation** simulates candidate parameters $\theta = (\mu, \mathbf{v}_0)$ through Newton, renders the alpha channel of the positioned object Gaussians, and minimizes a centroid silhouette loss. Only θ is optimized. **(c) Dynamic Reconstruction** applies the estimated SE(3) trajectory to the frozen object Gaussians and composites with the frozen background for the final 4D reconstruction.

in all regimes. Second, Newton's default material averaging ($k_e = 0.5(k_{e,\text{body}} + k_{e,\text{shape}})$) routes only 50% of the gradient to the optimizable shape material parameters. We bypass this averaging, yielding approximately $50\times$ improvement in gradient magnitude for contact parameters.

Centroid silhouette loss. Pixel-wise photometric losses combine two sources of error: the object's position (governed by physics) and its appearance (fixed from the pre-occlusion reconstruction). Because the object Gaussians are frozen, their spherical harmonic coefficients may not produce correct color from novel viewpoints, injecting appearance noise into physics gradients.

We instead supervise with a centroid silhouette loss that isolates position from appearance. The object Gaussians are transformed to the simulated SE(3) pose and rendered to produce an alpha map. The rendered centroid is compared against the ground-truth mask centroid:

$$\mathcal{L}_{\text{sil}} = \|\mathbf{c}_{\text{render}} - \mathbf{c}_{\text{gt}}\|_2^2, \quad \mathbf{c} = \frac{\sum_i \alpha_i \mathbf{p}_i}{\sum_i \alpha_i} \tag{1}$$

This loss provides a global basin of attraction: even when predicted and ground-truth silhouettes do not overlap, the centroid displacement yields a non-zero gradient toward the correct position. It also produces friction gradients approximately $100\times$ larger than photometric loss (Sec. IV-D), because centroid displacement is a first-order function of position, while pixel-wise RGB differences arise only at silhouette boundaries. Because centroid displacement is dominated by translation for any object whose radius is small relative to the scene, the loss provides effective positional gradients for physics parameters regardless of object geometry.

Observability-aware curriculum. Pre-contact frames constrain velocity but carry no friction information ($\partial\mathcal{L}/\partial\mu{=}0$ in free flight), while post-contact frames constrain both. We exploit this structure:

1) *Phase 1* (60 iter.): velocity from pre-contact frames, warm-started from finite differences, friction frozen.
2) *Phase 2* (60 iter.): friction from a post-contact frame window, velocity frozen.
3) *Phase 3* (80 iter.): joint refinement at $0.3\times$ learning rate.

We run 5 random initializations per scene and select the lowest-loss result.

C. Physics-Guided Reconstruction

The estimated parameters drive a full forward simulation, producing an SE(3) trajectory through the occlusion period. Background Gaussians remain static while object Gaussians are transformed per-frame following the SE(3) recipe for Gaussian dynamics [27]: positions $\boldsymbol{\mu}'{=}\mathbf{R}_t\boldsymbol{\mu}_{\text{can}}{+}\mathbf{t}_t$, quaternions $\mathbf{q}'{=}\mathbf{q}_{R_t}{\otimes}\mathbf{q}_{\text{can}}$, and scales unchanged under rigid motion. For spherical harmonics, we apply the inverse object rotation to the viewing direction before evaluation (color $=$ $\text{SH}(\mathbf{R}_t^{-1}\mathbf{d}_{\text{view}};\, \mathbf{C}_{\text{can}})$), which is algebraically equivalent to rotating all SH coefficients but avoids computing Wigner D-matrices.

During visible frames, per-frame residual translations are optimized over 3 passes through the visible frame set (learning rate 10^{-4}), minimizing photometric error against training views to correct small positional errors from imperfect parameter estimation. During occluded frames, the object follows the physics trajectory purely, with residuals disabled and no gradients propagating to the object representation. At

re-emergence, residual optimization and photometric supervision resume, ensuring a smooth transition back to vision-guided reconstruction. All Gaussians are composited in a single rasterization pass.

