BEYOND OBSERVED RAYS: QUERY-GUIDED COMPLETION OF LAYERED GAUSSIAN DRIVING WORLDS

Anonymous authors
Paper under double-blind review

ABSTRACT

Driving-scene reconstruction with 3D Gaussian Splatting (3DGS) turns sensor logs into 3D environments, making recorded driving scenes renderable and editable for closed-loop simulation. However, reconstructing scenes solely from recorded camera observations leaves weakly observed or unseen object surfaces and occluded background regions poorly reconstructed. During closed-loop simulation, ego-vehicle motion may shift the camera away from the recorded trajectory, exposing previously unobserved regions and associated reconstruction defects, which can compromise simulation reliability. We introduce FULLSPLAT, a *task-query completion* framework that uses expected exposure within a bounded camera–action distribution to identify low-support regions that later queries may expose and complete them in a reusable 3D scene. A structured 3D prior generates object-canonical Gaussian residuals that follow rigid object motion, while masked rectified flow generates world-fixed background residuals; cross-layer conflict resolution and observation-preserving gates integrate both into one persistent scene for later rendering and editing without further generation. Experiments on Waymo and nuScenes cover reconstruction and novel-view synthesis, with rigid-object edits and 4–6 m camera-offset stress tests beyond the 3 m range on Waymo. Results show better captured-view fidelity, novel-view quality, and temporal consistency, with improvements under object translation, rotation, and removal and in views beyond the completion range.

1 INTRODUCTION

High-fidelity visual simulation provides a controllable environment for developing and evaluating autonomous-driving systems. Vehicle sensor logs help these environments retain the appearance and spatial layout of real roads (Chen et al., 2025b; Yang et al., 2023; Tonderski et al., 2024; Yang et al., 2024b). 3D Gaussian Splatting (3DGS) provides an explicit scene representation with real-time rendering (Kerbl et al., 2023). Layered driving reconstruction separates background and objects so that object poses can be controlled independently (Peng et al., 2026; Chen et al., 2025b). This lets simulators change viewpoints and object configurations, rather than just replay recorded images. The scene must therefore represent content exposed by these changes while preserving the appearance and structure supported by the original observations.

For novel views beyond observed viewpoints, camera or object motion reveals object surfaces and occluded background absent from recorded images. Driving cameras sample limited rays, leaving object backs and occluded background weakly constrained or unseen. For example, a reconstruction may lack both a road sign's back and the background it occludes. Neither appears in renderings from observed viewpoints; camera changes and object edits expose these gaps. Existing methods use generative priors for off-trajectory restoration or complete regions exposed by local edits (Ni et al., 2025a; Xu et al., 2025; Wang et al., 2025). These approaches can improve a persistent 3D reconstruction, but their restoration targets are typically defined by selected views, objects, or local scene regions. A realistic repair of one view or target does not guarantee geometric, appearance, and cross-view consistency under later viewpoints and object actions. We therefore need to identify which hidden regions later queries expose and how their completions fit together. The results

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

must meet geometric and appearance constraints, with object surfaces following their instances and revealed background fixed in the world. They must then form one reusable scene state.

We introduce FULLSPLAT for *task-query completion* from a layered Gaussian reconstruction. Our camera–action query distribution covers vehicle-accessible camera poses near the recorded route and feasible rigid-object edits. It captures viewpoint changes and object operations in simulation-based evaluation. Expected exposure selects regions that lack observation support but may become visible under these queries. With the existing reconstruction frozen, we add Gaussian residuals in two coordinate frames. A structured 3D asset prior uses observed surfaces and scene context to complete unseen object parts in canonical coordinates. These 3D Gaussian residuals follow rigid object transformations. A masked rectified-flow model completes occluded background in world coordinates, keeping observed context fixed throughout generation. A cross-layer resolver handles occupancy and visibility conflicts. Observation checks protect well-observed content. We commit accepted completions once. Later camera queries and rigid edits reuse the state, with completed surfaces following objects and background staying fixed.

Our contributions are as follows:

- We introduce task-query completion that turns later camera views and rigid-object edits into 3D completion needs. Expected exposure and observation support limit targets to low-support regions that may become visible within the query range, without completing the entire unseen scene.
- We develop a scene-conditioned dual-frame 3D generative model that combines observations and 3D priors to directly generate Gaussian residuals in object-canonical and world coordinates. Completed surfaces follow object transforms while occluded background stays fixed, supporting later views and rigid edits rather than single-view repair.
- We introduce observation-constrained cross-layer integration that resolves occupancy and occlusion conflicts between separately generated object and background layers. Cumulative observation checks limit the effects of new content on existing observations, allowing the shared scene to support later rendering and editing without further generation.

2 RELATED WORK

3DGS reconstruction of driving scenes. Explicit Gaussian representations offer real-time rendering and high-fidelity appearance, making them a practical basis for driving-scene reconstruction (Kerbl et al., 2023). Extensions to dynamic driving logs usually model the static background and tracked objects separately, supporting reconstruction, controlled replay, and object-level editing (Zhou et al., 2024; Chen et al., 2025b; Xiong et al., 2026; Yan et al., 2025a; Peng et al., 2026). Because these pipelines are optimized mainly from captured data, they cannot reliably reconstruct surfaces without observation support. Once a query camera moves away from the recorded views, missing support causes large holes, distorted geometry, and rendering artifacts that reduce reconstruction fidelity (Han et al., 2025).

Novel-view synthesis and off-trajectory restoration. Generative priors regularize or refine 3D reconstructions for novel view synthesis (NVS) (Wu et al., 2024; 2025). To address reconstruction defects away from observed views, off-trajectory driving NVS uses diffusion priors to repair views outside the logged path, extending the camera range supported by the reconstruction. Existing systems either repair extrapolated images online or progressively extend the reconstruction from nearby to more distant views (Ni et al., 2025a; Zhao et al., 2025b; Fan et al., 2025). Other methods distill repaired trajectory videos into 3D representations or select generated supervision using pixel-level quality and information-gain estimates (Yang et al., 2025b; Du et al., 2025; Wang et al., 2026b). These strategies improve rendering along selected camera trajectories, but the target views still define where repair is performed. Supporting object edits also requires accounting for the gaps exposed by the edit. Even at a logged camera pose, object rotation reveals hidden surfaces, while translation or removal reveals previously occluded background. We therefore define each query with both a camera pose and a feasible rigid action. We preserve the captured rig's calibration and sample virtual poses within a fixed, bounded, collision-free neighborhood of the recorded route.

Diffusion models and 3D completion. When captured evidence is incomplete, generative models can use image, cross-view, or temporal cues to infer missing content. Image priors support object reconstruction from sparse views (Yang et al., 2024a) and surface completion from partial point clouds (Kasten et al., 2023). Some methods infer a complete Gaussian scene from a single image (Liao et al., 2025), while others distill multiview repairs into 3D representations (Mirzaei et al., 2023; Weber et al., 2024; Huang et al., 2025; Shi et al., 2025). StreetUnveiler completes background exposed by object removal (Xu et al., 2025), and GS-RoadPatching searches and places Gaussian patches from reconstructed regions (Chen et al., 2025a). Other methods incorporate surrounding scene constraints into object completion (Khademi & Fuxin, 2025), combine instance decomposition with completion (Ni et al., 2025b; Yang et al., 2025a), or restore an object and its contact region before interaction (Wang et al., 2025). Supporting later camera–action queries requires coordinated completion of the missing object surfaces and occluded background they expose. We derive these completion targets from expected exposure and integrate both generated layers under cross-layer consistency and observation constraints. Our 3D asset prior and masked rectified-flow background model store their completions in a reusable Gaussian scene, with object surfaces following their instances and background fixed in the world.

3 METHOD

3.1 TASK AND OVERVIEW

FULLSPLAT adds missing object surfaces and occluded background to a partially observed driving scene, forming one reusable 3D state for later camera views and rigid-object edits (Figure 1). The input evidence $\mathcal{O}_E$ contains images, camera calibration and poses, and measured depth where available. A frozen reconstruction front end provides a layered Gaussian scene $\mathcal{G}_0$, separating static background from objects whose unseen surfaces may become visible within the query range (Peng et al., 2026; Chen et al., 2025b).

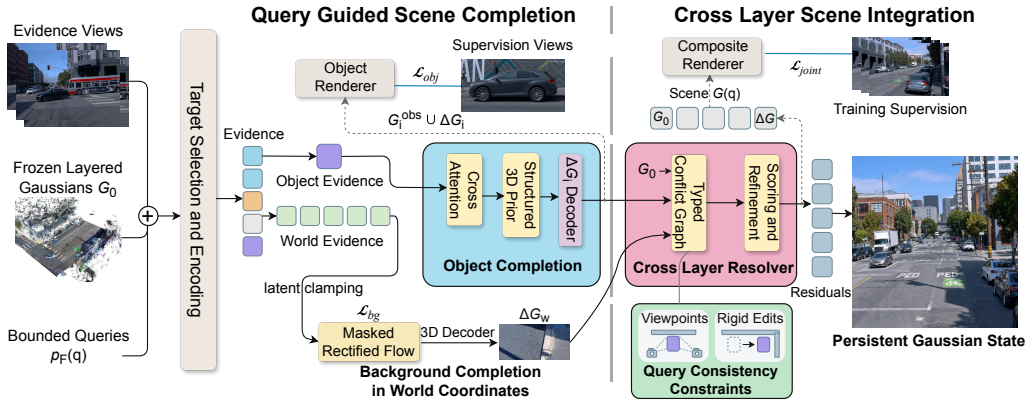

Figure 1: Task-query completion. Expected exposure identifies missing content. Shared evidence conditions object-canonical and world-fixed generators; cross-layer resolution and observation checks integrate their residuals into one reusable state. Auxiliary branches show training supervision.

A query $q = (p, a)$ specifies the camera pose p to render and object assignments a that keep, translate, rotate, or deactivate each instance. We sample queries from p_F, keeping the captured camera calibration and restricting camera poses to a bounded, collision-free region near the recorded route. Dynamic objects follow given tracks; we do not predict their future motion.

The completion model C_θ constructs one reusable state:

$$\mathcal{W}^* = C_\theta(\mathcal{G}_0, \mathcal{O}_E, p_F). \tag{1}$$

We fill query-relevant gaps with additional Gaussians (residuals) while preserving supported observations.

3.2 QUERY-GUIDED TARGET SELECTION

We identify missing regions that may become visible under the camera and action queries in p_F. For an object-grid cell c, support $S_i^{\mathrm{obs}}(c, \omega) \in [0, 1]$ measures depth-consistent evidence from direction ω. It is directional: observing the front does not establish support for the back. World-grid support $S_{\mathrm{w}}^{\mathrm{obs}}(c) \in [0, 1]$ combines static surface hits, measured free space, and agreement across views. Both fields are computed from $\mathcal{O}_E$ and then fixed.

For each query, we cast rays from the query camera toward the object and use observed geometry, tracked boxes, and coarse object shapes to record each ray's first-hit object-grid cell in $V_i(q)$. If camera motion or an object edit reveals previously occluded background, the segment between the former occluding surface and the next reliable geometry hit defines the newly exposed interval along that ray. Before computing demand, we merge overlapping intervals and define $J(q)$ as the world-grid cells intersecting them. The object and background completion demands are

$$\begin{aligned} E_i^{\mathrm{obj}}(c) &= \mathbb{E}_{q\sim p_F}\left[(1 - S_i^{\mathrm{obs}}(c, \omega_q))\alpha_q(c)\mathbf{1}\{c \in V_i(q)\}\right], \\ E_{\mathrm{w}}^{\mathrm{bg}}(c) &= \mathbb{E}_{q\sim p_F}\left[(1 - S_{\mathrm{w}}^{\mathrm{obs}}(c))\alpha_q(c)\mathbf{1}\{c \in J(q)\}\right]. \end{aligned} \tag{2}$$

Here ω_q is the viewing direction at the object cell in its canonical frame; $\alpha_q(c) \geq 0$ weights projected area, visibility, and confidence in the object/background assignment. Camera/action sampling weights enter only through p_F. We keep background regions supported by observations without generating them again. Within the declared query range, $\mathcal{C}$ includes all low-support regions that coarse object shapes or background reveal intervals identify as exposed. The object and background demand fields provide spatial conditions for their respective generators. Each cell in $\mathcal{C}$ records its layer: object cells belong to a specific instance and use its canonical frame; background cells use world coordinates. Image context and coarse object shapes can condition generation, but are less reliable than measurements such as depth. Unobserved regions are unknown, not necessarily empty. The generators infer occupancy within the target regions. Appendix B details the coarse object shapes and target construction.

3.3 SCENE-CONDITIONED DUAL-FRAME GENERATION

We generate object residuals in normalized object-canonical coordinates so they follow edits, and background residuals in world coordinates so they stay fixed (Figure 2). The map $A_i(x) = R_i D_i x + t_i$ places canonical point x at the recorded pose: R_i is rotation, D_i positive diagonal box scale, and t_i translation. A_i uses the replay timestamp, not flow time.

A shared encoder converts partial Gaussians, calibrated images, measured rays, free space, boxes, and road context into tokens H_i, H_{w} in the respective frames. Owner and observed/generated labels are retained. Generators F and Gaussian decoders D produce candidate residuals:

$$\begin{aligned} \widehat{\Delta\mathcal{G}}_i &= D_{\mathrm{obj}}\Big(F_{\mathrm{obj}}(H_i, E_i^{\mathrm{obj}})\Big), \\ \widehat{\Delta\mathcal{G}}_{\mathrm{w}} &= D_{\mathrm{bg}}\big(F_{\mathrm{bg}}(H_{\mathrm{w}}, E_{\mathrm{w}}^{\mathrm{bg}})\big). \end{aligned} \tag{3}$$

Random draws are implicit and use a fixed scene seed.

The object branch adapts a pretrained structured 3D prior (Xiang et al., 2025) with partial-evidence adapters and cross-attention. Given H_i and object demand, it first selects occupied canonical cells, then uses conditional flow to generate their geometry–appearance latents. The decoder converts these latents into Gaussian centers, covariances, opacity, and radiance.

The background branch uses a sparse world grid over the merged reveal intervals. Given H_{w} and background demand, masked rectified flow (Liu et al., 2023) generates hidden-cell latents. The decoder predicts occupancy and then Gaussian attributes. Road geometry and measured free space constrain where geometry can be added; images and semantic context guide its appearance.

