StyledStreets: Multi-Style Street Simulator with Spatial and Temporal Consistency

Anonymous authors
Paper under double-blind review

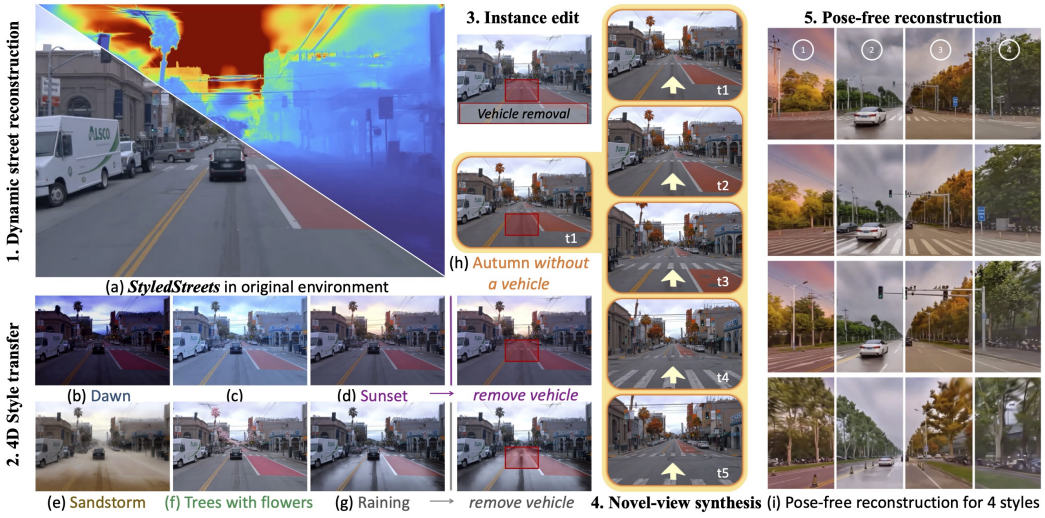

Figure 1: Our proposed *StyledStreets* migrates a high-fidelity 3DGS reconstruction for dynamic streets (a) into other styles (b–g), while the geometry and dynamic motions remain consistent with the original scene, preserving 3D simulation capability in the new styles for object editing (h). We also highlight multi-style reconstruction in a pose-free setup in (i).

Abstract

Urban street simulation requires more than realistic appearance transfer: it must preserve scene geometry, motion, and multi-camera consistency under changing appearance. We present *StyledStreets*, a multi-style 3D Gaussian street simulator for instruction-driven editing that combines physically grounded pose refinement, uncertainty-aware supervision, and unified stylization. Built on 3D Gaussian Splatting, the method addresses three coupled challenges: dynamic street reconstruction under rigid camera-vehicle constraints, noisy diffusion supervision across views and time, and geometry-preserving style transfer suitable for downstream simulation. In particular, it is designed to keep edited scenes useful not only for visual appearance control but also for downstream tasks such as vehicle tracking and ego-centric planning. Experiments on Waymo show stronger reconstruction quality than prior street-scene baselines, including an 18.2% lower Chamfer Distance than OmniRe, while edited scenes also improve tracking and planning robustness over diffusion-only alternatives. These results support controllable urban-scene simulation under diverse weather and appearance changes.

1 Introduction

Recent advances in neural rendering and street-scene reconstruction have improved geometric fidelity for dynamic urban scenes [7, 14, 19, 31], yet style-editing methods still struggle to preserve coherence across views and time [2, 29]. This gap matters for autonomous-driving simulation, where inconsistent edits can corrupt motion cues and cross-camera geometry used by downstream

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

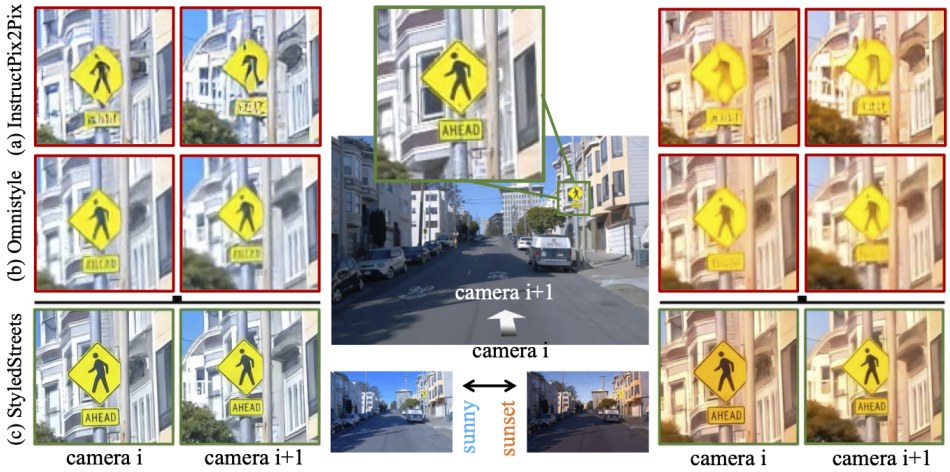

Figure 2: Leveraging InstructPix2Pix [2] (a) for appearance-consistent 3D reconstruction with StyledStreets (c) in novel styles, demonstrating geometrically consistent traffic sign preservation when sunny (left) and at sunset (right) compared with SoTA diffusion models such as Omnistyle [29] (b).

perception and planning systems [11, 12, 25]. Beyond visual quality alone, the core challenge is to preserve 3D structure strongly enough that edited scenes remain useful for tracking, planning, and broader simulation tasks.