IV. Experiments

A. Setup

Scenes. We evaluate on three synthetic scenes simulated with NVIDIA Newton [42]. Each features a rigid ball (radius 1.25) in a static environment with a ground plane and occluding wall, designed so that contact events occur during occlusion:

- **ball_fall:** Free fall with lateral drift; ground bounce behind occluder. Occlusion: 248 frames (69% of 360). GT: μ=0.3, $\mathbf{v}_0$=$(3, 0, 0)$ m/s.
- **ball_bounce:** Parabolic arc; second bounce hidden behind occluder. Occlusion: 80 frames (33% of 240). GT: μ=0.15, $\mathbf{v}_0$=$(5, 0, 7)$ m/s.
- **ball_roll:** Ground roll with friction deceleration behind occluder. Occlusion: 45 frames (13% of 360). GT: μ=0.4, $\mathbf{v}_0$=$(10, 0, 0)$ m/s.

Each scene uses 5 training cameras and 2 evaluation cameras (overhead and side, withheld from training) at 512×512, 60 fps.

Baselines. (1) **GT trajectory** (upper bound); (2) **Ours**; (3) **Linear interpolation** between last-visible and first-reappearance positions (non-causal: requires the re-emergence location); (4) **Constant velocity** with gravity but no contact model (the only causal kinematic baseline); (5) **No physics** (object absent during occlusion).

Metrics. PSNR, LPIPS, and SSIM on an object-region crop from evaluation cameras during occlusion, plus 3D trajectory RMSE.

Implementation. All experiments use a single NVIDIA RTX 5080 GPU (16 GB). Physics estimation takes $\sim$5.5 min/scene (200 iterations across 3 curriculum phases, 5 random seeds). Background Gaussians are trained for 5K iterations ($\sim$25K Gaussians). Object Gaussians are initialized from MV-SAM3D and fine-tuned for 7K iterations ($\sim$511K Gaussians, DropGaussian rate 0.5, degree-3 spherical harmonics).

B. Parameter Estimation

Table I reports estimated physical parameters. Friction is recovered within 10% and velocity within 0.5 m/s on all components. Trajectory RMSE during occlusion ranges from 0.288 (ball_roll) to 1.147 (ball_fall).

C. Reconstruction Quality During Occlusion

Table II presents the primary evaluation. PersistGS achieves 17.15 dB mean PSNR on evaluation cameras during occlusion, outperforming constant velocity by $+2.46$ dB and linear interpolation by $+1.41$ dB, while approaching the ground-truth upper bound (17.34 dB, gap of 0.19 dB). LPIPS confirms consistent improvements: 0.314 vs. 0.381 (linear) and 0.491 (constant velocity).

TABLE I: Estimated vs. ground-truth physical parameters.

Scene	Param	GT	Est.	Error
ball_fall	μ	0.300	0.290	3.3%
	v_x	3.000	2.999	0.001
	v_y	0.000	-0.007	0.007
	v_z	0.000	-0.045	0.045
ball_bounce	μ	0.150	0.151	0.7%
	v_x	5.000	4.993	0.007
	v_y	0.000	-0.094	0.094
	v_z	7.000	6.994	0.006
ball_roll	μ	0.400	0.363	9.3%
	v_x	10.000	9.527	0.473
	v_y	0.000	0.033	0.033
	v_z	0.000	-0.218	0.218

TABLE II: Reconstruction during occlusion (object-region crop, evaluation cameras withheld from training). GT trajectory is the upper bound.

	PSNR ↑ (dB)				LPIPS ↓			
Method	Fall	Bnc	Roll	Mean	Fall	Bnc	Roll	Mean
GT (upper bnd)	16.08	18.33	17.60	17.34	.403	.203	.247	.284
Ours	**15.76**	**18.02**	**17.67**	**17.15**	**.405**	**.279**	**.257**	**.314**
Linear interp.	11.98	17.75	17.50	15.74	.500	.339	.305	.381
Const. velocity	14.13	14.93	15.01	14.69	.555	.469	.450	.491
No physics	10.63	12.63	12.77	12.01	.761	.716	.669	.716

The physics advantage is scene-dependent. On **ball_fall** (248-frame occlusion with a ground bounce), physics outperforms interpolation by $+3.78$ dB because the nonlinear contact trajectory cannot be recovered by a straight-line path. On **ball_bounce**, the hidden second bounce yields a $+3.09$ dB advantage over constant velocity. On **ball_roll** (45-frame, nearly linear occlusion), interpolation performs comparably ($+0.17$ dB gap), but constant velocity overshoots by -2.66 dB because it ignores friction. Fig. 3 shows qualitative results on ball_bounce.

Table III reports trajectory RMSE during occlusion. Constant velocity diverges on ball_fall (8.936) and ball_bounce (5.716), reflecting its inability to model contact interactions.