The branches share evidence but use separate generative weights, latents, and decoders. Both keep observed Gaussians unchanged and reset observed latents to their input values at every flow step.

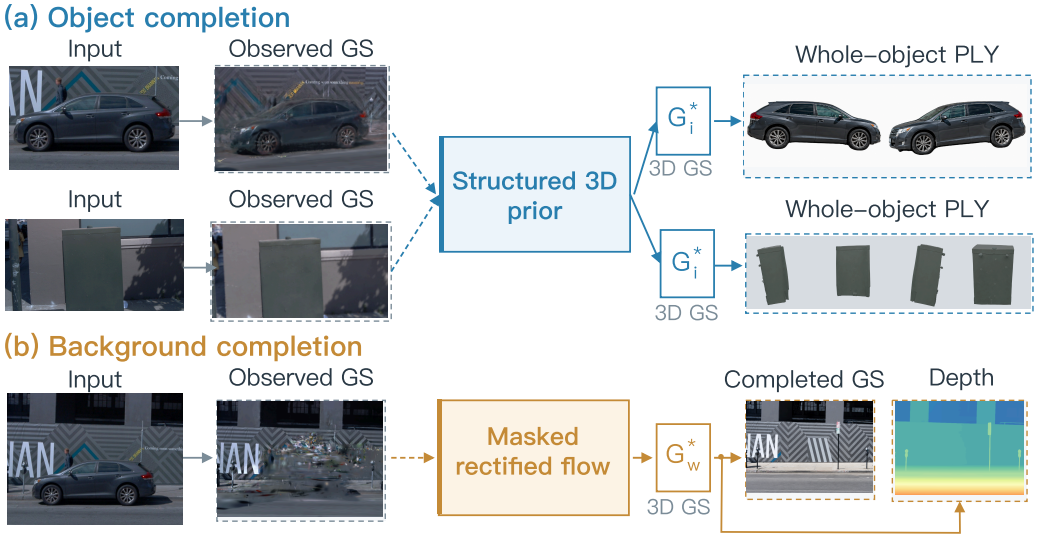

Figure 2: Dual-frame completion. The object branch generates canonical geometry; the background branch produces world-fixed geometry, shown as RGB and depth. Both retain observed anchors and pass the integration checks in Section 3.4. Dashed arrows denote conditioning.

3.4 Cross-layer consistency and observation protection

We jointly resolve object–background overlaps and depth-order conflicts across sampled queries. Rigid edits $Q_i \in SE(3)$ map active object candidates through $Q_i A_i$ into world coordinates. A graph separates object candidates, background candidates, and observed Gaussians by node type.

Differentiable rendering of both layers measures depth-order, first-hit ownership, and occupancy differences, which we combine across queries with visibility and demand weights into shared constraints for each candidate.

The resolver scores each candidate's agreement with observed evidence, same-layer structure, and cross-layer boundaries as η_g. Bounded refinement reduces unintended interpenetration and inconsistent occlusion in the shared candidates. Only generated content changes; layer assignments and observed Gaussians stay fixed. Object updates return to canonical coordinates; background stays world-fixed. Appendix D gives the distinct position and gradient maps needed for the scale in A_i.

Added Gaussians can obscure measured surfaces, so we check combined renderings at captured poses. Candidate g and accepted additions $\Delta\mathcal{G}_{\text{acc}}$ use world coordinates and recorded object poses. Starting with an empty accepted set, we require

$$\begin{aligned} &\bar{S}^{\text{obs}}(g) < \tau_{\text{obs}}, \qquad && M_{\text{meas}}(g) = 0, \\ &\eta_g > \tau_{\text{accept}}, && J_{\text{obs}}(\mathcal{G}_0 \cup \Delta\mathcal{G}_{\text{acc}} \cup \{g\}; \mathcal{G}_0) \le \epsilon_{\text{obs}}. \end{aligned} \tag{4}$$

The gate requires low support, no measured occupancy/free-space conflict, and a sufficient resolver score. Only the score depends on the resolver; observation checks are independent. For object candidates, $\bar{S}^{\text{obs}}$ is the maximum support over intersected observation directions; background candidates use world support. J_{obs} measures the largest normalized change in RGB, depth, transmittance, and first-hit ownership on supported rays relative to $\mathcal{G}_0$. Including all accepted additions bounds accumulated changes (Appendix B).

3.5 Learning and inference

Training hides surfaces supported by complementary views or screened full-log reconstructions; cells without targets receive no latent or geometry loss. Disjoint binary masks $M_{\text{obs}} + M_{\text{hid}} = 1$ separate observed latents z_{obs} from hidden targets z_{hid}. Observed values stay fixed. With noise ϵ,

hidden cells follow a linear path and the conditional field v_θ learns its velocity (Lipman et al., 2023):

$$z_t = M_{\text{obs}} z_{\text{obs}} + M_{\text{hid}}\big((1-t)\epsilon + t z_{\text{hid}}\big), \qquad t \in [0,1], \tag{5}$$

$$\mathcal{L}_{\text{flow}} = \mathbb{E}\left\| M_{\text{hid}}\big(v_\theta(z_t, t; H, E) - (z_{\text{hid}} - \epsilon)\big)\right\|_2^2. \tag{6}$$

Expectation covers training examples, noise, and uniform $t \in [0,1]$. The full training objective is

$$\mathcal{L} = \mathcal{L}_{\text{obj}} + \mathcal{L}_{\text{bg}} + \mathcal{L}_{\text{joint}} + \lambda_{\text{p}} \mathcal{L}_{\text{partition}}. \tag{7}$$

Object supervision uses held-out appearance, silhouettes, and supported depth/geometry; background supervision uses measured reveals, depth, semantics, and boundaries. Joint supervision checks rendering and cross-layer conflicts. All three losses penalize occupied free space. $\mathcal{L}_{\text{partition}}$ aligns occupancy and type logits on shared cells from two query subsets with the same seed. We train the object branch, background branch, then resolver, updating only added adapters, cross-attention, flow modules, and decoders. The base, ownership parser, and pretrained backbone stay frozen (Appendix D).

Inference and state reuse. After the checks in Equation 4, the unmet demand is

$$r(\mathcal{W}) = \frac{\sum_{c \in \mathcal{C}} E(c)(1 - C_{\mathcal{W}}(c))}{\sum_{c \in \mathcal{C}} E(c)}. \tag{8}$$

Coverage $C_{\mathcal{W}}(c) \in [0,1]$ checks first-hit ownership, depth, and opacity across queries exposing c, weighted by exposure (Appendix B). Targets and weights stay fixed. After bounded refinement, we commit only if $r(\mathcal{W}) \le 1 - \rho$; otherwise, we abstain. We fix $\rho \in (0,1]$ on validation scenes; $\rho < 1$ allows some weighted demand to remain unmet. Accepted additions extend their layers:

$$\begin{aligned} \mathcal{G}_{\text{w}}^{*} &= \mathcal{G}_{\text{w}}^{\text{obs}} \cup \Delta\mathcal{G}_{\text{w}}^{\text{gen}}, \\ \mathcal{G}_{i}^{*} &= \mathcal{G}_{i}^{\text{obs}} \cup \Delta\mathcal{G}_{i}^{\text{gen}}, \\ \mathcal{G}(q; \mathcal{W}^{*}) &= \mathcal{G}_{\text{w}}^{*} \cup \bigcup\nolimits_{i:b_i=1} Q_i A_i(\mathcal{G}_{i}^{*}). \end{aligned} \tag{9}$$

Queries set active flags $b_i \in \{0,1\}$ and transforms Q_i, initially $1, I$; splatting renders the stored layers without further generation (Kerbl et al., 2023). State-reuse checks are in Appendix I.

4 Experiments

We evaluate reconstruction, novel views, and rigid edits, with mechanism ablations.

Table 1: Captured-view reconstruction (0 m) and NVS (±3 m). Only 0 m views have paired RGB targets. PSNR: dB; NTA/NTL: IoU (%). PSNR/SSIM/NTA/NTL ↑; LPIPS/FID ↓.

Method	Waymo						nuScenes					
	Original (0 m)			Novel (±3 m)			Original (0 m)			Novel (±3 m)		
	PSNR	SSIM	LPIPS	FID	NTA	NTL	PSNR	SSIM	LPIPS	FID	NTA	NTL
Street Gaussians	29.40	<u>0.8969</u>	<u>0.200</u>	109.69	44.0	46.85	<u>33.73</u>	0.9404	0.126	195.06	30.3	43.08
OmniRe	27.88	0.8839	0.210	188.42	37.7	45.10	31.83	0.9216	0.148	237.25	26.7	41.61
StreetCrafter-G	27.08	0.8611	0.254	76.45	51.1	49.05	30.32	0.9056	0.159	168.84	32.2	40.01
ReconDreamer++	<u>30.93</u>	0.8939	0.217	72.02	<u>52.1</u>	52.44	33.67	<u>0.9450</u>	<u>0.122</u>	<u>136.67</u>	<u>34.6</u>	<u>46.43</u>
FaithFusion	29.86	0.8883	0.226	<u>71.51</u>	50.4	<u>54.21</u>	33.28	0.9318	0.135	140.76	32.4	43.87
Ours	**32.71**	**0.9146**	**0.165**	**61.71**	**58.8**	**58.60**	**34.95**	**0.9575**	**0.111**	**105.20**	**37.8**	**52.20**

4.1 Experimental setup

Data and queries. We use location-disjoint training, validation, and test logs from Waymo and nuScenes (Sun et al., 2020; Caesar et al., 2020). Captured views (0 m) and valid road-aligned ± 3 m novel views are evaluated on both datasets; rigid edits of vehicles and roadside objects use Waymo. FullSplat completes each scene using only $\mathcal{O}_E$ before evaluation and reuses it without query-time generation. Completion and evaluation queries are sampled separately from the same valid range (Appendix A). Per-scene fitting may use captured 0 m frames in $\mathcal{O}_E$, but not held-out targets; adaptation of pretrained priors uses training logs only.

Baselines. Street Gaussians (Yan et al., 2025a) and OmniRe (Chen et al., 2025b) compare reconstruction. StreetCrafter-G (Yan et al., 2025b), ReconDreamer++ (Zhao et al., 2025b), and FaithFusion (Wang et al., 2026b) compare generative NVS. All methods are rerun with the same evidence access, evaluation queries, and resolution (Appendix F). StreetCrafter-G uses its distilled Gaussian output. Each NVS baseline retains its native reconstruction procedure; our method and its ablations share the same frozen OmniRe base $\mathcal{G}_0$. We evaluate reconstruction and NVS from stored 3D outputs without online image enhancement.

Metrics. Captured views use PSNR, SSIM (Wang et al., 2004), and LPIPS (Zhang et al., 2018). Virtual views have no paired RGB targets. FID (Heusel et al., 2017) and FVD (Unterthiner et al., 2019) use unpaired real images and clips from the corresponding test logs to compare image/video distributions. NTA-IoU/NTL-IoU (Zhao et al., 2025a) assess traffic layout by comparing detections with projected annotations. MUSIQ (Ke et al., 2021) and BRISQUE (Mittal et al., 2012) assess frame quality. These scores do not establish unseen-surface geometry. Reference pools and metric settings are in Appendices E and G.

4.2 Reconstruction and novel-view synthesis

Scene completion improves novel-view and captured-view quality on both datasets (Table 1). The novel-view gains are consistent with completing low-support object surfaces and occluded background that camera shifts expose. Captured-view gains are consistent with residuals improving sparsely observed, low-support regions that contribute to rendering. The base Gaussians stay fixed, and observation checks limit changes on supported rays. The gains in both FID and traffic-layout IoU indicate improved appearance without sacrificing vehicle and lane structure. The stored scene also remains useful beyond its completion range, without further generation (Figures 3 and 4).

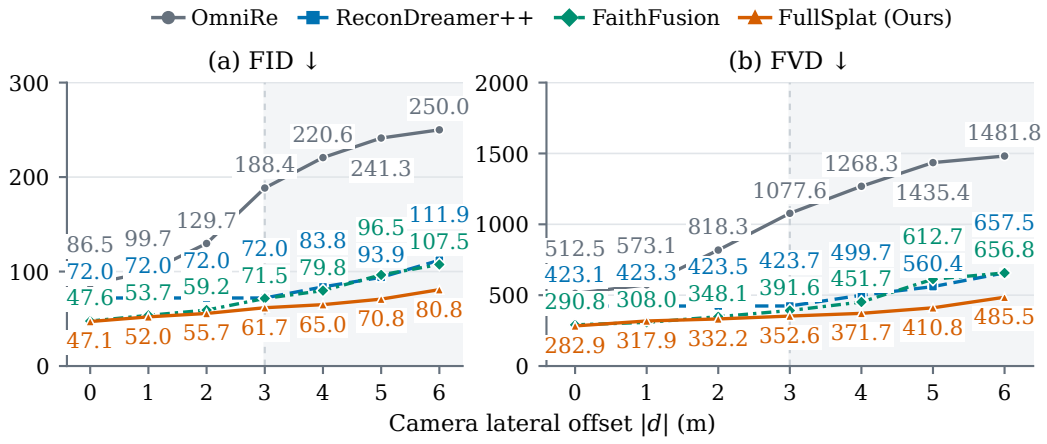

Figure 3: Camera-offset robustness on Waymo (lower is better). All offsets reuse the scene completed for the 3 m range. Shading marks 4–6 m stress tests beyond this range; these do not establish coverage of a wider query domain. Quality declines with distance. Our FID is lowest throughout; FVD is lowest except at 1 m, where FaithFusion performs better.



Figure 4: Novel-view rendering at lateral offsets of $+3$, $+4$, $+5$, and -6 m. Each column compares the original Gaussian scene $\mathcal{G}_0$ (top) with FULLSPLAT (bottom) at the same camera pose. Completion reduces visible vehicle distortions and road artifacts. All FULLSPLAT views reuse the scene completed for the 3 m range, without query-time generation.

4.3 DRIVING-VIDEO QUALITY UNDER RIGID-OBJECT EDITS

We test rigid edits of vehicles and roadside objects on Waymo, adapting HorizonSuite's categories (Wang et al., 2026a) to our valid rigid-action range. All methods share continuous camera paths, edit trajectories, frame rates, and clip lengths. Removal lasts throughout each clip.