We propose *StyledStreets*, a multi-style street simulator that combines dynamic 3D reconstruction with instruction-driven editing under explicit geometric and temporal constraints. Built on 3D Gaussian Splatting [14], it couples rigid camera-vehicle pose refinement, uncertainty-aware editing supervision, and unified style conditioning to preserve scene structure under appearance changes. Unlike diffusion-only pipelines [2, 29], it operates on dynamic street geometry directly, so edited scenes retain cross-view and temporal consistency and stay useful downstream. Figure 2 shows the motivation: 2D editing changes appearance but often fails to preserve the geometric consistency street-scene simulation requires. Our key contributions are:

1. **Physically grounded pose optimization** for dynamic street reconstruction, which refines vehicle motion under rigid camera-vehicle constraints instead of unconstrained per-camera pose updates. Its impact on reconstruction quality is validated in Tables 1 and 3.
2. **Uncertainty-aware instruction-driven editing** that filters inconsistent diffusion supervision and better preserves geometry during weather and object-level style edits. This behavior is evaluated in Table 4 and supported qualitatively by Figs. 6 and 7.
3. **Unified geometry-preserving stylization** that maintains stronger cross-view and temporal consistency under edited conditions. These effects are reflected in Table 6 and the ablation results in Table 9.
4. **Downstream validation for vehicle tracking and ego-centric planning**, showing that the edited scenes remain useful for temporal perception and decision-making. These benefits are validated in Tables 7 and 8.

2 Related Work

Neural Urban Scene Representations.

Recent urban 3D reconstruction methods based on Gaussian Splatting improve street-scale efficiency and geometric accuracy [6, 7, 14, 15, 20, 31, 32, 34]. More recent street multi-view reconstruction systems such as UniSim [33], NeuRAD [26], Desire-GS [20], MTGS [16], SimULi [28], and SplatAD [10] continue to strengthen rendering and simulation quality for driving scenes. These methods demonstrate strong rendering quality and dynamic-scene modeling, but they primarily optimize reconstruction rather than controllable appearance editing. As a result, they do not address the

problem of changing scene style while preserving motion, geometry, and multi-camera consistency. StyledStreets builds on this reconstruction line while extending it toward instruction-driven editing for dynamic street scenes.

Scene Editing & Stylization.

Text- and diffusion-driven scene editing methods can produce compelling appearance changes, but they often suffer from cross-view inconsistency and geometric drift [2, 5, 9, 22, 29]. Recent driving-scene editing systems such as DriveEditor [17], SceneCrafter [35], WeatherEdit [21], controllable weather video synthesis [18], and DrivingGaussian++ [30] further highlight demand for controllable street-scene manipulation. However, these methods still leave open the challenge of preserving multi-view geometry and temporal coherence under diverse edits. These observations motivate our use of explicit 3D Gaussian representations together with disentangled style and geometry updates.

Uncertainty in Neural Rendering.

Prior uncertainty modeling suggests that unreliable observations should be downweighted in dynamic reconstruction [1, 27]. Related works use uncertainty to handle occlusion, pose noise, or transient objects, which is closely related to the ambiguity introduced by diffusion-generated supervision. We adapt this idea to instruction-driven editing by learning reliability masks that suppress ambiguous edits while preserving stable scene structure across views and time.

3 Methodology

We build upon 3D Gaussian Splatting (3DGS) [14], representing scenes with anisotropic Gaussian primitives encoding position μ, covariance Σ, opacity α, and spherical harmonics (SH) coefficients for view-dependent appearance. Primitives are optimized via photometric loss:

$$\mathcal{L}_{\text{render}} = \sum_{t,m} \|I_t^m - \hat{I}_t^m\|_2^2, \tag{1}$$

where I_t^m and $\hat{I}_t^m$ denote rendered and ground-truth images for camera m at time t.

StyledStreets extends OmniRe [7]'s decomposition of static G_s (roads, buildings) and dynamic objects $\{O_k\}_{k=1}^K$ (vehicles, pedestrians), enhancing pose-tracker robustness, and pose-free reconstruction shown in Fig. 1 (i) too, while adding diffusion-based style transfer that mitigates spatio-temporal inconsistencies.

3.1 Dynamic Streets with Implicitly Optimized Poses

We first represent street scenes as $G = G_s \cup G_d$, $G_d = \bigcup_{k=1}^K (O_k \oplus \mathbf{T}_k^t)$ where G_s is static background, and G_d aggregates K dynamic objects O_k transformed by SE(3) poses $\mathbf{T}_k^t = (\mathbf{R}_k^t, \mathbf{t}_k^t)$. Each O_k contains local Gaussian parameters transformed to world space via:

$$\mu_i^{\text{world}} = \mathbf{R}_k^t \mu_i^{\text{local}} + \mathbf{t}_k^t, \quad \Sigma_i^{\text{world}} = \mathbf{R}_k^t \Sigma_i^{\text{local}} (\mathbf{R}_k^t)^\top \tag{2}$$

This decomposition enables efficient modeling while preserving motion consistency.

Instance Representation

Each O_k combines a deformable bounding box $B_k = (c_k, q_k, s_k)$ with object-centric Gaussians G_k. Local coordinates transform via $\mathbf{T}_k^t$, decoupling object motion from internal geometry for stable gradient flow through pose parameters.

Physically-grounded Pose Optimization

We optimize camera poses through physically-grounded decomposition: $\mathbf{T}_{\text{c}\to\text{w}}^{m,t}$ (camera-to-world transform for camera m at t) is expressed as

$$\mathbf{T}_{\text{c}\to\text{w}}^{m,t} = \mathbf{T}_{\text{v}\to\text{w}}^{t} \circ \mathbf{T}_{\text{c}\to\text{v}}^{m}, \text{ where the vehicle pose } \mathbf{T}_{\text{v}\to\text{w}}^{t} \in \text{SE(3)}\ , \tag{3}$$