D. Ablations

Centroid vs. photometric loss. Table IV compares the centroid silhouette loss (Eq. 1) against masked pixel-wise MSE for physics parameter estimation. Centroid achieves 40% lower mean RMSE (0.586 vs. 0.984) and $+0.41$ dB higher PSNR. The advantage is strongest on ball_fall ($3.4\times$ lower RMSE), where centroid loss produces $\sim100\times$ larger friction gradients ($|\nabla_\mu| > 10$ vs. ~0.05) because centroid displacement is a first-order function of position while pixel-wise RGB differences arise only at silhouette boundaries. On ball_bounce, where friction is unobservable from pre-contact visible frames ($\partial\mathcal{L}/\partial\mu$=0 regardless of loss), both perform comparably.

Noise tolerance. Adding i.i.d. noise (σ) to the ground-truth trajectory yields graceful degradation (Table V): approximately 1 dB per σ=0.25 increment. Even at σ=2.0, PSNR exceeds the no-physics baseline (12.01 dB). Our estimated trajectory RMSE (0.29–1.15) falls within $\sigma\approx$0.3–

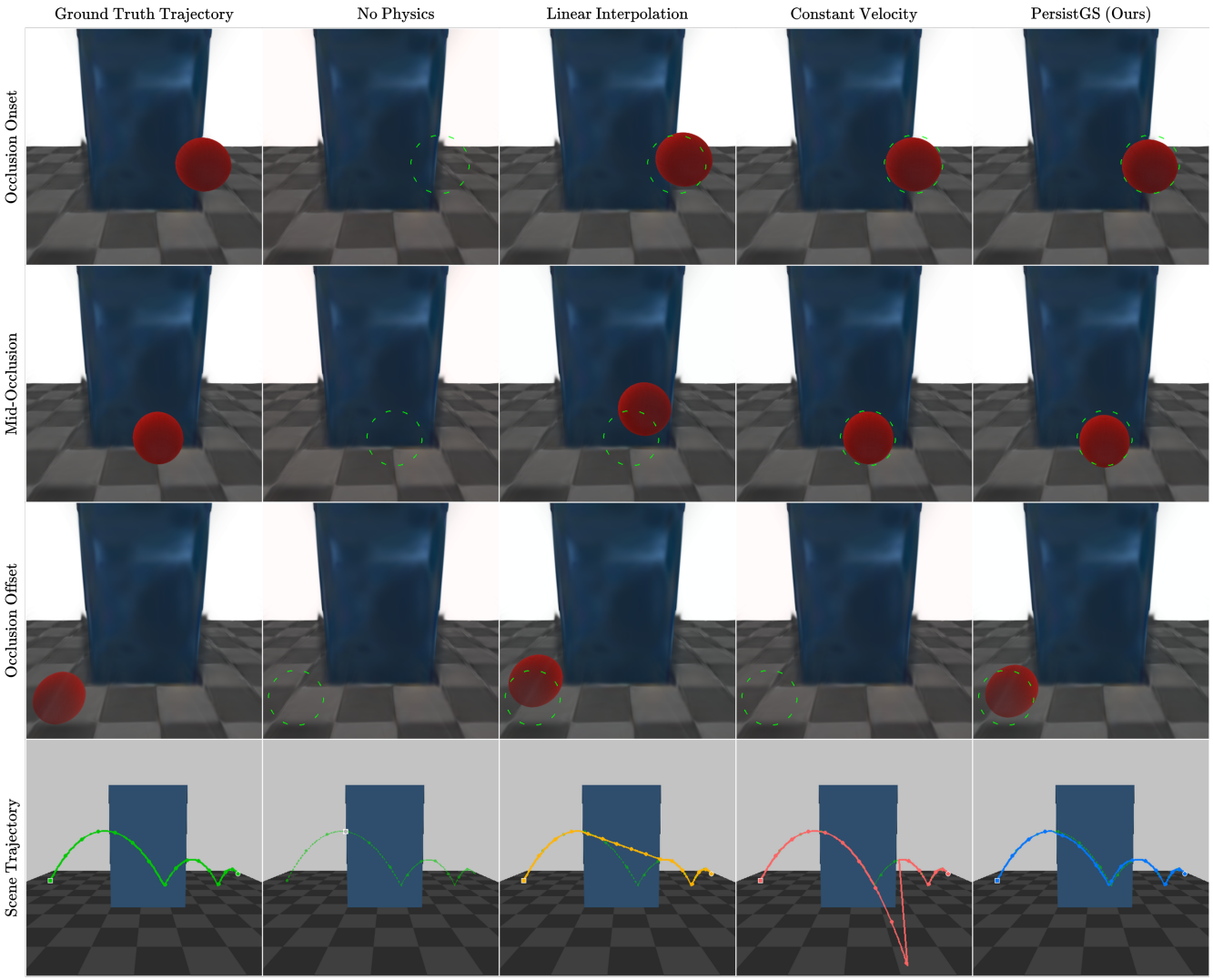

Fig. 3: Qualitative results on ball_bounce. Top three rows: renders from an evaluation camera (which sees past the occluder) at three stages of the occlusion event. Green dotted outlines indicate the ground-truth ball position. Without physics, the ball is absent; constant velocity misses the second bounce; linear interpolation follows a straight path through the nonlinear contact trajectory; PersistGS correctly tracks the physics-predicted arc. Bottom row: full-scene trajectories from a training viewpoint, with the ground-truth path (green) overlaid on each method's Gaussian reconstruction.

TABLE III: Trajectory RMSE during occlusion (lower is better).

Method	Fall	Bounce	Roll	Mean
Ours	**1.147**	**0.366**	0.288	**0.600**
Linear interp.	3.603	0.720	**0.222**	1.515
Const. velocity	8.936	5.716	1.427	5.360

TABLE IV: Centroid silhouette vs. photometric loss for physics estimation.

	Fall		Bounce		Roll		Mean	
Loss	RMSE↓	PSNR↑	RMSE↓	PSNR↑	RMSE↓	PSNR↑	RMSE↓	PSNR↑
Centroid (ours)	**0.61**	**16.44**	1.05	17.02	**0.10**	**17.57**	**0.59**	**17.01**
Photometric	2.08	15.59	**0.77**	**17.24**	0.10	16.96	0.98	16.60

1.2, confirming that the estimation accuracy is sufficient for high-quality reconstruction.

Occlusion duration. The physics advantage over constant velocity grows with occlusion length: on ball_roll, the gap widens from $+1.83$ dB (40 frames) to $+2.49$ dB (69 frames), because kinematic extrapolation accumulates error while physics respects contact constraints.