Edits can expose incomplete object surfaces or occluded background; gains across translation, rotation, and removal support reusing both completed layers (Table 2 and Figure 5).

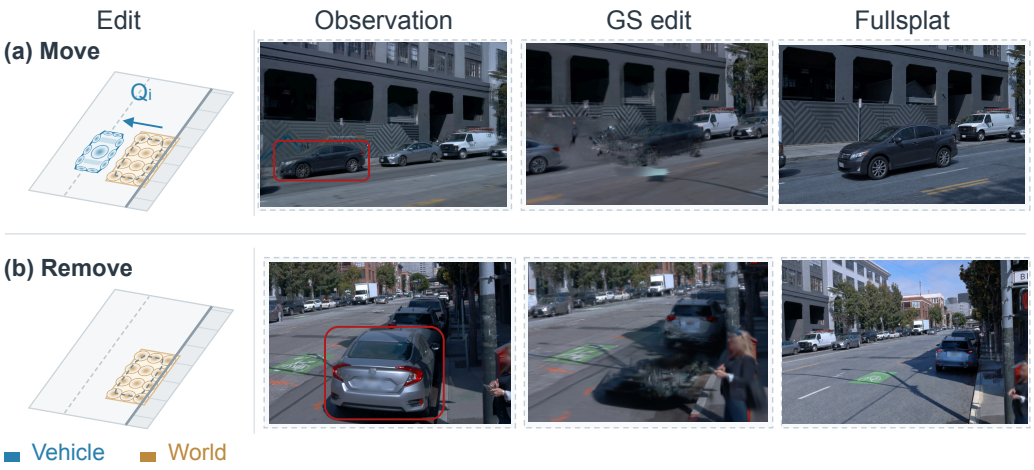

Figure 5: Vehicle editing with the same completed scene. Image columns compare the captured observation, direct editing of the reconstructed Gaussians, and FULLSPLAT. (a) Lateral motion exposes incomplete vehicle surfaces; (b) removal exposes road gaps in direct GS editing. FULLSPLAT's completed surfaces follow the object, while background stays world-fixed. Blue denotes objects, amber background, and dashed outlines the original pose.

4.4 MECHANISM ABLATIONS

We examine how each component affects captured-view fidelity and the quality of later scene edits. All variants complete scenes before testing without access to evaluation queries, using the same thresholds fixed on validation scenes. Rejected completions are evaluated using their base-scene outputs (Appendix H).

Table 2: Rigid-object edits on Waymo, including vehicles and roadside objects. FID/FVD: per category; MUSIQ/BRISQUE: equal scene/category averages. OmniRe and our method render stored scenes; † marks query-time diffusion, compared only for visual quality.

Method	Translation		Rotation		Removal		All edits	
	FID ↓	FVD ↓	FID ↓	FVD ↓	FID ↓	FVD ↓	MUSIQ ↑	BRISQUE ↓
OmniRe	183.34	1289.9	189.34	1419.6	178.52	1230.8	37.36	47.70
StreetCrafter†	150.32	1128.2	161.47	1209.8	145.37	1114.5	38.19	43.85
Difix†	<u>129.75</u>	<u>1009.3</u>	<u>143.29</u>	<u>1106.6</u>	<u>131.17</u>	<u>1015.2</u>	<u>43.76</u>	<u>41.89</u>
Ours	**98.60**	**720.0**	**103.20**	**755.0**	**94.10**	**690.0**	**47.90**	**35.60**

Query-guided target selection. Uniform target sampling selects eligible low-support regions without using their expected exposure. Evidence, generators, coordinate assignments, and integration remain fixed. Its lower edited-view quality shows the value of constructing targets from the views and actions the scene must support.

Table 3: Mechanism ablations on Waymo. All variants use the same scenes, frozen base, evidence, queries, and object assignments. PSNR (dB) and LPIPS evaluate reconstruction quality in unedited captured views (0 m). FID and FVD are computed on the joint edit pool; MUSIQ follows Table 2.

Variant	Original (0 m)		Rigid-object edits		
	PSNR ↑	LPIPS ↓	FID ↓	FVD ↓	MUSIQ ↑
Full model	**32.71**	**0.165**	**102.65**	**803.13**	**47.90**
Uniform target sampling	30.56	0.182	<u>114.19</u>	889.66	<u>43.72</u>
2D prior + GS distillation	30.64	0.183	118.23	905.10	42.96
w/o cross-layer resolution	<u>31.53</u>	<u>0.178</u>	114.50	840.50	43.45
w/o observation gate	28.48	0.183	115.39	<u>834.50</u>	42.12

3D prior versus 2D diffusion. We replace only the object branch: image-conditioned, depth-guided 2D diffusion generates views for fitting added canonical Gaussians (Yu et al., 2025). Observed anchors stay frozen; the background branch, resolver, and observation checks remain unchanged. Lower edit quality favors the 3D prior for object completion in this shared pipeline.

Cross-layer resolution. Removing the resolver, joint loss, and acceptance condition on η_g retains both generators and their individual losses. Support, measurement-conflict, and cumulative observation checks also remain active. Lower edited-view quality supports cross-layer resolution as a whole. These visual metrics do not directly count ownership or depth-order errors.

Observation protection. Disabling only J_{obs} keeps the base frozen and retains support, measured-occupancy, and free-space checks. The drop in captured-view quality shows that added content can obscure unchanged Gaussians, so the combined rendering must also be checked.

5 Conclusion

We presented FullSplat for query-guided completion of layered Gaussian driving scenes. Within a bounded camera–action range, it identifies missing object surfaces and occluded background that later queries may expose. Dual-frame generation keeps object completions attached to their instances and background world-fixed. Cross-layer resolution integrates both under observation constraints. Experiments on Waymo and nuScenes show improved novel-view quality while retaining captured-view quality, with further gains under rigid-object edits on Waymo. Permanently unseen geometry remains uncertain, and the method does not predict future motion. FullSplat thus completes the scene itself, allowing later views and edits to reuse the same 3D content without further generation.

AI USE STATEMENT

Generative AI tools were used solely for language editing of the manuscript. The authors take full responsibility for its final content.

REPRODUCIBILITY STATEMENT

The method and experimental setup are described in Sections 3 and 4, with additional implementation and evaluation details in Appendices D–G. We plan to release our implementation, configuration files, and data-preprocessing and evaluation scripts to facilitate reproduction of the reported results.

REFERENCES

Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11618–11628, 2020. doi: 10.1109/CVPR42600.2020.01164. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Caesar_nuScenes_A_Multimodal_Dataset_for_Autonomous_Driving_CVPR_2020_paper.html.

Guo Chen, Jiarun Liu, Sicong Du, Chenming Wu, Deqi Li, Shi-Sheng Huang, Guofeng Zhang, and Sheng Yang. GS-RoadPatching: Inpainting gaussians via 3D searching and placing for driving scenes. In *Proceedings of the SIGGRAPH Asia Conference Papers*, pp. 1–11, 2025a. doi: 10.1145/3757377.3763892. URL https://doi.org/10.1145/3757377.3763892.

Ziyu Chen, Jiawei Yang, Jiahui Huang, Riccardo de Lutio, Janick Martinez Esturo, Boris Ivanovic, Or Litany, Zan Gojcic, Sanja Fidler, Marco Pavone, Li Song, and Yue Wang. OmniRe: Omni urban scene reconstruction. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 85508–85527, 2025b. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/d4938496368e1c5806138bea8837595c-Paper-Conference.pdf.

Sicong Du, Jiarun Liu, Qifeng Chen, Hao-Xiang Chen, Tai-Jiang Mu, and Sheng Yang. RGE-GS: Reward-guided expansive driving scene reconstruction via diffusion priors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 25756–25764, 2025. doi: 10.1109/ICCV51701.2025.02389. URL https://openaccess.thecvf.com/content/ICCV2025/html/Du_RGE-GS_Reward-Guided_Expansive_Driving_Scene_Reconstruction_via_Diffusion_Priors_ICCV_2025_paper.html.

Lue Fan, Hao Zhang, Qitai Wang, Hongsheng Li, and Zhaoxiang Zhang. FreeSim: Toward free-viewpoint camera simulation in driving scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12004–12014, 2025. doi: 10.1109/CVPR52734.2025.01121. URL https://openaccess.thecvf.com/content/CVPR2025/html/Fan_FreeSim_Toward_Free-viewpoint_Camera_Simulation_in_Driving_Scenes_CVPR_2025_paper.html.

Xiangyu Han, Zhen Jia, Boyi Li, Yan Wang, Boris Ivanovic, Yurong You, Lingjie Liu, Yue Wang, Marco Pavone, Chen Feng, and Yiming Li. Extrapolated urban view synthesis benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 28718–28728, 2025. doi: 10.1109/ICCV51701.2025.02667. URL https://openaccess.thecvf.com/content/ICCV2025/html/Han_Extrapolated_Urban_View_Synthesis_Benchmark_ICCV_2025_paper.html.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/8a1d694707eb0fefe65871369074926d-Abstract.html.

Sheng-Yu Huang, Zi-Ting Chou, and Yu-Chiang Frank Wang. 3D gaussian inpainting with depth-guided cross-view consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26704–26713, 2025. doi: 10.1109/CVPR52734.2025.02487. URL https://openaccess.thecvf.com/content/CVPR2025/html/Huang_3D_Gaussian_Inpainting_with_Depth-Guided_Cross-View_Consistency_CVPR_2025_paper.html.

Yoni Kasten, Ohad Rahamim, and Gal Chechik. Point cloud completion with pretrained text-to-image diffusion models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 12171–12191, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/284afdc2309f9667d2d4fb9290235b0c-Abstract.html.

Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. MUSIQ: Multi-scale image quality transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5128–5137, 2021. doi: 10.1109/ICCV48922.2021.00510. URL https://openaccess.thecvf.com/content/ICCV2021/html/Ke_MUSIQ_Multi-Scale_Image_Quality_Transformer_ICCV_2021_paper.html.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):1–14, 2023. doi: 10.1145/3592433. URL https://doi.org/10.1145/3592433.

Wesley Khademi and Li Fuxin. Point-based instance completion with scene constraints. In *International Conference on Learning Representations (ICLR)*, pp. 16814–16844, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/2ac9a7afd788c75c159be5ca80496176-Abstract-Conference.html.

Ziwei Liao, Mohamed Sayed, Steven L. Waslander, Sara Vicente, Daniyar Turmukhambetov, and Michael Firman. Complete Gaussian Splats from a single image with denoising diffusion models. *arXiv preprint arXiv:2508.21542*, 2025. URL https://arxiv.org/abs/2508.21542.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations (ICLR)*. OpenReview.net, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z.

Ashkan Mirzaei, Tristan Aumentado-Armstrong, Konstantinos G. Derpanis, Jonathan Kelly, Marcus A. Brubaker, Igor Gilitschenski, and Alex Levinshtein. SPIn-NeRF: Multiview Segmentation and Perceptual Inpainting With Neural Radiance Fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20669–20679, June 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Mirzaei_SPIn-NeRF_Multiview_Segmentation_and_Perceptual_Inpainting_With_Neural_Radiance_Fields_CVPR_2023_paper.html.

Anish Mittal, Anush K. Moorthy, and Alan C. Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, 21(12):4695–4708, 2012. doi: 10.1109/TIP.2012.2214050. URL https://live.ece.utexas.edu/research/quality/brisque_journal.pdf.

Chaojun Ni, Guosheng Zhao, Xiaofeng Wang, Zheng Zhu, Wenkang Qin, Guan Huang, Chen Liu, Yuyin Chen, Yida Wang, Xueyang Zhang, Yifei Zhan, Kun Zhan, Peng Jia, Xianpeng Lang, Xingang Wang, and Wenjun Mei. ReconDreamer: Crafting world models for driving scene reconstruction via online restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1560–1569, 2025a. doi: 10.1109/CVPR52734.2025.00153. URL https://openaccess.thecvf.com/content/CVPR2025/html/Ni_ReconDreamer_Crafting_World_Models_for_Driving_Scene_Reconstruction_via_Online_CVPR_2025_paper.html.

Junfeng Ni, Yu Liu, Ruijie Lu, Zirui Zhou, Song-Chun Zhu, Yixin Chen, and Siyuan Huang. Decompositional neural scene reconstruction with generative diffusion prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6022–6033, 2025b. URL https://openaccess.thecvf.com/content/CVPR2025/html/Ni_Decompositional_Neural_Scene_Reconstruction_with_Generative_Diffusion_Prior_CVPR_2025_paper.html.

Sida Peng, Yushi Long, Yunzhi Yan, Haotong Lin, Chenxu Zhou, Haiyang Sun, Kun Zhan, Xianpeng Lang, Hujun Bao, and Xiaowei Zhou. Street Gaussians: Modeling dynamic urban scenes with gaussian primitives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(3): 3355–3367, March 2026. doi: 10.1109/TPAMI.2025.3634385. URL https://doi.org/10.1109/TPAMI.2025.3634385.

Zhihao Shi, Dong Huo, Yuhongze Zhou, Yan Min, Juwei Lu, and Xinxin Zuo. IMFine: 3D inpainting via geometry-guided multi-view refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26694–26703, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Shi_IMFine_3D_Inpainting_via_Geometry-guided_Multi-view_Refinement_CVPR_2025_paper.html.

Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurélien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2443–2451, 2020. doi: 10.1109/CVPR42600.2020.00252. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Sun_Scalability_in_Perception_for_Autonomous_Driving_Waymo_Open_Dataset_CVPR_2020_paper.html.

Adam Tonderski, Carl Lindström, Georg Hess, William Ljungbergh, Lennart Svensson, and Christoffer Petersson. NeuRAD: Neural Rendering for Autonomous Driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14895–14904, June 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Tonderski_NeuRAD_Neural_Rendering_for_Autonomous_Driving_CVPR_2024_paper.html.

Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. FVD: A new metric for video generation. In *Deep Generative Models for Highly Structured Data, ICLR 2019 Workshop*, 2019. URL https://openreview.net/forum?id=rylgEULtdN.