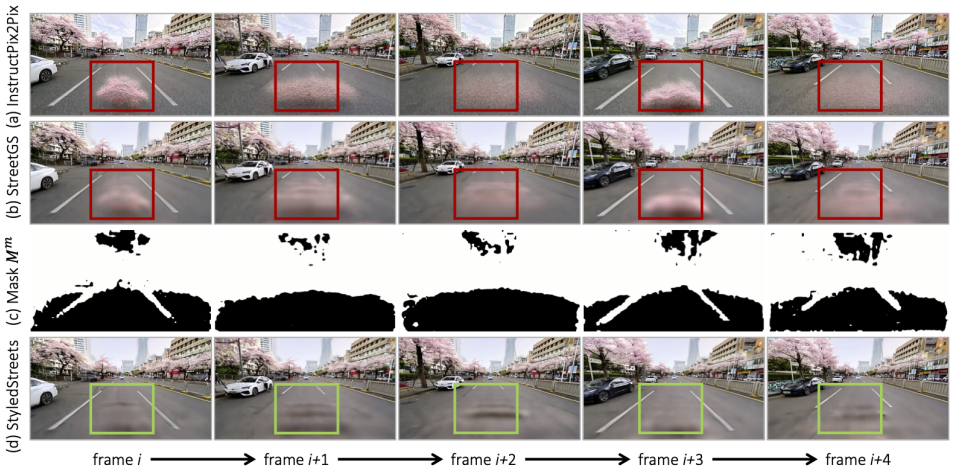

Figure 3: DINO-based ambiguity mitigation for rearview camera: (a) Temporally inconsistent InstructPix2Pix [2] supervision; (b) Reconstruction without masking; (c) Learned uncertainty mask M; (d) Our reconstruction using M preserves ground consistency.

and $\mathbf{T}^m_{\text{c}\to\text{v}} \in \text{SE(3)}$ are fixed camera-to-vehicle extrinsics (rigid mounting). Unlike OmniRe [7], which optimizes $\mathbf{T}^{m,t}_{\text{c}\to\text{w}}$ directly and can violate rigid mounting constraints, we freeze $\mathbf{T}^m_{\text{c}\to\text{v}}$ and optimize only $\mathbf{T}^t_{\text{v}\to\text{w}}$ via lightweight MLPs predicting SE(3) updates $\Delta\mathbf{T}^t_{\text{v}\to\text{w}}$. This preserves multi-camera geometry while ensuring temporal coherence through SE(3)-aware gradients; the resulting view-synthesis comparison is shown in Fig. 8 of the appendix.

Multi-view Stereo Training

Scenes are rendered as image I^m_t via the differentiable rasterizer $\mathcal{R}$ as $\mathcal{R}(G_s \cup G_d, \Pi^{m,t})$, with camera parameters $\Pi^{m,t} = \{\mathbf{K}, \mathbf{T}^t_{\text{v}\to\text{w}} \circ \mathbf{T}^m_{\text{c}\to\text{v}}\}$. The photometric loss

$$\mathcal{L}_{\text{render}} = \sum_{t,m} \underbrace{\|I^m_t - \hat{I}^m_t\|^2_2}_{\text{optimizes } \{G,\ \mathbf{T}^t_{\text{v}\to\text{w}}\}} \quad \text{with} \quad \mathbf{T}^m_{\text{c}\to\text{v}} \quad \text{frozen} \tag{4}$$

batches synchronized multi-camera frames to preserve rigid constraints on $\mathbf{T}^m_{\text{c}\to\text{v}}$ while aggregating gradients across views and timesteps. Fig. 1 (i) demonstrates multi-style scene reconstruction where $\mathbf{T}^i_{\text{v}\to\text{w}}$is learned implicitly without sensor extrinsics.

3.2 Instruction-driven Gaussian Editing

We enable photorealistic style transfer via instruction-guided Gaussian optimization. Given text instruction T (e.g., "heavy snow at night"), we adapt Gaussian primitives G through a joint loss function

$$\mathcal{L}_{\text{total}}(G(T)) = \mathcal{L}_{\text{style}} + \lambda_1 \mathcal{L}_{\text{reg}} + \lambda_2 \mathcal{L}_{\text{render}}\ , \tag{5}$$

where the style transfer loss

$$\mathcal{L}_{\text{style}} = \sum_{t,m} \|\mathcal{R}(G(T), \Pi^{m,t}) - \mathcal{D}_\theta(I^m_t, T)\|^2_2 \tag{6}$$

drives appearance alignment to diffusion-generated targets $\mathcal{D}_\theta(I^m_t, T)$. To avoid surface drifting from the original scene, a geometric preservation is enforced by

$$\mathcal{L}_{\text{reg}} = \sum_i \left(\|\mu_i(T) - \mu^0_i\|^2_2 + \|\Sigma_i(T) - \Sigma^0_i\|^2_F\right) \tag{7}$$

Original scene coherence is maintained through $\mathcal{L}_{\text{render}}$ from (4).

MITIGATING DIFFUSION AMBIGUITY

To address inconsistent 2D diffusion guidance as illustrated in Fig. 3 (a), we implicitly learn uncertainty maps σ^m for frame m initialized from DINOv2 features [4]. The reliability mask

$$M^m = \mathbb{I}\left(1/(2\sigma^{m^2}) > \tau\right), \quad \tau = 1/\sqrt{2} \tag{8}$$

identifies regions with stable multi-view and cross-frame correspondence. Such reliability mask is trained implicitly with our proposed masked style loss

$$\mathcal{L}_{\text{style}} = \sum_{t,m} \|M^m \odot (I_t^m(T) - \mathcal{D}_\theta(I_t^m, T))\|_2^2 , \tag{9}$$

which suppresses ambiguous supervision while preserving edits on persistent structures. As shown in Fig. 3, this yields better spatio-temporal consistency than applying diffusion supervision without masking.

The mask is what separates our supervision from a plain diffusion loss: by attenuating gradient on frames whose multi-view correspondence is unreliable, it keeps the residual signal concentrated on surfaces that actually persist across viewpoints. The effect is visible in the depth channel, where a per-style model drifts across the columns while the unified model holds the same road plane and vehicle silhouette regardless of weather.