Visible frame count. Ball_fall PSNR improves from 13.07 dB (N=5 visible frames) to 16.20 dB (all frames), while ball_bounce is stable at $\sim$17.3 dB regardless of N. Complex trajectories benefit from more observations; simpler dynamics can be estimated from few frames.

Sparse-view regularization. Without regularization, ball_bounce and ball_roll degrade by $+3.0$ and $+4.2$ dB respectively on evaluation cameras, confirming that regular-

TABLE V: PSNR (dB) vs. trajectory noise σ (evaluation cameras, occlusion).

σ	Fall	Bounce	Roll
0.00	16.08	18.33	17.60
0.10	15.07	17.19	17.09
0.25	13.97	15.66	16.17
0.50	12.74	14.30	15.02
1.00	11.47	13.24	13.75
2.00	10.92	12.76	13.06

ization is essential when training views are clustered.

V. Discussion and Future Work

When does physics help? The advantage of physics over kinematic baselines scales with both occlusion duration and contact complexity. On ball_fall (248 frames, ground bounce during occlusion), physics outperforms even the non-causal linear interpolation by $+3.78$ dB, while on ball_roll (45 frames, nearly linear segment), kinematic models are competitive. The noise tolerance ablation reveals a smooth, monotonic relationship between trajectory accuracy and reconstruction quality ($\sim$1 dB per σ=0.25), indicating that even approximate physics estimates are preferable to kinematic assumptions when contacts are present. This suggests a practical decision criterion: physics priors are most valuable when the occluded interval contains contact events that produce nonlinear trajectory changes, and least necessary when the trajectory is approximately ballistic.

Parameter observability. Parameter identifiability depends on which physical interactions the cameras can observe. On ball_bounce, where first ground contact occurs during occlusion, friction is unobservable from pre-contact visible frames regardless of loss function. The curriculum addresses this by matching each parameter to the frames where it is identifiable. Preliminary two-pass experiments, where post-occlusion observations refine the physics parameters, show up to $+0.80$ dB improvement on ball_bounce by enabling friction estimation from re-emergence frames, suggesting that iterative refinement is a promising direction. More broadly, the identifiability analysis reveals a fundamental tension: the scenes where physics helps most (complex contacts during occlusion) are precisely those where the relevant parameters are hardest to estimate from visible frames alone.