Miaowei Wang, Yibo Zhang, Weiwei Xu, Rui Ma, Changqing Zou, and Daniel Morris. DecoupledGaussian: Object-scene decoupling for physics-based interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11361–11372, 2025. doi: 10.1109/CVPR52734.2025.01061. URL https://openaccess.thecvf.com/content/CVPR2025/html/Wang_DecoupledGaussian_Object-Scene_Decoupling_for_Physics-Based_Interaction_CVPR_2025_paper.html.

Yifan Wang, Francesco Pittaluga, Zaid Tasneem, Chenyu You, Manmohan Chandraker, and Ziyu Jiang. HorizonForge: Driving scene editing with any trajectories and any vehicles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24895–24905, 2026a. URL https://openaccess.thecvf.com/content/CVPR2026/html/Wang_HorizonForge_Driving_Scene_Editing_with_Any_Trajectories_and_Any_Vehicles_CVPR_2026_paper.html.

YuAn Wang, Xiaofan Li, Chi Huang, Wenhao Zhang, Hao Li, Bosheng Wang, Xun Sun, and Jun Wang. FaithFusion: Harmonizing reconstruction and generation via pixel-wise information gain. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1198–1209, 2026b. URL https://openaccess.thecvf.com/content/CVPR2026/html/Wang_FaithFusion_Harmonizing_Reconstruction_and_Generation_via_Pixel-wise_Information_Gain_CVPR_2026_paper.html.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. doi: 10.1109/TIP.2003.819861. URL https://doi.org/10.1109/TIP.2003.819861.

Ethan Weber, Aleksander Holynski, Varun Jampani, Saurabh Saxena, Noah Snavely, Abhishek Kar, and Angjoo Kanazawa. NeRFiller: Completing Scenes via Generative 3D Inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20731–20741, June 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Weber_NeRFiller_Completing_Scenes_via_Generative_3D_Inpainting_CVPR_2024_paper.html.

Jay Zhangjie Wu, Yuxuan Zhang, Haithem Turki, Xuanchi Ren, Jun Gao, Mike Zheng Shou, Sanja Fidler, Zan Gojcic, and Huan Ling. DIFIX3D+: Improving 3D reconstructions with single-step diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26024–26035, 2025. doi: 10.1109/CVPR52734.2025.02424. URL https://openaccess.thecvf.com/content/CVPR2025/html/Wu_DIFIX3D_Improving_3D_Reconstructions_with_Single-Step_Diffusion_Models_CVPR_2025_paper.html.

Rundi Wu, Ben Mildenhall, Philipp Henzler, Keunhong Park, Ruiqi Gao, Daniel Watson, Pratul P. Srinivasan, Dor Verbin, Jonathan T. Barron, Ben Poole, and Aleksander Hołyński. ReconFusion: 3D Reconstruction with Diffusion Priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21551–21561, June 2024. doi: 10.1109/CVPR52733.2024.02036. URL https://openaccess.thecvf.com/content/CVPR2024/html/Wu_ReconFusion_3D_Reconstruction_with_Diffusion_Priors_CVPR_2024_paper.html.

Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3D latents for scalable and versatile 3D generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21469–21480, 2025. doi: 10.1109/CVPR52734.2025.02000. URL https://openaccess.thecvf.com/content/CVPR2025/html/Xiang_Structured_3D_Latents_for_Scalable_and_Versatile_3D_Generation_CVPR_2025_paper.html.

Yajiao Xiong, Xiaoyu Zhou, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. DrivingGaussian++: Toward realistic reconstruction and editable simulation for surrounding dynamic driving scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(7):7289–7306, 2026. doi: 10.1109/TPAMI.2026.3667072. URL https://doi.org/10.1109/TPAMI.2026.3667072.

Jingwei Xu, Yikai Wang, Yiqun Zhao, Yanwei Fu, and Shenghua Gao. 3D StreetUnveiler with semantic-aware 2DGS - a simple baseline. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 12027–12060, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/1fc6c343d8dbb4c369ab6e04225f5a65-Paper-Conference.pdf.

Yunzhi Yan, Haotong Lin, Chenxu Zhou, Weijie Wang, Haiyang Sun, Kun Zhan, Xianpeng Lang, Xiaowei Zhou, and Sida Peng. Street Gaussians: Modeling dynamic urban scenes with gaussian splatting. In *Computer Vision – ECCV 2024*, volume 15131 of *Lecture Notes in Computer Science*, pp. 156–173. Springer, 2025a. doi: 10.1007/978-3-031-73464-9_10. URL https://doi.org/10.1007/978-3-031-73464-9_10.

Yunzhi Yan, Zhen Xu, Haotong Lin, Haian Jin, Haoyu Guo, Yida Wang, Kun Zhan, Xianpeng Lang, Hujun Bao, Xiaowei Zhou, and Sida Peng. StreetCrafter: Street view synthesis with controllable video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 822–832, 2025b. doi: 10.1109/CVPR52734.2025.00085. URL https://openaccess.thecvf.com/content/CVPR2025/html/Yan_StreetCrafter_Street_View_Synthesis_with_Controllable_Video_Diffusion_Models_CVPR_2025_paper.html.

Chen Yang, Sikuang Li, Jiemin Fang, Ruofan Liang, Lingxi Xie, Xiaopeng Zhang, Wei Shen, and Qi Tian. GaussianObject: High-quality 3D object reconstruction from four views with gaussian splatting. *ACM Transactions on Graphics*, 43(6):1–13, 2024a. doi: 10.1145/3687759. URL https://doi.org/10.1145/3687759.

Jiawei Yang, Boris Ivanovic, Or Litany, Xinshuo Weng, Seung Wook Kim, Boyi Li, Tong Che, Danfei Xu, Sanja Fidler, Marco Pavone, and Yue Wang. EmerNeRF: Emergent spatial-temporal scene decomposition via self-supervision. In *International Conference on Learning Representations (ICLR)*, pp. 16739–16766, 2024b. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/47fc64d05a394955b1ae2487bfef1ab0-Abstract-Conference.html.

Ze Yang, Yun Chen, Jingkang Wang, Sivabalan Manivasagam, Wei-Chiu Ma, Anqi Joyce Yang, and Raquel Urtasun. UniSim: A Neural Closed-Loop Sensor Simulator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1389–1399, June 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Yang_UniSim_A_Neural_Closed-Loop_Sensor_Simulator_CVPR_2023_paper.html.

Zesong Yang, Bangbang Yang, Wenqi Dong, Chenxuan Cao, Liyuan Cui, Yuewen Ma, Zhaopeng Cui, and Hujun Bao. InstaScene: Towards complete 3D instance decomposition and reconstruction from cluttered scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7771–7781, 2025a. URL https://openaccess.thecvf.com/content/ICCV2025/html/Yang_InstaScene_Towards_Complete_3D_Instance_Decomposition_and_Reconstruction_from_Cluttered_ICCV_2025_paper.html.

Zeyu Yang, Zijie Pan, Yuankun Yang, Xiatian Zhu, and Li Zhang. Driving view synthesis on free-form trajectories with generative prior. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 28083–28092, 2025b. doi: 10.1109/ICCV51701.2025.02607. URL https://openaccess.thecvf.com/content/ICCV2025/html/Yang_Driving_View_Synthesis_on_Free-form_Trajectories_with_Generative_Prior_ICCV_2025_paper.html.

Zhongrui Yu, Haoran Wang, Jinze Yang, Hanzhang Wang, Jiale Cao, Zhong Ji, and Mingming Sun. SGD: Street view synthesis with gaussian splatting and diffusion prior. In *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, pp. 3812–3822, 2025. URL https://openaccess.thecvf.com/content/WACV2025/html/Yu_SGD_Street_View_Synthesis_with_Gaussian_Splatting_and_Diffusion_Prior_WACV_2025_paper.html.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 586–595, 2018. doi: 10.1109/CVPR.2018.00068. URL https://openaccess.thecvf.com/content_cvpr_2018/html/Zhang_The_Unreasonable_Effectiveness_CVPR_2018_paper.html.

Guosheng Zhao, Chaojun Ni, Xiaofeng Wang, Zheng Zhu, Xueyang Zhang, Yida Wang, Guan Huang, Xinze Chen, Boyuan Wang, Youyi Zhang, Wenjun Mei, and Xingang Wang. DriveDreamer4D: World models are effective data machines for 4D driving scene representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12015–12026, 2025a. doi: 10.1109/CVPR52734.2025.01122. URL https://openaccess.thecvf.com/content/CVPR2025/html/Zhao_DriveDreamer4D_World_Models_Are_Effective_Data_Machines_for_4D_Driving_CVPR_2025_paper.html.

Guosheng Zhao, Xiaofeng Wang, Chaojun Ni, Zheng Zhu, Wenkang Qin, Guan Huang, and Xingang Wang. ReconDreamer++: Harmonizing generative and reconstructive models for driving scene representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 26718–26728, 2025b. doi: 10.1109/ICCV51701.2025.02480. URL https://openaccess.thecvf.com/content/ICCV2025/html/Zhao_ReconDreamer_Harmonizing_Generative_and_Reconstructive_Models_for_Driving_Scene_Representation_ICCV_2025_paper.html.

Xiaoyu Zhou, Zhiwei Lin, Xiaojun Shan, Yongtao Wang, Deqing Sun, and Ming-Hsuan 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)*, pp. 21634–21643, 2024. doi: 10.1109/CVPR52733.2024.02044. URL https://openaccess.thecvf.com/content/CVPR2024/html/Zhou_DrivingGaussian_Composite_Gaussian_Splatting_for_Surrounding_Dynamic_Autonomous_Driving_Scenes_CVPR_2024_paper.html.

A TASK QUERIES AND COMPLETION SETTING

This appendix develops the three steps by which FULLSPLAT turns later camera views and rigid-object edits into one reusable driving scene. Query exposure and observation support identify the required content (Appendices A–B). Complementary observations train object-canonical and world-fixed generators, whose outputs are jointly integrated into the scene (Appendices C–D). The resulting state supplies both novel-view renderings and edited videos through the same stored layers. Appendices E–H specify these evaluations and their component comparisons; Appendix I describes the state properties supporting subsequent reuse.

A.1 EVIDENCE, PLACEMENT, AND THE STORED SCENE

The evidence $\mathcal{O}_E$ contains calibrated images, camera poses, and available measured depth. Measurements determine observation support; the layered reconstruction $\mathcal{G}_0$ supplies the initial world and object layers. Existing geometry can remain weakly supported from directions absent in the evidence.

We write τ for replay time and reserve $t \in [0, 1]$ for flow time. At replay time τ, the placement of canonical object i is

$$A_i(\tau)x = R_i(\tau)D_i x + t_i(\tau), \tag{S1}$$

where the positive diagonal matrix D_i gives the box normalization. Replay time selects the recorded placement; a query supplies an additional rigid assignment to the stored geometry.

The completed world layer $\mathcal{G}^*_{\mathrm{w}}$ and canonical layers $\{\mathcal{G}^*_i\}$ contain observed primitives and accepted residuals. Owner labels determine their motion; observed/generated labels identify their provenance. Refinement and later rendering retain both labels.

A.2 CAMERA AND ACTION QUERIES

A query $q = (p, a)$ combines a calibrated camera pose p with absolute object assignments $a = \{(Q_i, b_i)\}_i$. Here $Q_i \in SE(3)$ is an additional rigid transform, and $b_i \in \{0, 1\}$ controls whether the instance contributes to rendering. Identity uses $(I, 1)$, translation and rotation change Q_i, and removal sets $b_i = 0$. Assignments act on the recorded placement $A_i(\tau)$; they do not overwrite the canonical Gaussians.

The operating cameras follow the captured rig calibration in a road-aligned coordinate frame. The camera-offset construction uses $0, \pm 1, \pm 2, \pm 3$ m near the recorded route. A drivable-region test and an occupancy-based collision test retain valid poses. On curved roads, offsets follow the local road frame rather than a fixed global axis. The 4–6 m sweep uses the same stored scene and is described in Appendix E.4; its queries do not enter the completion distribution.

Translation follows a valid path in the local support region, and rotation is defined about the object's support point. Validity is checked against the road or support region, protected static geometry, and other active objects. Vehicles and roadside rigid objects use the same three operation types. For vehicles, translation and rotation adopt the lane-change and direction-change categories of HorizonSuite (Wang et al., 2026a). The action list determines displacements, rotations, and interpolation paths shared by the compared methods.

These operations produce complementary completion needs. A rotation can expose a weakly observed side of an object, while translation can expose both object surfaces and previously occluded

background. Removal requests the latter without requiring the removed object's surface to be rendered. A camera change can expose either type. Their common computational role is to identify content that must be available when the stored scene is queried.

A.3 Query distribution and finite sampling

Let h index a camera-stratum and action-family pair. A hierarchical query distribution can be written as

$$p_F(q) = \sum_{h=1}^{H} \pi_h p_h(q), \qquad \pi_h \geq 0, \qquad \sum_{h=1}^{H} \pi_h = 1, \tag{S2}$$

where p_h is supported on valid queries in its stratum. The stratum weight describes how that part of the task contributes to completion. Geometric validity is applied before constructing demand, and every positive-weight stratum is represented by valid samples.

Completion queries $\mathcal{Q}_F$ construct exposure, generation conditions, and joint constraints. Independently sampled queries $\mathcal{Q}_{\text{eval}}$ evaluate the stored state in the same domain, without selecting its seed or targets.

If stratum h supplies $K_h > 0$ equally weighted samples from p_h, the corresponding quadrature weights are $w_k = \pi_h / K_h$. Direct sampling from the mixture instead uses the usual equal-sample average. Projected area, visibility, and ownership confidence enter separately through $\alpha_q(c)$.

Evidence and completion queries fix support, target identities, and demand weights. Later queries render the resulting stored completion.

B Observation support and completion demand

Target construction connects the intended use of the scene to the two generators. Support describes what the captured evidence already constrains; exposure describes what a later query may reveal. Their combination specifies where additional content is useful. Coverage then measures whether the integrated scene supplies that content, while observation checks retain its agreement with supported rays.

B.1 Directional object support and world support

Object evidence is transformed into the instance's normalized canonical frame. Support $S_i^{\text{obs}}(c, \omega)$ is associated with a grid cell and an observation direction. Evidence contributions account for mask confidence, transmittance at the first object hit, track confidence, and depth consistency. Nearly repeated camera rays are downweighted when accumulating angular evidence. Consequently, many observations of the front can strengthen front-facing support without assigning the same support to the back.