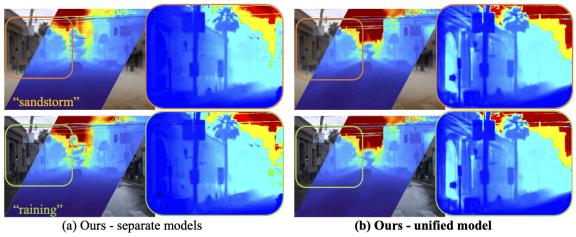

Figure 4: Depth consistency across stylistic edits: our unified model preserves geometric structure under different weather conditions (columns), while standalone models exhibit artifacts.

UNIFIED STYLIZATION MODEL

Our unified model stylizes scenes while preserving geometry through compact camera-aware transformations:

$$\begin{aligned} \mu_i(T) &= \mu_i^0 + \text{MLP}_\mu(E_i(T)), \\ \Sigma_i(T) &= \Sigma_i^0 \odot \exp(\text{MLP}_\Sigma(E_i(T))), \\ c_i(T) &= c_i^0 + \text{MLP}_c(E_i(T), \mu_i(T)). \end{aligned} \tag{10}$$

where $c_i(T)$ represents the appearance properties (spherical harmonics coefficients) for Gaussian i under style T. The composite embedding $E_i(T) = [\mathbf{e}_i^{\text{cam}}; \mathbf{e}^{\text{style}}(T)]$ combines camera parameters (learned encoding) and CLIP-based style codes [23]. Crucially, MLP_c jointly conditions on both the style embedding $E_i(T)$ and position $\mu_i(T)$ to model view-dependent effects.

This unified approach offers two key advantages over per-style training: 1. restricted gradient flow: MLP_μ and MLP_Σ update exclusively via $\mathcal{L}_{\text{render}}$, preserving geometry shown in Fig. 4 (b) 2. stable optimization <2% depth variation versus 8-12% in per-style models (Table 5)

The joint conditioning in MLP_c enables consistent appearance changes across viewpoints while maintaining geometric fidelity, as validated in Tables 3 and 6.

OPTIMIZATION

We minimize $\mathcal{L}_{\text{total}}$ through gradient updates to MLP parameters θ

$$\frac{\partial \mathcal{L}_{\text{total}}}{\partial \theta} = \frac{\partial \mathcal{L}_{\text{style}}}{\partial \theta} + \lambda_1 \frac{\partial \mathcal{L}_{\text{reg}}}{\partial \theta} + \lambda_2 \frac{\partial \mathcal{L}_{\text{render}}}{\partial \theta} \tag{11}$$

Fixed camera poses $\mathbf{T}_{\text{c}\to\text{v}}^m$ preserve multi-view consistency. This maintains spatial relationships (Fig. 4) while enabling 3D-consistent stylization (Fig. 5).



Figure 5: Three camera synthesis in 3 new styles (b,c,d) and the original style in (a). Environmental structures maintain shape fidelity while adopting new appearances.

4 Implementation

Scene Reconstruction

We initialize dynamic object poses $\mathbf{T}_k^t$ via Perspective-n-Point from detection tracks [24]. Vehicle trajectory $\{\mathbf{T}_{\text{v}\to\text{w}}^t\}$ is optimized while freezing $\mathbf{T}_{\text{c}\to\text{v}}^m$ gradients. Training batches sample three adjacent timesteps and cameras, with Gaussian primitives initialized from COLMAP sparse points following StreetGS [31]. The 4DGS component (Fig. 9 (b) of the appendix) handles complex deformations without skeletal priors.

Stylization Networks

Transformation MLPs use lightweight architectures: MLP_μ and MLP_Σ employ two linear layers with a latent feature dimension of 32 with ReLU activations, while MLP_c (appearance network) uses two linear layers with a default latent feature dimension of 64 with ReLU. MLP_Σ includes exponential output to ensure positive-definite covariances. Style codes $\mathbf{e}^{\text{style}}(T)$ derive from CLIP text embeddings [23], and uncertainty masks M^m are initialized from DINOv2 features [4].

Optimization

We minimize $\mathcal{L}_{\text{total}}$ using AdamW with learning rate 0.001. Hyperparameters are set to optimal values from our ablation study (Table 9): mask threshold $\tau = 1/\sqrt{2}$, $\lambda_1 = 0.1$ (geometric regularization), $\lambda_2 = 0.5$ (rendering consistency). These preserve geometric fidelity (<2% depth error) while enabling efficient stylization.

5 Experiments

5.1 Original Style Simulation

We follow OmniRe's experimental protocol [7] on the Waymo Open Dataset [25], using 3 front cameras for training. All methods are evaluated on vehicle reconstruction quality, excluding deformable objects per our problem scope. Metrics include PSNR↑ and SSIM↑ for both scene reconstruction (original views) and novel view synthesis.

View Synthesis

Our method improves vehicle PSNR by +2.24 dB over OmniRe while maintaining strong human reconstruction quality. OmniRe suffers from blurred geometries due to pose misalignment in Fig. 8 (c) of the appendix, whereas our joint pose optimization yields precise geometric alignment, evident in crisp details such as wheel spokes (panel (a)). The full model also outperforms both ablations on the human columns, indicating that pose refinement and multi-view training benefit deformable-object reconstruction. Table 2 further shows the best PandaSet PSNR among recent baselines, so the gains are not limited to Waymo.

Table 1: Comprehensive reconstruction and synthesis results on Waymo Open Dataset. Our method handles both rigid vehicles and non-rigid humans. Metrics: PSNR↑/SSIM↑ for scene reconstruction and novel-view synthesis.