Causality and practical applicability. Linear interpolation, despite competitive performance on two scenes, requires the re-emergence position, which is unavailable in any causal or predictive setting. Against constant velocity, the only causal baseline, physics provides a consistent $+2.46$ dB advantage, making PersistGS the strongest causal method evaluated. This causal property is essential for downstream applications such as robotic planning and digital twin maintenance, where an agent must reason about occluded object locations in real time without access to future observations. The decomposed architecture (static background plus per-object SE(3)) further supports such applications by providing an explicit object-level representation that can interface directly with a planner or controller.

Architecture. The decomposed architecture proved essential: a joint 4DGS baseline [2] overfits by $+6.9$ dB with only 5 training cameras, confirming that the sparse-view setting demands separation of static and dynamic components. This finding aligns with concurrent work on decomposed dynamic Gaussian representations [13], [15] and suggests that physics-based trajectory supervision is most naturally integrated within object-centric architectures where each rigid body has an explicit SE(3) trajectory to constrain.

Limitations and future work. Spherical objects provide a clean validation of the translational component of our centroid silhouette loss, since sphere silhouettes are rotation-invariant. For asymmetric objects, the silhouette covariance (second central moment of the alpha map) captures projected orientation and offers a natural extension to jointly constrain translation and rotation without relying on appearance. The full SE(3) pipeline, including rotation-aware SH evaluation, is implemented and exercised during reconstruction. Validating on objects where orientation produces distinct visual changes is a natural next step. Extending to multi-object scenes with inter-object contact, and integrating automatic decomposition methods [12] to remove the requirement for known object segmentation, are further directions.

VI. Conclusion

We presented PersistGS, a method that uses differentiable rigid body simulation as a physics prior for 4D Gaussian Splatting, maintaining object permanence through temporal occlusion. By estimating friction and velocity from visible frames via a centroid silhouette loss and observability-aware curriculum, and positioning object Gaussians along the resulting SE(3) trajectory, PersistGS produces faithful reconstructions through observation gaps. On three scenes with contact events during occlusion, it outperforms constant velocity extrapolation by $+2.46$ dB PSNR and comes within 0.19 dB of the ground-truth upper bound. The centroid loss yields 40% lower trajectory error than photometric supervision, and modifications to Newton's contact kernels enable effective gradient flow through rigid body dynamics. These results establish analytical physics simulation as a principled alternative to generative priors for 4D reconstruction from temporally incomplete observations.

References

[1] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics*, vol. 42, no. 4, July 2023. [Online]. Available: https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/

[2] G. Wu, T. Yi, J. Fang, L. Xie, X. Zhang, W. Wei, W. Liu, Q. Tian, and X. Wang, "4d gaussian splatting for real-time dynamic scene rendering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 20 310–20 320.

[3] Z. Yang, X. Gao, W. Zhou, S. Jiao, Y. Zhang, and X. Jin, "Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 20 331–20 341.

[4] Y.-H. Huang, Y.-T. Sun, Z. Yang, X. Lyu, Y.-P. Cao, and X. Qi, "Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes," *arXiv preprint arXiv:2312.14937*, 2023.

[5] Y. Yan, H. Lin, C. Zhou, W. Wang, H. Sun, K. Zhan, X. Lang, X. Zhou, and S. Peng, "Street gaussians: Modeling dynamic urban scenes with gaussian splatting," in *ECCV*, 2024.

[6] J. Luiten, G. Kopanas, B. Leibe, and D. Ramanan, "Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis," in *3DV*, 2024.

[7] P. Tokmakov, J. Li, W. Burgard, and A. Gaidon, "Learning to track with object permanence," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 10 860–10 869.

[8] S. Sabour, L. Goli, G. Kopanas, M. Matthews, D. Lagun, L. Guibas, A. Jacobson, D. Fleet, and A. Tagliasacchi, "Spotlesssplats: Ignoring distractors in 3d gaussian splatting," *ACM Trans. Graph.*, vol. 44, no. 2, Apr. 2025. [Online]. Available: https://doi.org/10.1145/3727143

[9] W.-H. Chu, L. Ke, J. Liu, M. Huo, P. Tokmakov, and K. Fragkiadaki, "Robust multi-object 4d generation for in-the-wild videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025, pp. 22 067–22 077.