Directionality matters even when a coarse primitive overlaps several surface regions. The Gaussian representation may extend across a cell boundary, but the measurements that support it remain associated with particular rays. Target construction reads the support field, rather than treating overlap with an existing Gaussian as complete surface evidence. Measured free space in front of a surface is recorded separately from support for the surface itself.

World support $S_{\text{w}}^{\text{obs}}(c)$ combines static surface hits, available occupancy measurements, measured free space, and agreement across views. A well-observed world region can be reused when an edit exposes it. A low-support region may require generation, with measured free space restricting where occupied geometry can be added. An unobserved cell and a cell traversed by a measured free-space ray thus provide different conditions to the background branch.

Both fields are constructed from $\mathcal{O}_E$ before generation. Generated candidates can explain a target in the coverage calculation, but they do not increase its observation support. This keeps the roles of conditioning evidence and inferred content consistent throughout completion.

B.2 Object exposure and background reveal intervals

The pre-completion object proxy combines observed geometry, the tracked box, and coarse category shape information. It provides an exposure envelope for ray tests. For a valid query, first intersections with this proxy identify the object-grid cells $V_i(q)$. Viewing directions are expressed in the canonical frame. Projected footprint tests remove sub-pixel requests at the rendering resolution, while directional support determines whether the exposed region is already supported.

The proxy predicts where an object surface may be needed before the missing surface has been generated. The generator still infers occupancy and geometry within the requested region. A proxy intersection is therefore a target-location constraint, not a geometric supervision target. This separation allows coarse boxes and category information to guide exposure without being treated as measured object surfaces.

Background demand is constructed from rays whose occlusion changes under a camera or object query. For each such ray u, the previously occluding region gives a near endpoint, and the next reliable geometric constraint gives a far endpoint. Write the valid interval as

$$I_{q,u} = [z^-_{q,u}, z^+_{q,u}], \qquad z^+_{q,u} > z^-_{q,u} + \epsilon_z. \tag{S3}$$

Available constraints include reliable world hits, road intersections, and measured free-space or local support information behind the occluder. Intervals without a valid ordered endpoint pair do not contribute targets. World-grid cells intersecting valid intervals form $J(q)$.

Several moving objects can reveal the same world region. Intervals are merged in world coordinates before their demand is accumulated. Within one query, a world cell has one world identity even if several occluders expose it. Across queries, exposure is accumulated according to the query weights. This avoids duplicate background ownership while retaining the greater task relevance of a region needed in many queries.

B.3 Discrete demand and typed target storage

For a finite completion sample, the expectations in Equation 2 have the discrete form

$$\widehat{E}^{\mathrm{obj}}_i(c) = \sum_{k=1}^{K} w_k \big(1 - S^{\mathrm{obs}}_i(c, \omega_{q_k})\big) \alpha_{q_k}(c) \mathbf{1}\{c \in V_i(q_k)\}, \tag{S4}$$

$$\widehat{E}^{\mathrm{bg}}_{\mathrm{w}}(c) = \sum_{k=1}^{K} w_k \big(1 - S^{\mathrm{obs}}_{\mathrm{w}}(c)\big) \alpha_{q_k}(c) \mathbf{1}\{c \in J(q_k)\}. \tag{S5}$$

The weights follow Appendix A. Each summand has three distinct factors: missing observation support, contribution within a query, and exposure of the cell by that query. A low-support cell with no exposure contributes zero demand. Conversely, exposure of a supported cell can be served by the existing layer.

Directional support and query records retain the information needed to check coverage. Demand conditions generation; occupancy is inferred within the requested domain.

The target set contains low-support regions identified as exposed by the proxy and reveal tests in the finite query pool. Object target records include an instance identity and canonical cell index; background records include a world cell index. The typed set can be expressed as

$$\mathcal{C} = \big(\{\mathrm{w}\} \times \mathcal{C}_{\mathrm{w}}\big) \sqcup \bigsqcup_i \big(\{i\} \times \mathcal{C}_i\big), \tag{S6}$$

where tags distinguish cells in different coordinate frames. Identical integer indices in two grids do not identify the same spatial region. The tags survive generation, resolution, acceptance, and storage.

All eligible low-support regions exposed in the finite query pool enter the typed requirement, including regions needed by a single query. The two demand fields condition generation in the corresponding frames, and the fixed target set subsequently evaluates the integrated scene.

Combining camera and action requirements. The finite sum also explains how different uses contribute to a common completion. Let $f(c, q)$ denote the appropriate summand before its query weight: either the object expression with directional support or the world expression with merged reveal membership. For stratum h, write its empirical contribution as

$$\widehat{E}_h(c) = \frac{1}{K_h} \sum_{k:q_k \in h} f(c, q_k), \qquad \widehat{E}(c) = \sum_h \pi_h \widehat{E}_h(c). \tag{S7}$$

A camera-motion stratum and an object-rotation stratum can request different parts of the same canonical instance. Their contributions are accumulated into one field for that owner. Likewise, translation and removal can contribute to the demand for the same world region. The sum combines their task relevance without creating separate stored copies for the two operations.

For example, a camera shift and an object rotation may both request the same vehicle side. Once completed, that side belongs to the instance and can serve either operation. Similarly, a road region requested by translation and removal is stored once in the world layer. Merging intervals within a query establishes this unique world identity; summing demand across queries expresses how often the region matters. The target representation therefore already encodes the reuse required of the final scene.

B.4 Coverage of a fixed completion requirement

Coverage evaluates whether the stored scene supplies requested content when the corresponding query is rendered. For target c, let $o(c)$ denote its declared owner. Its single-query indicator is

$$\begin{aligned} C_{\mathcal{W}}(c, q) = \mathbf{1}\{&\text{first owner} = o(c)\} \\ &\cdot \mathbf{1}\{\text{hit lies in the target region}\} \\ &\cdot \mathbf{1}\{\text{target-owned opacity} \geq \tau_\alpha\}. \end{aligned} \tag{S8}$$

The allowed region is the object-proxy surface band or background reveal interval. The opacity condition concerns content of the declared owner. A different layer becoming opaque in front of a target does not satisfy that target's coverage condition.

Let $\beta_{c,k} \geq 0$ be the fixed exposure weight of query q_k for target c, positive only for queries that expose it. For a target with positive total exposure,

$$C_{\mathcal{W}}(c) = \frac{\sum_k \beta_{c,k} C_{\mathcal{W}}(c, q_k)}{\sum_k \beta_{c,k}} \in [0, 1]. \tag{S9}$$

The notation β denotes pre-generation exposure weights, not an additional learned mechanism. All checks render the same stored geometry, transforming objects according to the query and keeping background in world coordinates.

With $E(c)$ denoting object or world demand, residual demand is

$$r(\mathcal{W}) = \frac{\sum_{c \in \mathcal{C}} E(c)(1 - C_{\mathcal{W}}(c))}{\sum_{c \in \mathcal{C}} E(c)}. \tag{S10}$$

The empty-demand case uses $r = 0$. For positive total demand, $0 \leq r \leq 1$, since weights and uncovered fractions are nonnegative. Equivalently, $1 - r$ is demand-weighted coverage. The condition $r \leq 1 - \rho$ therefore requires the stored scene to account for at least a fraction ρ of the weighted demand. This criterion is evaluated on the fixed completion sample $\mathcal{Q}_F$. Separately sampled evaluation queries measure how the stored completion serves other uses within the declared domain.

The denominator and target identities remain fixed during refinement. Removing redundant candidates can simplify the representation without changing the requirement. Refinement can improve coverage indicators, but it cannot improve the ratio by deleting an uncovered target. The final decision consequently refers to the same task that produced the generation conditions.

B.5 Cumulative observation-constrained acceptance

Support and measurement checks determine whether a candidate can be added at its proposed location. For object candidate g, let $\Omega_E(g)$ contain evidence-direction bins whose rays intersect its

three-standard-deviation ellipsoid, and let $c(g)$ denote its associated support-grid cell. Its gate support is

$$\bar{S}_i^{\mathrm{obs}}(g) = \max_{\omega \in \Omega_E(g)} S_i^{\mathrm{obs}}(c(g), \omega), \tag{S11}$$

with zero for an empty set; world candidates use scalar world support. The indicator $M_{\mathrm{meas}}(g)$ tests candidate extent against measured occupancy and free space. These tests retain their measurement-based role independently of the learned resolver score η_g.

The rendering check evaluates all accepted additions together. Let $\mathcal{R}_E$ be the fixed supported-ray set, and let $\mathcal{K}(r)$ contain valid discrepancy components for ray r. For positive, validation-fixed scales σ_k, define

$$J_{\mathrm{obs}}(\mathcal{G}; \mathcal{G}_0) = \max_{r \in \mathcal{R}_E} \max_{k \in \mathcal{K}(r)} \frac{d_k(\mathcal{G}, \mathcal{G}_0; r)}{\sigma_k}. \tag{S12}$$

Components compare RGB, depth, transmittance, and first-hit ownership. Depth is included only where evidence makes it valid; first-owner change is an indicator. The scales put different components on the declared common tolerance scale. The ray set and valid components are determined before candidates are evaluated. If there are no valid supported-ray components, we use $J_{\mathrm{obs}} = 0$; the other acceptance conditions still apply.

Candidates are processed in decreasing resolver score, with owner, cell index, and persistent candidate ID breaking ties. If $\mathcal{G}^{(m)} = \mathcal{G}_0 \cup \Delta\mathcal{G}_{\mathrm{acc}}^{(m)}$ is the accepted scene, the candidate test is applied to

$$\mathcal{G}^{+} = \mathcal{G}^{(m)} \cup \{g\}. \tag{S13}$$

The support, measurement-conflict, resolver-score, and cumulative observation conditions in Equation 4 must all pass. The render uses captured poses and recorded placements before deliberate edits are applied.

Coverage and observation agreement. Every accepted scene passes the check against the same base $\mathcal{G}_0$; rejection leaves that scene unchanged. Refinement also rechecks the full composition and rolls back failing updates. Testing all additions together accounts for their accumulated effect along a ray. By Equation S12, acceptance gives

$$d_k(\mathcal{G}, \mathcal{G}_0; r) \le \epsilon_{\mathrm{obs}} \sigma_k \quad (r \in \mathcal{R}_E,\ k \in \mathcal{K}(r)) \tag{S14}$$

whenever the gate passes.

The coverage criterion asks whether the new scene supplies the requested surfaces; the observation criterion checks how those surfaces combine with existing content. A committed state satisfies both, with additions stored in their owners' frames. If coverage does not reach the threshold, the method returns the base-scene outcome.

C Supervision from real driving logs

The two branches learn from complementary observations of the same physical scene. Object supervision aligns observations by instance identity and canonical coordinates. Background supervision aligns them by world position. This mirrors the coordinate assignment at inference: the object branch learns missing parts of a persistent instance, while the world branch learns regions that become visible when an occluder moves. Waymo and nuScenes provide the driving logs used in the paper (Sun et al., 2020; Caesar et al., 2020).

C.1 Object partial-to-held-out examples

Tracks are filtered for rigid pose consistency, stable identity, sufficient projected extent, and calibration agreement. A moving vehicle uses the recorded placement at each time. A static roadside object uses a fixed world placement and a canonical frame derived from its box axes. Image crops, masks, rays, available depth, and base Gaussians are expressed relative to that same instance.

For a world-space observation y_j of object i at time τ_j, the corresponding canonical location is

$$x_j = D_i^{-1} R_i(\tau_j)^{\top} \big(y_j - t_i(\tau_j)\big). \tag{S15}$$

The transformation removes recorded placement, so observations from different times can constrain the same object cells. Their image rays still retain different viewing directions. Canonical alignment therefore does not turn front-facing evidence into back-facing evidence; it makes their spatial relationship explicit.

An evidence subset is selected to preserve at least one clear measured surface while controlling angular coverage. Complementary views provide targets for surfaces hidden from that subset. Target views are grouped by their angular distance from the evidence directions. A target pixel is valid when the instance is its depth-consistent first owner and the image boundary does not truncate the required observation. The target remains tied to the same physical instance.

Pseudo-targets for locations without direct depth are derived from screened full-log reconstructions. Their source labels are retained separately from measurement-backed targets. Category, object size, angular gap, evidence coverage, and motion status remain associated with the example. These attributes organize the training pairs and their sampling; they do not change the instance identity used by the canonical decoder.

The resulting pair has a common geometric reference throughout learning. The evidence subset supplies partial Gaussians, appearance, and measured rays; the complementary subset supplies valid targets for the hidden portion of that same instance. A target surface can therefore constrain canonical structure and, through its calibrated projection, the appearance, silhouette, and depth terms. These losses refer to one generated object layer. This matches its later use: query transforms change the placement of that layer while retaining the completed surface in the instance's coordinates.

C.2 Occluded-to-revealed background examples

For background supervision, an evidence object mask and calibrated rays identify an occluded volume. A complementary observation is mapped into the same world frame and accepted when static feature agreement, camera geometry, semantic class, and available depth support the correspondence. The target object must no longer own the first hit. Synchronous adjacent cameras provide one source of such observations; later frames require static-world agreement, and repeated traversals require alignment and illumination screening.

The training pair records more than an image crop. It associates the occluder identity, evidence and target cameras, first-hit depths, reveal mask, boundary ring, context ring, and validity mask. These records let the loss distinguish the newly supervised region from its already observed surroundings. Dynamic foreground and pixels without a valid finite static correspondence are excluded from background targets.

Let M_{reveal} select the revealed region and M_{valid} select valid correspondences. Their intersection defines the supervised reveal pixels:

$$M_{\mathrm{target}}(u) = M_{\mathrm{reveal}}(u) M_{\mathrm{valid}}(u). \tag{S16}$$

The reveal loss uses this intersection, while boundary supervision uses the ring around it. Context conditions generation even when it is not counted as a newly revealed target. The distinction prevents the large visible surroundings from overwhelming the supervision for the smaller hidden region.

The training correspondence is between two observations of a common world region. It teaches how visible context relates to a subsequently observed background surface. During completion, that learned relation is applied to the query-induced reveal intervals, with the current scene's own measured geometry and free space as conditions.