	Scene Reconstruction						Novel View Synthesis					
	Full Image		Human		Vehicle		Full Image		Human		Vehicle	
Method	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
EmerNeRF [32]	31.93	0.902	22.88	0.578	24.65	0.723	29.67	0.883	20.32	0.454	22.07	0.609
3DGS [14]	26.00	0.912	–	–	16.18	0.425	25.57	0.906	–	–	16.00	0.407
DeformGS [34]	28.40	0.929	17.80	0.460	19.53	0.570	27.72	0.922	17.30	0.426	18.91	0.530
PVG [6]	32.37	0.937	24.06	0.703	25.02	0.787	30.19	0.919	21.30	0.567	22.28	0.679
HUGS [15]	28.26	0.923	16.23	0.404	24.31	0.794	27.65	0.914	15.99	0.378	23.27	0.748
StreetGS [31]	29.08	0.936	16.83	0.420	27.73	0.880	28.54	0.928	16.55	0.393	26.71	0.846
OmniRe [7]	34.25	0.954	28.15	0.845	28.91	0.892	32.57	0.942	24.36	0.727	27.57	0.858
Ours	**36.40**	**0.967**	**29.02**	**0.879**	**31.15**	**0.918**	**34.28**	**0.958**	**26.33**	**0.812**	**29.33**	**0.885**
w/o Pose Opt.	34.81	0.958	28.20	0.847	29.43	0.901	32.92	0.947	25.96	0.808	27.92	0.863
w/o Multi-View	34.63	0.956	28.18	0.846	29.25	0.897	32.75	0.941	25.90	0.806	27.75	0.859

Table 2: PandaSet view-synthesis comparison. Metric: PSNR↑.

Method	UniSim [33]	NeuRAD [26]	OmniRe [7]	DeSiReGS [20]	MTGS [16]	SplatAD [10]	SimULi [28]	**Ours**
PSNR↑	23.53	26.50	26.68	25.17	26.08	28.58	29.76	**30.22**

Geometric Reconstruction

Our approach achieves superior geometric reconstruction with 18.2% lower Chamfer Distance than OmniRe (0.198 vs. 0.242), validated by LiDAR. The pose optimization and multi-view ablations remain clearly better than prior reconstruction baselines.

5.2 Instruction-driven Style Transform

Our framework achieves photorealistic style transfer while preserving geometry across views and time. As shown in Figs. 5 and 6, multi-camera renders remain coherent under large appearance changes, while the regularization term in Eq. 7 helps maintain stable scene structure. Extrapolating to a viewpoint the capture never observed is the stricter test, because there is no recorded image for the diffusion guidance to fall back on: the style must be carried entirely by the scene representation, and any geometric drift in the underlying Gaussians becomes visible as warped lane markings or a distorted vehicle profile. The sandstorm and snow columns show that the road plane and vehicle proportions survive that extrapolation, which we take as evidence that the appearance change is applied to the reconstruction rather than painted over it.

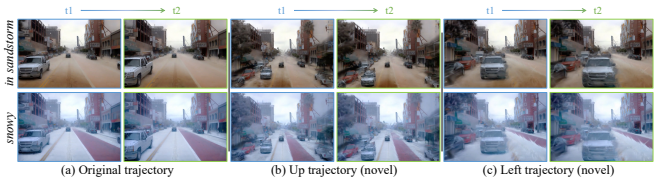

Figure 6: Realistic novel-view extrapolation (b,c) in extreme styles (sandstorm and snow) while preserving road topology and vehicle proportions as presented in (a).

To measure editing quality more directly, Table 4 reports instruction alignment together with temporal and cross-view consistency for weather editing, object insertion, and object removal. The method performs best on both weather editing and object insertion while also achieving the lowest LPIPS for object removal.

Quantitative results in Table 6 show that the unified model preserves geometry more effectively than per-style training and remains robust under the main ablations. Combined with Fig. 4, these results support the role of unified conditioning and uncertainty-aware supervision in reducing geometry drift across styles.

This challenging case highlights that coupled object insertion and weather manipu-



Table 3: LiDAR depth accuracy. Our model reduces Chamfer Distance (CD) by 18.2% versus OmniRe [7] (0.198 vs. 0.242).

	Training		Novel	
Method	CD↓	RMSE↓	CD↓	RMSE↓
3DGS [14]	0.415	2.804	0.467	2.896
DeformGS [34]	0.384	2.965	0.383	2.990
StreetGS [31]	0.274	2.199	0.286	2.228
OmniRe [7]	0.242	1.894	0.244	1.909
Ours	**0.198**	**1.652**	**0.201**	**1.667**
w/o Pose Opt.	0.231	1.827	0.238	1.845
w/o Multi-View	0.225	1.802	0.232	1.823

Table 4: Instruction-following and temporal consistency. CLIP direction similarity S_{CLIP} and 3D/2D LPIPS.

	Weather		Obj. Ins.		Obj. Rem.
Method	S_{CLIP} ↑	LPIPS_{3D} ↓	S_{CLIP} ↑	LPIPS_{3D} ↓	LPIPS_{2D} ↓
InstructPix2Pix [2]	0.224	0.226	0.126	0.362	0.329
DriveEditor [17]	–	–	0.144	0.248	0.321
SceneCrafter [35]	0.229	0.196	0.109	0.320	0.318
WeatherEdit [21]	0.225	0.229	0.102	0.355	0.324
WeatherWeaver [18]	0.251	0.242	–	–	–
DrivingGaussian++ [30]	0.246	0.217	0.087	0.269	0.328
Ours (IP2P)	0.257	0.188	0.182	0.195	**0.317**
Ours ($\text{COSMOS}_{2.5}$)	**0.261**	**0.187**	**0.183**	**0.193**	0.317

Table 5: Per-style model performance when each style is trained separately. CD↓ and RMSE↓ for training and novel views.

Style	CD↓	RMSE↓	CD↓	RMSE↓
snowy road	0.212	1.745	0.221	1.768
sandstorm	0.225	1.812	0.233	1.841
sunset	0.205	1.703	0.211	1.721
rainy	0.218	1.782	0.227	1.805
midnight	0.208	1.715	0.215	1.732
sunny	0.200	1.668	0.206	1.689
spring	0.204	1.695	0.210	1.712

Table 6: Unified model performance compared with the average of the per-style models.