[10] W.-H. Chu, L. Ke, and K. Fragkiadaki, "Dreamscene4d: Dynamic multi-object scene generation from monocular videos," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. [Online]. Available: https://openreview.net/forum?id=YlIvhHFwQ2

[11] Q. Wang, V. Ye, H. Gao, W. Zeng, J. Austin, Z. Li, and A. Kanazawa, "Shape of motion: 4d reconstruction from a single video," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2025, pp. 9660–9672.

[12] M. Ye, M. Danelljan, F. Yu, and L. Ke, "Gaussian grouping: Segment and edit anything in 3d scenes," in *ECCV*, 2024.

[13] J. Lee, C. Won, H. Jung, I. Bae, and H.-G. Jeon, "Fully explicit dynamic gaussian splatting," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. [Online]. Available: https://openreview.net/forum?id=g8pyTkxyIV

[14] D. Wan, R. Lu, and G. Zeng, "Superpoint Gaussian splatting for real-time high-fidelity dynamic scene reconstruction," in *Proceedings of the 41st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, Eds., vol. 235. PMLR, 21–27 Jul 2024, pp. 49 957–49 972. [Online]. Available: https://proceedings.mlr.press/v235/wan24f.html

[15] X. Zhou, Z. Lin, X. Shan, Y. Wang, D. Sun, and M.-H. Yang, "Drivinggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 21 634–21 643.

[16] Z. Yu, A. Chen, B. Huang, T. Sattler, and A. Geiger, "Mip-splatting: Alias-free 3d gaussian splatting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 19 447–19 456.

[17] H. Park, G. Ryu, and W. Kim, "Dropgaussian: Structural regularization for sparse-view gaussian splatting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025, pp. 21 600–21 609.

[18] J. Li, J. Zhang, X. Bai, J. Zheng, X. Ning, J. Zhou, and L. Gu, "Dngaussian: Optimizing sparse-view 3d gaussian radiance fields with global-local depth normalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 20 775–20 785.

[19] M. Kim, J. Lim, and B. Han, "4d gaussian splatting in the wild with uncertainty-aware regularization," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. [Online]. Available: https://openreview.net/forum?id=0sycTGl4In

[20] T. Wu, C. Zheng, F. Guan, A. Vedaldi, and T.-J. Cham, "Amodal3r: Amodal 3d reconstruction from occluded 2d images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2025, pp. 9181–9193.

[21] E. Ozguroglu, R. Liu, D. Surís, D. Chen, A. Dave, P. Tokmakov, and C. Vondrick, "pix2gestalt: Amodal segmentation by synthesizing wholes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 3931–3940.

[22] S. Sabour, S. Vora, D. Duckworth, I. Krasin, D. J. Fleet, and A. Tagliasacchi, "Robustnerf: Ignoring distractors with robust losses," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 20 626–20 636.

[23] K. M. Jatavallabhula, M. Macklin, F. Golemo, V. Voleti, L. Petrini, M. Weiss, B. Considine, J. Parent-Levesque, K. Xie, K. Erleben, L. Paull, F. Shkurti, D. Nowrouzezahrai, and S. Fidler, "gradsim: Differentiable simulation for system identification and visuomotor control," *International Conference on Learning Representations (ICLR)*, 2021. [Online]. Available: https://openreview.net/forum?id=c_E8kFWfhp0

[24] X. Li, Y.-L. Qiao, P. Chen, K. Jatavallabhula, M. Lin, C. Jiang, and C. Gan, "Pac-nerf: Physics augmented continuum neural radiance fields for geometry-agnostic system identification," in *ICLR*, 2023.

[25] S. Le Cleac'h, H.-X. Yu, M. Guo, T. Howell, R. Gao, J. Wu, Z. Manchester, and M. Schwager, "Differentiable physics simulation of dynamics-augmented neural objects," *IEEE Robotics and Automation Letters*, vol. 8, no. 5, pp. 2780–2787, 2023.

[26] J. Ni, Y. Chen, B. Jing, N. Jiang, B. Wang, B. Dai, P. Li, Y. Zhu, S.-C. Zhu, and S. Huang, "Phyrecon: Physically plausible neural scene reconstruction," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. [Online]. Available: https://openreview.net/forum?id=QrE9QPq4ya

[27] T. Xie, Z. Zong, Y. Qiu, X. Li, Y. Feng, Y. Yang, and C. Jiang, "Physgaussian: Physics-integrated 3d gaussians for generative dynamics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 4389–4398.