The occluding object's identity locates the reveal event, while the target remains attached to the world. The same road surface may be observed after different objects move or from another calibrated camera. These observations constrain a common background location. This is the supervision counterpart of merging reveal intervals: both operations associate the content with the surface to be stored, rather than with the particular action that exposed it.

C.3 SUPERVISED DOMAINS AND HIDDEN TARGETS

On the supervised latent domain, observed context and valid hidden targets have disjoint binary masks:

$$M_{\text{obs}} + M_{\text{hid}} = 1, \qquad M_{\text{obs}} \odot M_{\text{hid}} = 0. \tag{S17}$$

Here the domain consists of observed context together with hidden cells for which a measured or screened pseudo-target exists. An unsupported cell with no target is not assigned an empty-space label merely because the dataset does not observe it.

The supervision follows the information available at each location. Measured surface hits and screened geometry constrain occupancy and depth; measured empty segments constrain free space. Valid target latents supervise flow, and calibrated target pixels supervise rendered appearance. Boundary pixels connect a background reveal to its visible context. The masks route these signals to the relevant terms in Appendix D, allowing geometry and appearance to share a training example without assuming identical validity masks.

The evidence, target, and requested regions have different roles. Evidence conditions generation, complementary targets teach completion, and query demand identifies the region needed for the task. During learning, requested regions with valid complementary targets supply hidden supervision, conditioned on the evidence subset. At inference, the learned conditional model generates requested content from the current scene evidence. Object and background training thus follow the same partial-to-complete principle while preserving their distinct spatial correspondences.

C.4 DATA SPLITS AND SCENE-SPECIFIC FITTING

Locations and logs are split before examples are mined. Model learning and prior adaptation use training locations; validation fixes model selection and acceptance thresholds. Instance identities and views from the same location are kept within the same split. Test-scene conditioning and reconstruction use only the declared evidence $\mathcal{O}_E$.

Training-time complementary views provide supervision for learning the completion rule. At test time, the corresponding role is played by the learned model, not by newly accessed target images. Per-scene Gaussian fitting uses only the declared evidence $\mathcal{O}_E$, which includes the captured 0 m frames used for reconstruction evaluation. Captured-view reconstruction and virtual-view distributional evaluation are reported separately.

Calibration, tracks, and map annotations define common scene geometry and query validity. The data split determines access to image and depth evidence, while the query split determines which later renderings are used to build a completion and which are used to evaluate it. Keeping these two splits explicit makes the same supervision construction applicable to both object and world branches.

D DUAL-FRAME GENERATION AND SCENE INTEGRATION

Generation and integration operate on one typed completion requirement. The two branches first use the evidence and demand from Appendices B–C to propose content in their storage frames. The resolver then places these candidates into sampled queries, combines their interactions, and updates the shared parameters. Acceptance commits the resulting object and background layers together. This section follows that sequence from conditioning to the renderings supplied by the stored scene.

D.1 EVIDENCE TOKENS AND CANONICAL GEOMETRY

The condition encoder collects three kinds of information. Gaussian tokens describe the current representation: means, scales, rotations, opacity, radiance, owner, support, and observation labels. Image tokens carry appearance from calibrated object crops and world context. Geometry tokens carry rays, box dimensions, road directions, measured surface hits, and free-space segments. Coordinates are expressed in the receiving branch's frame before positional encoding.

The encoder provides shared evidence to the object and background branches, with frame and type information maintaining their roles. An object token is interpreted relative to a normalized instance. A world token refers to a fixed scene position. Cross-attention can use surrounding context without

assigning that context to the object's stored geometry. The distinction is preserved through the decoders, which produce candidates with fixed layer identities.

For a Gaussian with world mean μ_w and covariance Σ_w, canonicalization gives

$$\mu_o = D_i^{-1} R_i^\top (\mu_w - t_i), \tag{S18}$$

$$\Sigma_o = D_i^{-1} R_i^\top \Sigma_w R_i D_i^{-\top}. \tag{S19}$$

These expressions follow by applying the affine inverse to the Gaussian's first and second moments. The forward relation is

$$\mu_w = R_i D_i \mu_o + t_i, \qquad \Sigma_w = R_i D_i \Sigma_o D_i^\top R_i^\top. \tag{S20}$$

Thus scale normalization changes covariance as well as position. The resulting canonical residual can be stored once and placed at different recorded poses without re-estimating its geometry.

Frame conversion also determines the meaning of scene conditioning. Object dimensions connect the normalized shape to its physical size; calibrated rays connect image features to the partial surface. Road directions and nearby world geometry supply context for placement and boundaries. The encoder can therefore condition an object completion on its surroundings while the decoder continues to express every new primitive in that object's frame. For the background branch, the same measurements retain their world positions and describe the static surface around the reveal interval.

D.2 Object structure, latent generation, and decoding

The object branch adapts the structured 3D prior used in the main method (Xiang et al., 2025). Its input combines partial-instance evidence H_i with canonical demand E_i^{obj}. The structural stage identifies occupied cells within the requested domain. The latent stage produces geometry–appearance features for the occupied structure. The residual decoder then maps those features to Gaussian centers, covariances, opacity, and radiance.

The two stages address different aspects of an incomplete instance. Structure determines which part of the requested domain carries a surface. Geometry–appearance latents then describe how that surface is represented and rendered. Conditioning both stages on the partial instance connects the proposed shape to its observed extent and appearance. A completed side can subsequently be inspected from a shifted camera or exposed by rotating the instance, using the same canonical primitives in both cases.

Demand and occupancy serve different purposes. Demand identifies low-support object regions exposed by the sampled queries; occupancy specifies where the generative model places surfaces within the requested domain. A requested volume need not be filled uniformly. Observed partial geometry, the object's dimensions, image appearance, and scene context condition the structural decision and the attributes of the resulting primitives.

The pretrained backbone stays frozen. Partial-evidence adapters, cross-attention, conditional flow modules, and residual decoders provide the trainable connection between the partial driving-scene instance and the prior. Observed Gaussians remain anchors. The output is an addition to that instance's canonical layer, rather than a replacement for all of its reconstructed primitives.

A candidate retains its instance identity throughout this process. That identity is used to compute directional support, transform the candidate into each query, apply cross-layer constraints, and write an accepted update back to the same object layer. Scene conditioning therefore affects what is generated without changing which instance owns it.

D.3 World-grid background completion

The background branch represents the merged reveal intervals on a sparse world grid. Its conditions H_{w} combine measured world geometry and free space with image and road context; its demand $E_{\mathrm{w}}^{\mathrm{bg}}$ indicates which world locations later queries expose. A masked rectified-flow model generates hidden-cell latents (Liu et al., 2023), and a separate decoder predicts occupancy followed by Gaussian attributes.

The world frame is important for the meaning of the conditions. A road point revealed by two different object edits is represented at the same location in the shared world grid. Its context and

support are collected in the common grid, and merging reveal intervals happens before generation. The resulting residual is stored in the world layer and is reused when either occluder moves.

Road geometry and measured free space constrain geometric placement; images and semantic context condition appearance. These conditions have complementary roles. Appearance tokens can help continue a road surface across an occluded patch, while a measured free-space segment constrains where occupied geometry may exist along the ray. The candidate must satisfy both the generative proposal and the subsequent evidence checks.

The two branches use separate generative weights, latent variables, and decoders. They share scene evidence and later interact through joint constraints. This structure gives each branch a coordinate system suited to its content while keeping the final result tied to the same observed scene.

The background output has the same residual role as the object output. It supplies occupied surfaces inferred within the requested region, leaving measured empty space unoccupied and retaining visible world anchors. A completed road patch participates in all later renderings whose rays reach it. Its visibility follows the current camera and object configuration, while its geometry and appearance remain parameters of the common world layer.

D.4 Masked flow and fixed observed context

On a training domain with valid hidden target latents, Equation 5 defines the path

$$z_t = M_{\mathrm{obs}} z_{\mathrm{obs}} + M_{\mathrm{hid}}\big((1-t)\epsilon + t z_{\mathrm{hid}}\big), \qquad t \sim \mathcal{U}[0,1]. \tag{S21}$$

The masks and observed latent do not depend on flow time. Differentiating therefore gives

$$\frac{\mathrm{d}z_t}{\mathrm{d}t} = M_{\mathrm{hid}}(z_{\mathrm{hid}} - \epsilon). \tag{S22}$$

The target velocity is zero on the observed part. On hidden supervised cells, it is the difference between the target latent and its initial noise. The corresponding regression is

$$\mathcal{L}_{\mathrm{flow}} = \mathbb{E}\left\| M_{\mathrm{hid}}\big(v_\theta(z_t, t; H, E) - (z_{\mathrm{hid}} - \epsilon)\big)\right\|_2^2. \tag{S23}$$

Only cells with target latents enter this training loss, as specified in Appendix C. The same conditioning convention applies to the object and world branches, using their respective frames and parameters.

At inference, the active generation domain is partitioned into observed context and cells to generate. The latter need not have measured targets. To express the observed-value overwrite, define

$$\Pi_{\mathrm{obs}}(z) = M_{\mathrm{obs}} z_{\mathrm{obs}} + M_{\mathrm{hid}} z. \tag{S24}$$

After an integration step produces $\widetilde{z}_{n+1}$, the state used by the next step is $z_{n+1} = \Pi_{\mathrm{obs}}(\widetilde{z}_{n+1})$. This notation does not prescribe a particular numerical solver.

Binary, disjoint masks imply

$$M_{\mathrm{obs}} z_{n+1} = M_{\mathrm{obs}} z_{\mathrm{obs}}, \qquad \Pi_{\mathrm{obs}}(\Pi_{\mathrm{obs}}(z)) = \Pi_{\mathrm{obs}}(z). \tag{S25}$$

Thus observed context remains fixed while the conditional field evolves the hidden component. The field uses the partial scene and demand to predict a completion in the branch's latent space; decoding turns that prediction into explicit Gaussian candidates. Flow learning supplies the conditional generative prior, and the subsequent scene constraints determine how its proposals fit the current reconstruction.

Latent anchoring and rendered observation protection operate at different stages. The former fixes the input-supported part of the generative state. The latter checks the composite image after new primitives have been decoded and integrated. Both are used because added content can influence a ray even when the original primitives and their latent values have remained fixed.

D.5 Cross-layer constraints in a shared query frame

Each query maps an active object's candidates through $Q_i A_i$ into world coordinates. Background candidates are already in that frame. The resolver graph distinguishes observed Gaussians, object

candidates, and background candidates, retaining their fixed owners. It can therefore compare layer interactions while limiting updates to generated content.

Visibility-order edges connect candidates that compete along a ray. They relate first-hit ownership and depth ordering to the active object placements in each query. Geometric-exclusion edges act on intersecting candidate extents outside the allowed road-contact bands. They are active only for owners present in the queried configuration. This distinguishes unintended interpenetration from ordinary occlusion and contact with the supporting scene.

Visibility changes with the query; ownership persists. Consider a completed vehicle side and a background road patch behind it. In a query that keeps the vehicle in place, the object may own the first hit and occlude the road. Translation can reveal part of the road while exposing another part of the object; removal leaves only the world surface along those rays. These configurations impose different visibility relations on the same two stored layers. The resolver evaluates each relation under its corresponding query, including which objects are active and where they have been placed.

A background candidate can therefore be hidden in one query and visible in another without changing its owner or stored position. Likewise, different sides of an object become first hits under different cameras and rotations. Joint integration combines these query-dependent constraints on a common set of candidates. It seeks compatible surface placement and occlusion across the sampled uses, while retaining the object/world assignment that makes later edits meaningful.

The resolver combines agreement with evidence, same-layer structure, and cross-layer boundaries into η_g. Bounded refinement changes candidate geometry and attributes within the completion construction. An object candidate has one canonical parameterization, shared by all query renderings; it does not acquire a different stored geometry for each query. World candidates similarly receive one update in world coordinates.

All query renderings share the candidates' stored parameters. Demand and visibility weights combine their constraints, while observed Gaussians and owner labels remain fixed.

The refinement and acceptance stages have complementary roles. Refinement uses the shared constraints to adjust proposed geometry and attributes. The score η_g summarizes a candidate's agreement with evidence, local structure, and cross-layer boundaries. The independent observation check evaluates the accumulated rendering at captured poses. Coverage then measures whether the accepted composition meets the original query demand. Together, these stages connect the two generative proposals to a single scene-level outcome.

D.6 Position, covariance, and gradient writeback

Write the affine map for object i under query q as

$$y = Q_i A_i(x) = B_{i,q} x + b_{i,q}. \tag{S26}$$

If $U_{i,q}$ is the rotation in Q_i, then $B_{i,q} = U_{i,q} R_i D_i$. The positive scales make $B_{i,q}$ invertible. The canonical position corresponding to a world position is

$$x = B_{i,q}^{-1}(y - b_{i,q}). \tag{S27}$$

By contrast, the differential of a scalar query loss is

$$\mathrm{d}\mathcal{L}_q = (\nabla_y \mathcal{L}_q)^\top B_{i,q}\, \mathrm{d}x.$$

Identifying the coefficient of $\mathrm{d}x$ yields

$$\nabla_x \mathcal{L}_q = B_{i,q}^\top \nabla_y \mathcal{L}_q. \tag{S28}$$

The inverse writes a position into its storage frame; the transpose propagates a derivative to parameters in that frame. Because the placement contains scale, these operations cannot be interchanged.

For covariance, the corresponding pair is

$$\Sigma_y = B_{i,q} \Sigma_x B_{i,q}^\top, \qquad \Sigma_x = B_{i,q}^{-1} \Sigma_y B_{i,q}^{-\top}. \tag{S29}$$

For positive-definite Σ_x, this preserves the Gaussian's quadratic form:

$$(y - \mu_y)^\top \Sigma_y^{-1} (y - \mu_y) = (x - \mu_x)^\top \Sigma_x^{-1} (x - \mu_x). \tag{S30}$$

Thus the canonical and world descriptions refer to the same transformed ellipsoid.

For a differentiable covariance-dependent loss, the matrix differential gives the additional chain rule

$$\nabla_{\Sigma_x}\mathcal{L}_q = B_{i,q}^{\top}(\nabla_{\Sigma_y}\mathcal{L}_q)B_{i,q}. \tag{S31}$$

Any scale or rotation parameterization of covariance then applies its own chain rule to this matrix derivative. View-dependent radiance is rotated by the corresponding rigid rotation. The affine scale affects the geometry rather than acting as a spherical-harmonic rotation.