Models	CD↓	RMSE↓	CD↓	RMSE↓
Per-style (*avg.*)	0.210	1.731	0.217	1.752
Unified model	**0.201**	**1.674**	**0.206**	**1.693**
w/o Pose Opt.	0.215	1.752	0.222	1.778
w/o Multi-View	0.208	1.711	0.214	1.729
w/o Reg.	0.223	1.803	0.231	1.828
w/o DINO Mask	0.219	1.792	0.226	1.814

lation remain harder than appearance-only edits. Even so, the inserted car and its shadow remain more plausible and geometrically grounded than with 2D street-editing baselines, consistent with the trends in Table 4.

5.3 Tracking and Planning for Autonomous Driving

Tracking on nuScenes

We validate spatio-temporal consistency preservation using UniAD [11] on nuScenes [3]. Tracking performance depends strongly on consistent object geometries and motion patterns across frames. As shown in Table 7, 2D style-transfer methods show only marginal gains, whereas StyledStreets improves tracking metrics through stronger temporal coherence.

Table 7 demonstrates that 3D-consistent transfer improves temporal perception: compared with 2D style-transfer baselines, our method increases AMOTA and reduces both AMOTP and IDS, consistent with stronger geometric coherence across frames.

Planning on Bench2Drive

We validate practical utility using Bench2Drive [12], assessing success rate (SR) for violation-free route completion, driving score (DS) combining completion and infractions [8], efficiency (Eff) for speed maintenance, and comfortness (Com) against human standards [13]. To evaluate style augmentation, we transfer scene clips to rainy, cloudy, foggy, and sunny conditions, creating supplementary training sequences that preserve the original driving actions while introducing weather diversity.

Table 8 shows that the edited training data improves downstream planning robustness over 2D style-transfer baselines, with the strongest gains in driving score and success rate.

Table 7: Multi-object tracking on nuScenes validation. AMOTA/Recall (higher ↑), AMOTP/IDS (lower ↓).

Training data	AMOTA↑	AMOTP↓	Recall↑	IDS↓
nuScenes original	0.359	1.320	0.467	906
InstructPix2Pix [2]	0.355	1.328	0.463	915
InstructVid2Vid [22]	0.361	1.319	0.468	910
Omnistyle [29]	0.363	1.315	0.470	899
Ours (IP2P-tuned)	0.372	1.302	0.479	874
Ours (Omni-tuned)	**0.376**	**1.295**	**0.483**	**862**

Table 8: Closed-loop planning on Bench2Drive validation. DS: driving score [8], SR: success rate (%), Eff: efficiency, Com: comfortness [13].

Training data	DS↑	SR↑	Eff↑	Com↑
Bench2Drive original	42.4	15.0	157.9	46.0
InstructPix2Pix [2]	43.2	15.3	158.0	45.8
InstructVid2Vid [22]	44.3	15.7	158.0	45.7
Instruct-CLIP [5]	45.5	16.1	158.1	45.9
Omnistyle [29]	46.7	16.8	158.2	46.0
Ours (IP2P-tuned)	**48.9**	**18.7**	**158.6**	46.1
Ours (Omni-tuned)	48.6	18.4	158.5	**46.2**

5.4 Ablation Study

Table 9 shows that λ_1 and λ_2 are the most important terms for geometric stability, while mask threshold and latent dimension remain comparatively insensitive within the tested ranges. In particular, removing geometric regularization or scene-consistency supervision produces the largest degradation in both CD and RMSE, confirming that these two losses are the main drivers of stable stylization. By contrast, moderate changes in the mask threshold τ and the latent feature dimension have a much smaller effect, which suggests that the framework is relatively robust to those hyperparameters once the main geometric constraints are preserved.

Across all four groups the released setting is best and the row ordering is stable, so the gains do not depend on a lucky hyperparameter choice. The regularization weights dominate, while latent width matters only at the narrow end. We keep $\tau = 1/\sqrt{2}$, a 64-dimensional MLP_c, and $\lambda_1 = \lambda_2 = 0.1$ throughout.

Table 9: Ablation study on critical hyperparameters.

Variants	Training		Novel	
	CD↓	RMSE↓	CD↓	RMSE↓
Full Model	**0.198**	**1.652**	**0.206**	**1.693**
Mask threshold τ				
$\tau = 0.5$	0.205	1.685	0.211	1.723
$\tau = 1/\sqrt{2}$ (ours)	**0.198**	**1.652**	**0.206**	**1.693**
$\tau = 0.85$	0.202	1.662	0.207	1.701
Latent dim of MLP_c				
16-dim	0.208	1.712	0.214	1.745
64-dim (ours)	**0.198**	**1.652**	**0.206**	**1.693**
96-dim	0.199	1.658	0.207	1.699
Loss weight λ_1				
$\lambda_1 = 0.0$	0.241	1.912	0.253	1.945
$\lambda_1 = 0.1$ (ours)	**0.198**	**1.652**	**0.206**	**1.693**
$\lambda_1 = 0.5$	0.199	1.660	0.207	1.705
Loss weight λ_2				
$\lambda_2 = 0.0$	0.215	1.723	0.232	1.812
$\lambda_2 = 0.5$ (ours)	**0.198**	**1.652**	**0.206**	**1.693**
$\lambda_2 = 1.0$	0.200	1.665	0.215	1.732

6 Conclusion

StyledStreetscouples rigid camera-vehicle pose refinement, uncertainty-aware editing supervision, and unified style conditioning within a single 3D Gaussian representation, preserving scene structure more reliably than diffusion-only pipelines while remaining suitable for downstream simulation. On Waymo it improves reconstruction quality, editing consistency, and vehicle tracking and planning. A limitation is that styled edits rely on extra diffusion supervision and can struggle with coupled physical effects such as shadows.

Reproducibility Statement

Baseline reconstruction details (datasets, camera configuration, metrics) are given in Sec. 5.1; the diffusion-assisted multi-style tuning details (guidance model, uncertainty-aware supervision) are given in Sec. 3.2 and Sec. 4.

AI Use Statement

Generative AI tools assisted with drafting and polishing portions of the manuscript, including language editing, organization, and LaTeX formatting. All AI-assisted text was reviewed and revised by the authors, who verified the technical claims, experimental setup, implementations, numerical results, and conclusions, and take full responsibility for the final content.