[28] Y. Lin, C. Lin, J. Xu, and Y. MU, "OmniphysGS: 3d constitutive gaussians for general physics-based dynamics generation," in *The Thirteenth International Conference on Learning Representations*, 2025. [Online]. Available: https://openreview.net/forum?id=9HZtP6I5lv

[29] J. Cai, Y. Yang, W. Yuan, Y. He, Z. Dong, L. Bo, H. Cheng, and Q. Chen, "Gic: Gaussian-informed continuum for physical property identification and simulation," in *Advances in Neural Information Processing Systems*, A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, Eds., vol. 37. Curran Associates, Inc., 2024, pp. 75 035–75 063. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2024/file/89379d5fc6eb34ff98488202fb52b9d0-Paper-Conference.pdf

[30] L. Zhong, H.-X. Yu, J. Wu, and Y. Li, "Reconstruction and simulation of elastic objects with spring-mass 3d gaussians," *European Conference on Computer Vision (ECCV)*, 2024.

[31] J. Cao, S. Guan, Y. Ge, W. Li, X. Yang, and C. Ma, "NeuMA: Neural material adaptor for visual grounding of intrinsic dynamics," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. [Online]. Available: https://openreview.net/forum?id=AvWB40qXZh

[32] R.-Z. Qiu, G. Yang, W. Zeng, and X. Wang, "Language-driven physics-based scene synthesis and editing via feature splatting," in *European Conference on Computer Vision (ECCV)*, 2024.

[33] H. Jiang, H.-Y. Hsu, K. Zhang, H.-N. Yu, S. Wang, and Y. Li, "Phystwin: Physics-informed reconstruction and simulation of deformable objects from videos," *ICCV*, 2025.

[34] Y. Rubanova, T. Lopez-Guevara, K. R. Allen, W. F. Whitney, K. Stachenfeld, and T. Pfaff, "Learning rigid-body simulators over implicit shapes for large-scale scenes and vision," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. [Online]. Available: https://openreview.net/forum?id=QDYts5dYgq

[35] C. Chen, Z. Dou, C. Wang, Y. Huang, A. Chen, Q. Feng, J. Gu, and L. Liu, "Vid2sim: Generalizable, video-based reconstruction of appearance, geometry and physics for mesh-free simulation," in *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, June 2025, pp. 26 545–26 555.

[36] J. Abou-Chakra, K. Rana, F. Dayoub, and N. Suenderhauf, "Physically embodied gaussian splatting: A realtime correctable world model for robotics," in *8th Annual Conference on Robot Learning*, 2024. [Online]. Available: https://openreview.net/forum?id=AEq0onGrN2

[37] B. Moran, M. Comi, A. Byravan, S. Bohez, T. Erez, Z. Li, and L. Hasenclever, "Splatting physical scenes: End-to-end real-to-sim from imperfect robot data," 2025. [Online]. Available: https://arxiv.org/abs/2506.04120

[38] J. Yu, K. Hari, K. El-Refai, A. Dalil, J. Kerr, C.-M. Kim, R. Cheng, M. Z. Irshad, and K. Goldberg, "Persistent object gaussian splat (pogs) for tracking human and robot manipulation of irregularly shaped objects," *ICRA*, 2025.

[39] B. Bianchini, M. Zhu, M. Sun, B. Jiang, C. J. Taylor, and M. Posa, "Vysics: Object reconstruction under occlusion by fusing vision and

contact-rich physics," in *Robotics: Science and Systems (RSS)*, june 2025.

[40] L. Yang, B. Kang, Z. Huang, Z. Zhao, X. Xu, J. Feng, and H. Zhao, "Depth anything v2," in *Advances in Neural Information Processing Systems*, A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, Eds., vol. 37. Curran Associates, Inc., 2024, pp. 21 875–21 911. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2024/file/26cfdcd8fe6fd75cc53e92963a656c58-Paper-Conference.pdf

[41] M. Macklin, "Warp: A high-performance python framework for gpu simulation and graphics," Mar. 2022, presented at the NVIDIA GPU Technology Conference (GTC). [Online]. Available: https://github.com/NVIDIA/warp

[42] The Newton Contributors, "Newton: Gpu-accelerated physics simulation for robotics and simulation research," https://github.com/newton-physics/newton, 2025, released April 22, 2025. Apache-2.0 License.