These relations let the shared world-space constraint system update object-canonical residuals consistently. Background residuals need no object-frame writeback. After refinement, both remain attached to the owners with which they entered the resolver.

Accumulating constraints on one stored candidate. Consider a canonical mean x_g rendered at $y_{g,q} = B_{i,q}x_g + b_{i,q}$ in several sampled queries. For a composite loss whose dependence on x_g is through these world-space means, the chain rule gives

$$\nabla_{x_g}\mathcal{L}_{\text{composite}} = \sum_{q\in\mathcal{Q}_F} B_{i,q}^{\top}\frac{\partial\mathcal{L}_{\text{composite}}}{\partial y_{g,q}}. \tag{S32}$$

Each partial derivative includes the query's objective weights and any differentiable visibility terms. Direct canonical-coordinate losses contribute their gradients separately.

Equation S32 gives one update to the stored variable x_g from its several query placements. A constraint on a shifted-camera rendering and a constraint on an edited-object rendering therefore act on the same canonical surface. Subsequent queries transform the updated mean through their own placements.

The argument applies to covariance through Equation S31, and to the background with the identity world-coordinate map. Thus both generators supply content in convenient storage frames, while the resolver evaluates their interactions in the common scene frame. Observation checks then act on the resulting composite rather than on independent branch outputs.

D.7 Training objectives and the role of each supervision term

The object objective groups latent, appearance, silhouette, geometry, and free-space supervision:

$$\begin{aligned}\mathcal{L}_{\text{obj}} = {} & \lambda_{\text{f}}\mathcal{L}^{\text{obj}}_{\text{flow}} + \lambda_{\text{r}}\mathcal{L}^{\text{obj}}_{\text{render}} + \lambda_{\text{m}}\mathcal{L}_{\text{mask}} + \lambda_{\text{d}}\mathcal{L}^{\text{obj}}_{\text{depth}} \\ & + \lambda_{\text{g}}\mathcal{L}^{\text{obj}}_{\text{geom}} + \lambda_{\text{free}}\mathcal{L}^{\text{obj}}_{\text{free}}.\end{aligned} \tag{S33}$$

RGB and silhouette terms use valid measured target pixels. Depth and geometry terms use locations supported by measurements or screened reconstruction targets. The free-space term penalizes occupancy in measured empty segments. The terms consequently supervise different attributes and domains, rather than treating every unknown cell as a fully labelled training example.

These terms connect the latent proposal to its rendered use. Flow supervision learns the hidden representation; appearance and silhouette supervision evaluate its projection; depth and geometry supervision constrain its placement. All refer to the canonical residual placed at the target observation's recorded pose. This couples the learned completion to the same coordinate transform used by later queries.

The background objective is

$$\begin{aligned}\mathcal{L}_{\text{bg}} = {} & \lambda_{\text{f}}\mathcal{L}^{\text{bg}}_{\text{flow}} + \lambda_{\text{r}}\mathcal{L}_{\text{reveal}} + \lambda_{\text{d}}\mathcal{L}^{\text{bg}}_{\text{depth}} + \lambda_{\text{s}}\mathcal{L}_{\text{sem}} \\ & + \lambda_{\text{b}}\mathcal{L}_{\text{boundary}} + \lambda_{\text{free}}\mathcal{L}^{\text{bg}}_{\text{free}}.\end{aligned} \tag{S34}$$

Reveal supervision concerns the pixels selected in Equation S16. Boundary supervision acts around this region, connecting its completion to visible context. Depth, semantics, and free space provide complementary information about the world surface and its allowed occupancy. These domains follow the real-log correspondences rather than arbitrary virtual-camera targets.

Joint training uses

$$\mathcal{L}_{\text{joint}} = \lambda_{\text{c}}\mathcal{L}_{\text{composite}} + \lambda_{\text{o}}\mathcal{L}_{\text{ownership}} + \lambda_{\text{n}}\mathcal{L}_{\text{nonoverlap}} + \lambda_{\text{z}}\mathcal{L}_{\text{order}} + \lambda_{\text{b}}\mathcal{L}_{\text{boundary}} + \lambda_{\text{free}}^{\text{joint}}\mathcal{L}_{\text{free}}^{\text{joint}}. \tag{S35}$$

Composite rendering ties the candidates to their combined image. Ownership and depth-order terms concern which layer explains a ray; non-overlap and free-space terms concern where occupied content may be placed. Their gradients act on generated candidates, using the coordinate rules above for object updates.

Two query partitions with the same scene seed also contribute $\mathcal{L}_{\text{partition}}$. Its comparison domain is the intersection of their typed demand supports. It aligns occupancy and type logits on those common cells. The intersection is important: a cell requested only by one partition does not supply a matching prediction from the other. Matching the seed controls stochastic variation while the shared-cell comparison addresses consistency of the predicted structure and ownership.

The complete objective remains

$$\mathcal{L} = \mathcal{L}_{\text{obj}} + \mathcal{L}_{\text{bg}} + \mathcal{L}_{\text{joint}} + \lambda_{\text{p}}\mathcal{L}_{\text{partition}}. \tag{S36}$$

This decomposition follows the stages of the method: learn each type of missing content, then constrain how the types jointly form a scene.

D.8 Training schedule and completion algorithm

The base reconstruction, ownership front end, and pretrained backbone are frozen. The object branch is trained on canonical partial-to-target examples, followed by the background branch on reveal events. The resolver and conditioning components are then trained with composite and cross-layer constraints. Adapters, cross-attention, flow modules, and decoders provide the trainable parameters specified in the main method. Training batches account for object category, angular gap, reveal source, and action strata. Loss weights and acceptance thresholds are selected using validation locations.

At completion time, the process has the following order:

1. Fix $\mathcal{O}_E$, $\mathcal{G}_0$, the query sample, and the scene seed. Compute support and the typed target set with its demand.
2. Generate object-canonical and world-fixed residual candidates conditioned on their evidence and demand fields.
3. Map active candidates into each sampled query, aggregate cross-layer constraints, and perform bounded generated-only refinement.
4. Process candidates in the fixed acceptance order, applying the support, measurement, resolver-score, and cumulative observation tests from Appendix B.
5. Recheck any subsequent refinement and evaluate coverage on the unchanged targets and weights. Commit only if the coverage threshold is met; otherwise return the base-scene outcome.
6. Partition accepted residuals by owner, store them in their declared frames, and expose only the query-rendering interface for later use.

A scene seed determines the sampled completion candidates before evaluation. Bounded refinement adjusts their geometry and attributes, and acceptance and coverage checks determine whether the resulting scene is committed. Once committed, the object and world layers are stored together; later queries neither resample a latent nor rerun the resolver. Appendix I makes the resulting state-update semantics explicit.

Implementation settings. We use a 64^3 canonical object grid (Xiang et al., 2025) and 0.10 m world-grid cells within the merged reveal intervals. Images have a 960-pixel long side, with aspect ratio preserved and camera intrinsics scaled accordingly. We sample four valid completion queries per camera-offset/action stratum, at most 112 per scene, and retain one sampled completion per scene.

The object, background, and joint stages run for 50,000, 50,000, and 20,000 updates, with effective batches of 16 object examples, 4 background examples, and 2 scenes, respectively. We use AdamW with peak learning rates of 10^{-4} for the branch stages and 5×10^{-5} for joint training. Losses are averaged over valid elements; flow, rendered-appearance, and composite-rendering terms have weight 1, and the remaining terms have weight 0.1.

Completion uses 50 uniform Euler steps per flow trajectory and at most 32 generated-only Adam refinement updates at learning rate 10^{-3}. We set $\tau_{\text{obs}} = 0.25$, $\tau_\alpha = 0.8$, $\rho = 0.9$, and $\epsilon_{\text{obs}} = 1$. The remaining acceptance settings are selected on validation scenes and fixed before evaluation.

D.9 From completed surfaces to query renderings

Camera shifts and rigid-object edits change which stored surfaces are visible. The completion targets are the low-support object and background regions identified as exposed by the query tests. Once integrated, completed surfaces are rendered according to each query's camera pose and object assignments. Novel-view synthesis and object editing therefore use the same stored completion.

The connection follows the compositing rule of Gaussian splatting (Kerbl et al., 2023). For front-to-back samples on a ray, let a_j be opacity, c_j color, and $T_j = \prod_{\ell<j}(1 - a_\ell)$ transmittance. The rendered color is

$$\widehat{C} = \sum_j T_j a_j c_j + T_{\text{end}} c_{\text{bg}}, \qquad T_{\text{end}} = \prod_j (1 - a_j). \tag{S37}$$

For a single inserted candidate g, hold the other samples fixed. Let T_{front} be transmittance up to its depth and C_{behind} the composite color behind that depth with unit incoming transmittance. Insertion changes the ray color by

$$\Delta C = T_{\text{front}} a_g \big(c_g - C_{\text{behind}}\big). \tag{S38}$$

The new surface contributes its own color and attenuates the content behind it under the compositing rule above. Cross-layer refinement coordinates the participating surfaces through their occupancy, boundaries, and depth order.

Object-canonical storage makes this contribution follow the edited instance. World storage keeps a revealed background surface in the same scene location. Multiple rendered views consequently read the same completed content, with visibility and view-dependent radiance determined by the query. The generative branches supply candidate surfaces; their placement and integration connect those candidates to rendering quality.

Captured, novel, and edited views are rendered from the same stored scene. The observation gate limits rendering changes on supported rays at captured poses. The main tables report improvements over the frozen OmniRe base in captured-view reconstruction, novel-view synthesis, and edited-video quality. Appendices E–H specify the evaluation protocols and ablations.

Accepted completions add geometry to query-exposed regions with low observation support. The added object surfaces follow their instances under rigid edits, while the added background surfaces remain fixed in world coordinates. Novel and edited views are rendered from these stored layers using the same Gaussian compositing rule.

E Evaluation definitions and aggregation

The evaluations in the main paper address two uses of the completed scene. Captured views compare the rendered scene with recorded images. Virtual views and edits test renderings for which the source logs do not provide pose- and edit-matched RGB targets. This section specifies the associated measurement and aggregation rules; edit-specific temporal sampling is given in Appendix G.

E.1 Captured-view reconstruction

The 0 m evaluation uses captured evidence frames at their original poses and times, with the original object assignments. Let $I(u), \widehat{I}(u) \in [0,1]^3$ be the recorded and rendered RGB values at pixel u,

and let M be the fixed valid-pixel mask. For a nonempty mask, the masked error is

$$\mathrm{MSE}_M = \frac{\sum_u M(u)\|\widehat{I}(u) - I(u)\|_2^2}{3\sum_u M(u)}, \qquad \mathrm{PSNR}_M = -10\log_{10}\mathrm{MSE}_M\,. \tag{S39}$$

The mask is determined independently of predictions and is shared by all methods. The reported reconstruction columns concern the full valid scene.

SSIM uses an 11×11 Gaussian window with standard deviation 1.5 and data range 1, averaged over RGB channels and valid locations. LPIPS uses the AlexNet-based version 0.1 with RGB inputs scaled to $[-1, 1]$ (Wang et al., 2004; Zhang et al., 2018). Frame scores are first averaged within each source log, then equally across logs. With $\mathcal{T}_s$ the evaluated frames of log s, this rule is

$$\bar{m} = \frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}}\frac{1}{|\mathcal{T}_s|}\sum_{j\in\mathcal{T}_s} m(\widehat{I}_{s,j}, I_{s,j}). \tag{S40}$$

Empty masks are handled identically for every method. Log-level averaging gives each source scene the same outer weight despite differences in its number of evaluated frames.

E.2 Novel-view feature pools and distributional scores

All compared methods render the same valid $+3$ m and -3 m queries. Real reference images come from captured frames in the corresponding test logs. The reference pool provides the real-image distribution; it is not paired to each virtual camera.

FID uses 2048-dimensional Inception-v3 pool features with the standard FID weights (Heusel et al., 2017). Reference and generated images use the same RGB conversion and 299×299 resizing, without method-specific crops or enhancement. The two signed offsets are pooled once per dataset for the main NVS score. Video evaluation uses the corresponding video feature extractor described in Appendix G.

For either feature map ϕ, let (μ_r, Σ_r) and (μ_g, Σ_g) be the empirical feature means and covariances of real and generated samples. The common Fréchet expression is

$$D_\phi = \|\mu_r - \mu_g\|_2^2 + \mathrm{tr}\Big(\Sigma_r + \Sigma_g - 2(\Sigma_r^{1/2}\Sigma_g\Sigma_r^{1/2})^{1/2}\Big)\,. \tag{S41}$$

Each score is computed from its full feature pool. FID is not averaged from per-frame scores, and FVD is not averaged from per-clip scores.

Pool membership fixes the samples entering both feature moments. The joint-edit score is therefore recomputed after combining the selected features, with the category membership and weights specified in Appendix G.

E.3 Traffic layout and frame quality

NTA-IoU follows the vehicle-detection and projected-box protocol of DriveDreamer4D (Zhao et al., 2025a). Annotated 3D boxes are projected at the query time and pose. A fixed YOLO11 detector provides image boxes, with the same center-based matching rule for every method. An unmatched annotated vehicle contributes zero IoU. NTL-IoU compares a fixed TwinLiteNet lane extractor with projected lane annotations on valid map-supported pixels. Both use

$$\mathrm{IoU}(P, T) = \frac{|P \cap T|}{|P \cup T|}, \tag{S42}$$

for the matched boxes or valid lane masks. Annotation validity, visibility, class conventions, and matching are shared. Scores are averaged within logs and then across logs, and multiplied by 100 in the tables. These NVS checks use the unedited query geometry.

MUSIQ and BRISQUE evaluate frame quality. MUSIQ uses the fixed KonIQ-10k quality-assessment checkpoint (Ke et al., 2021); BRISQUE uses the pretrained LIVE distortion model (Mittal et al., 2012). Both receive the same decoded frames at the common rendering resolution, followed only by their required internal preprocessing. Higher MUSIQ and lower BRISQUE are better. Their temporal and category aggregation is specified in Appendix G.