REFERENCES

[1] Jonathan Ackerman, Xie Xie, et al. Wildgaussians: 3d gaussian splatting in the wild. *arXiv preprint arXiv:2407.08447*, 2024.

[2] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*, 2022. URL https://arxiv.org/pdf/2211.09800.

[3] 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. *arXiv preprint arXiv:1903.11027*, 2019.

[4] Mathilde Caron, Hugo Touvron, et al. Emerging properties in self-supervised vision transformers. *ICCV*, 2021.

[5] Sherry X Chen, Misha Sra, and Pradeep Sen. Instruct-clip: Improving instruction-guided image editing with automated data refinement using contrastive learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 28513–28522, 2025.

[6] Yurui Chen, Chun Gu, Junzhe Jiang, Xiatian Zhu, and Li Zhang. Periodic vibration gaussian: Dynamic urban scene reconstruction and real-time rendering. *International Journal of Computer Vision*, 134(3):83, 2026.

[7] 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 *International Conference on Learning Representations (ICLR)*, 2025.

[8] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on robot learning*, pp. 1–16. PMLR, 2017.

[9] Ayaan Haque, Matthew Tancik, Alexei A. Efros, Aleksander Holynski, and Angjoo Kanazawa. Instruct-nerf2nerf: Editing 3d scenes with instructions. *arXiv preprint arXiv:2303.12789*, 2023. URL https://arxiv.org/pdf/2303.12789.

[10] Georg Hess, Carl Lindstr"om, Maryam Fatemi, Christoffer Petersson, and Lennart Svensson. Splatad: Real-time lidar and camera rendering with 3d gaussian splatting for autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 11982–11992, 2025.

[11] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 17853–17862, 2023.

[12] Xiaosong Jia, Zhenjie Yang, Qifeng Li, Zhiyuan Zhang, and Junchi Yan. Bench2drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. *Advances in Neural Information Processing Systems*, 37:819–844, 2024.

[13] Napat Karnchanachari, Dimitris Geromichalos, Kok Seang Tan, Nanxiang Li, Christopher Eriksen, Shakiba Yaghoubi, Noushin Mehdipour, Gianmarco Bernasconi, Whye Kit Fong, Yiluan Guo, et al. Towards learning-based planning: The nuplan benchmark for real-world autonomous driving. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 629–636. IEEE, 2024.

[14] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), 2023. URL https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/3d_gaussian_splatting_high.pdf.

[15] Muhammed Kocabas, Jen-Hao Rick Chang, James Gabriel, Oncel Tuzel, and Anurag Ranjan. HUGS: Human gaussian splats. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[16] Tianyu Li, Yihang Qiu, Zhenhua Wu, Carl Lindstr"om, Peng Su, Matthias Nießner, and Hongyang Li. Mtgs: Multi-traversal gaussian splatting. *arXiv preprint arXiv:2503.12552*, 2025.

[17] Yiyuan Liang, Zhiying Yan, Liqun Chen, Jiahuan Zhou, Luxin Yan, Sheng Zhong, and Xu Zou. Driveeditor: A unified 3d information-guided framework for controllable object editing in driving scenes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 5164–5172, 2025.

[18] Chih-Hao Lin, Zian Wang, Ruofan Liang, Yuxuan Zhang, Sanja Fidler, Shenlong Wang, and Zan Gojcic. Controllable weather synthesis and removal with video diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13580–13591, 2025.

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

[20] Chensheng Peng, Chengwei Zhang, Yixiao Wang, Chenfeng Xu, Yichen Xie, Wenzhao Zheng, Kurt Keutzer, Masayoshi Tomizuka, and Wei Zhan. Desire-gs: 4d street gaussians for static-dynamic decomposition and surface reconstruction for urban driving scenes. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 6782–6791, 2025.

[21] Chenghao Qian, Wenjing Li, Yuhu Guo, and Gustav Markkula. Weatheredit: Controllable weather editing with 4d gaussian field. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 8511–8519, 2026.

[22] Bosheng Qin, Juncheng Li, Siliang Tang, Tat-Seng Chua, and Yueting Zhuang. Instructvid2vid: Controllable video editing with natural language instructions. In *2024 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE, 2024.

[23] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

[24] Paul-Edouard Sarlin et al. Superglue: Learning feature matching with graph neural networks. *CVPR*, 2020.

[25] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurélien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Waymo open dataset: Autonomous driving dataset. *CVPR*, 2020.

[26] Adam Tonderski, Carl Lindstr"om, 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*, pp. 14895–14904, 2024.

[27] Prune Truong, Marie-Julie Rakotosaona, Fabian Manhardt, and Federico Tombari. Sparf: Neural radiance fields from sparse and noisy poses. IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2023.

[28] Haithem Turki, Qi Wu, Xin Kang, Janick Martinez Esturo, Shengyu Huang, Ruilong Li, Zan Gojcic, and Riccardo de Lutio. Simuli: Real-time lidar and camera simulation with unscented transforms. *arXiv preprint arXiv:2510.12901*, 2025.

[29] Ye Wang, Ruiqi Liu, Jiang Lin, Fei Liu, Zili Yi, Yilin Wang, and Rui Ma. Omnistyle: Filtering high quality style transfer data at scale. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 7847–7856, 2025.

[30] Yajiao Xiong, Xiaoyu Zhou, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. Driving-gaussian++: Towards realistic reconstruction and editable simulation for surrounding dynamic driving scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

[31] 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 *European Conference on Computer Vision (ECCV)*, 2024.

[32] 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)*, 2024.

[33] 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*, pp. 1389–1399, 2023.

[34] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *International Conference on Learning Representations (ICLR)*, 2024.

[35] Zehao Zhu, Yuliang Zou, Chiyu Max Jiang, Bo Sun, Vincent Casser, Xiukun Huang, Jiahao Wang, Zhenpei Yang, Ruiqi Gao, Leonidas Guibas, et al. Scenecrafter: Controllable multi-view driving scene editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6812–6822, 2025.

APPENDIX

A WHY 3D CONSISTENCY MATTERS FOR DYNAMIC STREET EDITING

For dynamic street scenes, a useful editor must preserve more than appearance realism. It must maintain scene layout, object identity, and cross-view correspondences across cameras and time so that the edited outputs remain compatible with reconstruction, tracking, and planning. If local structures change inconsistently across viewpoints, the resulting sequence may still look plausible frame by frame, but it no longer corresponds to a stable 3D environment.

A concise way to express this requirement is through shared 3D correspondences. For two views with poses $\mathbf{P}_k$ and $\mathbf{P}_l$, image structures that correspond to the same world point should back-project to a common 3D location. Under editing, this geometric relation should remain valid even when appearance changes:

$$\pi^{-1}(\mathbf{x}_k^i; \mathbf{P}_k, d_k^i) = \pi^{-1}(\mathbf{x}_l^j; \mathbf{P}_l, d_l^j), \tag{12}$$

where $\mathbf{x}_k^i$ and $\mathbf{x}_l^j$ denote matched image structures, d_k^i and d_l^j are their depths, and π^{-1} denotes back-projection into 3D. In other words, editing should preserve not only appearance correspondence but also the underlying 3D agreement across viewpoints.

This condition is closely related to the geometric assumptions used by multi-view reconstruction methods such as NeRF [19] and 3D Gaussian Splatting [14]. In the setting of controllable street editing, violating these correspondences can introduce unstable depth, inconsistent object geometry, and frame-to-frame motion artifacts, which in turn degrade downstream temporal tasks. Preserving them supports stable geometry, more reliable object trajectories, and stronger cross-camera consistency after weather or object-level modifications.

This perspective motivates the main design choices in StyledStreets: rigid camera-vehicle pose refinement, uncertainty-aware editing supervision, and geometry-preserving unified stylization. Together, these components encourage the edited scene to remain a coherent 3D representation that supports downstream vehicle tracking and ego-centric planning instead of serving only as a visually stylized output.

B POSE OPTIMIZATION: QUALITATIVE COMPARISON

This section provides the qualitative evidence for the pose-optimization claim of the main paper. Freezing the camera-to-vehicle extrinsics and optimizing only the vehicle trajectory keeps multi-camera geometry intact, and the effect is most visible in the high-frequency structure of the reconstructed vehicles rather than in aggregate PSNR. Figure 8 contrasts the three settings.



Figure 8: View synthesis comparison: (a) our model preserves details (headlights, wheel spokes) with pose optimization; (b) without pose optimization, structural elements remain but high-frequency details are lost; (c) OmniRe [7] exhibits blurred geometries due to pose misalignment. Quantitative superiority is reported in Table 1 of the main paper.

The contrast in panel (c) is the informative one: OmniRe optimizes per-camera poses directly, so its cameras can drift independently of the rigid mounting that produced the capture. The reconstruction still resolves large structure, which is why the aggregate metrics stay competitive, but thin elements such as wheel spokes and headlight housings lose coherence. Our constrained formulation avoids that failure mode at its source rather than correcting it after the fact, which is also why the gain shows up more clearly in vehicle-only metrics than in full-image PSNR.

C Deformable Object Modeling with 4DGS

The 4DGS component is a secondary element of the framework: it affects how deformable objects such as pedestrians and cyclists are represented, but it does not change the pose optimization, the editing supervision, or the downstream evaluation that carry the main results. We document it here for completeness, and Figure 9 shows the comparison that motivates the design choice.



Figure 9: Deformable object modeling: 4DGS (b) outperforms skeleton-based methods like OmniRe (a) by handling self-occlusion and subtle motions without skeletal priors.

Skeletal priors assume that a deformable object's motion can be factored into a small set of articulated joints. That assumption holds for pedestrians walking in the open, but it breaks down when a limb is occluded by the body or by another object, and it cannot express soft deformation such as clothing or a carried item. Panel (a) shows the resulting artifacts under OmniRe. Modeling the deformation directly in the Gaussian representation, as in panel (b), avoids the articulation assumption entirely and recovers motion under self-occlusion. We adopted it for that reason, at a modest cost in training time that the results of the main paper already account for.

Table 10: Additional human results. P.=PSNR, S.=SSIM. Both the human skeleton and surface mesh are estimated by EasyMocap (`https://github.com/zju3dv/EasyMocap`). Added style-editing cost is shown in parentheses.

	Recon.		Novel		Train Time (h)		
Method	P.	S.	P.	S.	1C+S	3C+S	5C+S
StreetGS	16.83	.420	16.55	.393	3.58	4.20	4.82
OmniRe	28.15	.845	24.36	.727	2.61	3.07	3.53
Ours	28.25	.848	20.00	.442	**1.48(+.6)**	**1.72(+.7)**	**2.06(+.7)**
Ours+Skel	28.87	.872	25.96	.808	1.52(+.7)	1.87(+.7)	2.22(+.7)
Ours+Surf	**29.02**	**.879**	**26.33**	**.812**	1.56(+.7)	1.91(+1.0)	2.26(+1.1)

D Additional Quantitative Results

This supplementary document focuses on the most useful extra quantitative material beyond the main paper. In particular, we report non-rigid human results and efficiency trends under multi-camera settings, which complement the main paper's street-scene reconstruction and editing evaluation.

For non-rigid objects such as humans, we study two training-free regularization strategies derived from EasyMocap (`https://github.com/zju3dv/EasyMocap`): a skeleton-based variant and a surface-based variant. The surface-based regularization yields the strongest human reconstruction and novel-view synthesis results, while all variants remain faster than prior street-scene baselines under 1-, 3-, and 5-camera settings. We adapt the StreetGaussian backbone from the public implementation at `https://github.com/LightwheelAI/street-gaussians-ns`, which provides the reconstruction basis for these comparisons.