The metrics supply complementary observations about the stored scene: paired reconstruction measures agreement with recorded images, distributional scores concern rendered image or video features, and layout scores concern known traffic structure. The discussion of results in the main paper uses the appropriate comparison for each of these quantities.

E.4 Camera-offset evaluation

The Waymo offset evaluation uses $|d| \in \{0, 1, 2, 3, 4, 5, 6\}$ m with unedited object assignments. Every point renders the scene completed for the 3 m operating range. The 4–6 m queries change only the camera request; they do not enlarge the completion distribution or trigger another generation. The road-aligned construction and geometric validity tests follow Appendix A.

Each magnitude has its own rendered-image and rendered-video feature pool. FID uses the image extractor above, while FVD uses the fixed video evaluator. Real references are captured images and continuous clips from the corresponding source logs. Different offset magnitudes are not merged into one score. At 0 m, these feature-distribution scores remain distinct from the paired PSNR, SSIM, and LPIPS measurements.

Connection to the reported results. The observation gate measures change from $\mathcal{G}_0$ on supported rays; reconstruction metrics measure agreement with real images on the full valid mask. Relative to the OmniRe base, Table 1 reports PSNR gains of 4.83 dB on Waymo and 3.12 dB on nuScenes. Its NVS FID values decrease by 67.2% and 55.7%, respectively. In Table 2, FID decreases by 46.2%, 45.5%, and 47.3% for translation, rotation, and removal. These differences are computed directly from the main tables. Together, the two uses support the value of the stored completion; Appendix D.9 explains how its surfaces can contribute to both.

F Baseline adaptation and shared settings

F.1 Stored-scene reconstruction and novel views

The main reconstruction and NVS comparison evaluates stored 3D outputs. Street Gaussians uses the ECCV 2024 method (Yan et al., 2025a). OmniRe retains its native layered scene representation (Chen et al., 2025b); its row is the frozen base used by FullSplat before completion. StreetCrafter-G uses the Gaussian-distilled output of StreetCrafter (Yan et al., 2025b). ReconDreamer++ and FaithFusion provide their reconstructed Gaussian scenes after generative refinement (Zhao et al., 2025b; Wang et al., 2026b).

Each baseline retains its reconstruction procedure and stored-scene rendering interface. FullSplat and its ablations share the exact OmniRe base, isolating completion within that common starting state.

The comparison shares source images, available sensor evidence, evaluation times, calibration, rendering resolution, and valid query lists. Dataset adapters express tracks, camera parameters, masks, and map annotations in each method's supported format. They do not supply an extra image at a virtual target pose. Learned adaptation uses training logs; scene fitting uses the permitted evidence from that scene.

F.2 Method interfaces for edited videos

The edit comparison evaluates stored-scene renderings and diffusion-refined outputs. OmniRe and FullSplat apply rigid assignments to stored object layers. StreetCrafter uses its video diffusion stage with edited geometric conditioning. Difix (Wu et al., 2025) refines each frame rendered from the edited OmniRe scene. Both diffusion pipelines are marked with † in the main table.

All pipelines use the same cameras, instance identities, transforms, and activation flags. For Difix, edits enter through the rendered images; Difix does not update the stored Gaussians. Their outputs use the same evaluation pools and preprocessing. The NVS comparison concerns stored 3D outputs, while the edit comparison also includes image and video refinement.

F.3 Common settings and evaluation membership

Query geometry is fixed by Appendix A; evidence and data splits follow Appendix C. Metric preprocessing and aggregation follow Appendix E, with the temporal settings in Appendix G. Rendering resolution, query lists, reference membership, and clip sampling are common within each comparison. Pretrained model versions and any driving-data adaptation belong to the corresponding method configuration.

Acceptance thresholds are fixed on validation scenes. Rejected completions use the base rendering and remain in the preselected evaluation population. This gives a method one output for every selected scene and query, including scenes for which it does not commit additional content. It also keeps the full-model and ablation comparisons on the same source scenes.

G Rigid-edit queries and video evaluation

G.1 Absolute actions and continuous query sequences

The Waymo edit set covers vehicles and roadside rigid objects. Translation changes position along a valid path, rotation changes orientation about a support point, and removal deactivates the instance. For vehicles, the first two correspond to lane change and direction change in the adopted HorizonSuite categories (Wang et al., 2026a).

For a world-space point y, translation by δ and rotation $U \in SO(3)$ about support point c_i have the forms

$$Q^{\mathrm{trans}}(y) = y + \delta, \qquad Q^{\mathrm{rot}}(y) = c_i + U(y - c_i). \tag{S43}$$

These are additional transforms applied after the recorded placement. They specify the desired pose relative to that placement, rather than adding numerical deltas to already edited Gaussian coordinates.

For frame j, let τ_j be replay time, p_j the camera, and $\{(Q_{i,j}, b_{i,j})\}$ the assignments. A fixed completed state renders as

$$\begin{aligned} \mathcal{G}_j &= \mathcal{G}^*_{\mathrm{w}} \cup \bigcup_{i:b_{i,j}=1} Q_{i,j} A_i(\tau_j)(\mathcal{G}^*_i), \\ \widehat{I}_j &= \mathcal{R}(\mathcal{G}_j, p_j). \end{aligned} \tag{S44}$$

The world and canonical object layers are unchanged over the clip. Recorded placement, camera pose, and action variables determine each frame. An object surface generated before the sequence is therefore the same stored surface when it becomes visible later.

Camera paths and edit trajectories are continuous and checked for validity throughout the clip. Removal sets the object's active flag to zero for the whole clip. A clip is selected from its source geometry before any compared method renders. The completion sample and evaluation sequence remain distinct, as described in Appendix A.

G.2 Reference clips and feature pooling

Real reference clips come from the source logs of each edit category, with the same camera channels and temporal sampling. They provide real-video distributions for the edited outputs. Each method uses the same reference and query lists, including the base-scene outputs for rejected completions.

FID pools decoded output frames separately for translation, rotation, and removal. FVD uses a fixed I3D encoder pretrained on Kinetics-400 (Unterthiner et al., 2019). A clip contains L consecutive sampled frames at frame rate f_s; the same L, f_s, RGB normalization, and 224×224 evaluator preprocessing apply to real and generated clips.

Temporal sampling settings. With the configuration in Appendix D.8, we sample $L = 16$ frames at $f_s = 10$ Hz, spanning 1.5 s between the first and last samples. We use rigid translations of magnitude 1, 2, or 3 m and yaw rotations of 10°, 20°, or 30°, with both signs and geometric validity checks. A smooth interpolation $s(u) = 3u^2 - 2u^3$, $u = j/(L-1)$, $j = 0, \ldots, L-1$, takes translation

from zero to its selected displacement along the valid path and yaw from zero to its selected angle. Removal remains active throughout the clip.

Each category's FID/FVD is computed from its complete feature pool using Equation S41. The ablation table instead uses one balanced, preselected joint pool from all three edit categories. Features are combined before the score is computed. Real and generated pool membership is shared across methods within each comparison. Category scores and joint scores retain these different reference populations and are reported separately.

G.3 Frame quality and category aggregation

For method m, let I_{mskvf} be frame f of clip v from source scene s in category k. Let $\mathcal{S}_k$ be its source-scene set, $\mathcal{V}_{s,k}$ its clip set, and F_v the number of evaluated frames. A frame-level quality metric Q is aggregated as

$$\bar{Q}_{m,k} = \frac{1}{|\mathcal{S}_k|} \sum_{s \in \mathcal{S}_k} \frac{1}{|\mathcal{V}_{s,k}|} \sum_{v \in \mathcal{V}_{s,k}} \frac{1}{F_v} \sum_{f=1}^{F_v} Q(I_{mskvf}). \tag{S45}$$

The all-edit score gives the three categories equal weight:

$$\bar{Q}_m = \frac{1}{3} \sum_{k=1}^{3} \bar{Q}_{m,k}. \tag{S46}$$

A longer clip consequently does not receive more outer weight, and a category with more clips does not dominate the all-edit MUSIQ or BRISQUE score. This arithmetic aggregation is appropriate for the frame-level Q values; feature-distribution distances continue to use their pooled computation.

Together, the video and frame metrics evaluate continuous use of the completed layers along the fixed camera and edit trajectories.

H Ablation construction and interpretation

The four ablations in Table 3 correspond to the main steps of completion: constructing useful targets, generating object content, integrating layers, and preserving observations. They share source scenes, the frozen OmniRe base, permitted evidence, query domain, evaluation clips, and validation-fixed thresholds. Each variant stores its completed layers before evaluation and uses the same rigid rendering interface.

H.1 Uniform target sampling

Uniform target sampling replaces exposure-guided target construction with uniform sampling from eligible low-support object and background regions within the same scene bounds. It removes expected exposure from target selection and its spatial conditions. Instance ownership, coordinate assignment, evidence, geometric validity, generators, losses, resolver, and observation gate remain unchanged.

The evaluation camera distribution and edit queries remain fixed. The lower edited-view quality of this variant supports constructing targets from the content those queries expose. This comparison links the demand calculation in Appendix B to the usefulness of the resulting stored scene.

H.2 Replacing the object prior with 2D generation and fitting

The 2D-prior variant replaces the structured 3D object branch with image-conditioned, depth-guided 2D diffusion followed by Gaussian fitting. It follows the generation-and-distillation route of SGD (Yu et al., 2025), adapted here to object residuals. Reference images come from $\mathcal{O}_E$; depth conditions use the same observed geometry and target proxies available to the full method.

For completion-query views, generated object regions provide supervision for the added Gaussians. Known boxes and query transforms define the instance masks and canonical correspondence. The

fitting variables are the added canonical Gaussians; the observed object anchors remain frozen. This preserves the same destination representation while changing how its residual content is proposed.

The background branch remains unchanged. Object residuals from the replacement route pass through the resolver and observation checks and are stored canonically before evaluation. The resolver retains its architecture, losses, and training schedule with candidates from this variant. Any adaptation of the 2D prior uses training logs, and no evaluation view supplies diffusion or fitting targets.

The lower edited-view quality of this variant favors the structured 3D-prior route for object completion in the common scene pipeline. Both routes ultimately produce canonical residuals; the comparison concerns how that content is generated and fitted, with background completion and integration shared.

H.3 Removing cross-layer resolution

This variant retains both generators and their individual losses. It removes the resolver, its joint composite-rendering and cross-layer supervision, and the acceptance condition $\eta_g > \tau_{\text{accept}}$. Candidate layers are merged using their existing owner and coordinate tags, with the same world-cell deduplication. A fixed order by owner, cell index, and candidate ID replaces resolver-score sorting.

Support, measured-occupancy/free-space checks, and cumulative J_{obs} remain active. Thus the variant retains observation-based acceptance while removing the component that jointly refines the generated object and background candidates. The distinction is central: preserving observations and resolving relations among new layers are separate operations.

The lower edited-view quality supports the visual benefit of joint integration in addition to the individual generators and remaining checks. The shared-candidate construction in Appendix D explains how this component coordinates the object and background content used by each edit.

H.4 Removing the cumulative observation gate

Only the cumulative J_{obs} condition is disabled. The base remains frozen, and the support threshold, measured occupancy/free-space rejection, target bounds, and resolver remain active. This tests the effect of checking the combined rendering after adding content.

Equation S38 shows how additions change the contribution of content behind them even when the base parameters stay fixed. The captured-view degradation of this variant supports checking that accumulated composition. The gate complements the resolver by constraining the effect of accepted content on supported observations.

H.5 Reading the four comparisons together

The ablations connect different stages of one computation. The target control concerns which regions are requested; the prior replacement concerns how object content is constructed; the resolver control concerns how generated layers interact; the observation control concerns their combined effect on the captured scene. Their shared storage and rendering interface keeps the final task the same.

All-edit FID/FVD use the joint pool in Appendix G; MUSIQ uses its equal-scene, equal-category rule. Rejected completions remain in each variant's pool as base-scene outputs. The comparisons therefore assess each design choice through the same final rendering task.

I Properties of the stored scene state

The representation separates persistent content from query variables. After commit, let $\mathcal{W}^* = (\mathcal{G}^*_{\text{w}}, \{\mathcal{G}^*_i\}_i)$ denote the stored layers, and let $a = \{(Q_i, b_i)\}_i$ denote the current absolute assignments. Replay time τ selects the recorded placements. Subsequent operations update a, while rendering reads $\mathcal{W}^*, a, \tau$ and the camera.

I.1 World immutability and action locality

Let T_i^{ξ} assign (Q_i, b_i) the value $\xi = (Q_i', b_i')$. Its write set contains only that instance's query assignment. The world layer and every canonical object layer remain fixed. Translation and rotation therefore carry the observed and completed object surfaces together; removal exposes stored background at its unchanged world location. These are consequences of the storage and rendering interface in Equation S44.

I.2 Absolute assignment and independent-instance order

Absolute assignment is idempotent:

$$T_i^{\xi} \circ T_i^{\xi} = T_i^{\xi}. \tag{S47}$$

Repeating a requested pose leaves that pose unchanged. More generally, $T_i^{\zeta} \circ T_i^{\xi} = T_i^{\zeta}$: the latest assignment selects the configuration to render.

For distinct instances, the write sets are disjoint, so

$$T_i^{\xi} \circ T_j^{\zeta} = T_j^{\zeta} \circ T_i^{\xi}, \qquad i \neq j. \tag{S48}$$

Both orders produce the same per-instance assignments. This property applies to independent absolute assignments after commit, with valid intermediate configurations in both orders.

I.3 Restoration and coordinate round trips

Restoring the replay timestamp, camera, and absolute assignments restores the same rendering inputs: world layer, active instances, placements, and canonical contents. With $B = B_{i,q}$, the coordinate round trip preserves a mean and covariance:

$$\begin{aligned} B^{-1}\big((B\mu + b) - b\big) &= \mu, \\ B^{-1}(B\Sigma B^{\top})B^{-\top} &= \Sigma. \end{aligned} \tag{S49}$$

Positive canonical scales ensure the inverse exists. Radiance uses the corresponding inverse rigid rotation on a round trip.

Implementation checks compare the saved layers before and after queries, repeat absolute assignments, exchange independent-instance assignments, and restore a saved timestamp and configuration. A no-op render establishes the floating-point tolerance for render-level comparisons. These checks verify reuse of the stored content; the main-paper image and video evaluations assess the quality of its renderings.