Diffusion-guided Generalizable Enhancer for Urban Scene Reconstruction

Henry Che[1,3] Jingkang Wang[1,2] Yun Chen[1,2] Ze Yang[1,2] Sivabalan Manivasagam[1,2] Raquel Urtasun[1,2]
[1]Waabi [2] University of Toronto [3]University of Illinois Urbana-Champaign
{jwang, ychen, zyang, siva, urtasun}@waabi.ai, hungdc2@illinois.edu

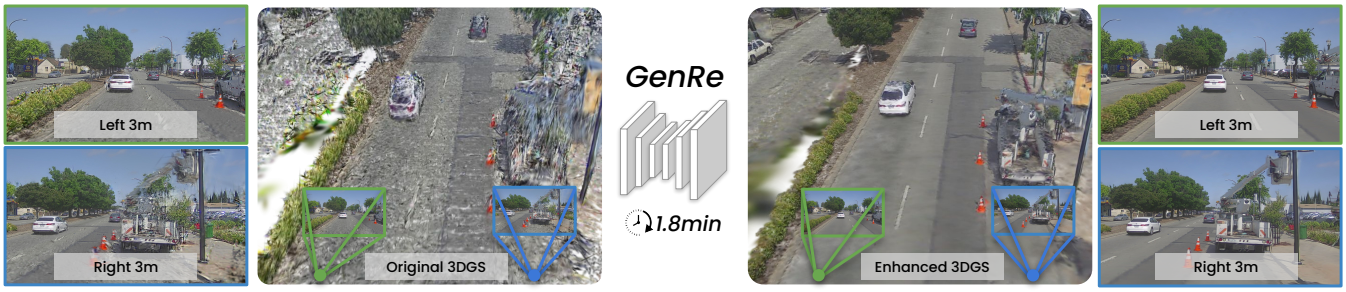

Fig. 1. We introduce *GenRe*, a novel diffusion-guided generalizable enhancer for urban scene reconstruction. *GenRe* takes as input any pretrained 3D Gaussian representation and fixes the deficiencies within minutes, producing robust, high-fidelity reconstructions that render reliably at novel viewpoints.

***Abstract*— Urban scene reconstruction from real-world observations has emerged as a powerful tool for self-driving development and testing. While current neural rendering approaches achieve high-fidelity rendering along the recorded trajectories, their quality degrades significantly under large viewpoint shifts, limiting the applicability for closed-loop simulation. Recent works have shown promising results in using diffusion models to enhance quality at these challenging viewpoints and distill improvements back into 3D representations. However, they often require costly per-scene optimization, and the distilled representations remain fragile and fail to generalize beyond limited synthesized views. To address these limitations, we propose *GenRe*, a novel diffusion-guided generalizable enhancer for urban scene reconstruction. *GenRe* takes as input any pretrained 3D Gaussian representation and fixes the deficiencies within a few minutes. By learning to distill generative priors across diverse scenes, *GenRe* produces robust and high-fidelity representation efficiently that generalizes reliably to challenging unseen viewpoints (*e.g.*, lane change). Experiments show that *GenRe* outperforms existing methods in both quality and efficiency and benefits various downstream tasks, enabling robust and scalable sensor simulation for autonomous driving.**

I. Introduction

Realistic simulation is essential to test safety-critical self-driving systems in a safe and scalable manner [28]. Data-driven approaches, which construct digital twins from real-world observations [14], [27], have emerged as a key paradigm for sensor simulation. In contrast to artist-created, game-engine–based virtual worlds [6], [24], it provides scalability, realism, and diversity, forming a strong foundation for large-scale, closed-loop simulation for autonomy development.

Neural rendering approaches such as NeRF [16] achieve realistic reconstruction of urban driving scenes from camera and LiDAR data [37], [25], but are slow to render. Recently, 3D Gaussian Splatting (3DGS) [12] models scenes as large sets of explicit anisotropic Gaussians and renders them via rasterization, yielding faster rendering. Subsequent works [34], [2], [4], [9] extended this technique to dynamic urban driving scenes. However, these differentiable-rendering pipelines often overfit to the training trajectories, leading to significant artifacts and quality degradation when extrapolating beyond original trajectory (*e.g.*, meter-scale shifts). In particular, 3DGS's over-parameterized primitives, in combination with the shape-radiance ambiguity when trained on single-trajectory ego views [30], can exacerbate memorization of training views, producing floaters/holes and inconsistent surfaces under extrapolation (Fig. 1 left). Moreover, they cannot hallucinate plausible content in unobserved/occluded regions, resulting in holes and missing structures at extrapolated views. These artifacts may reduce the fidelity of closed-loop sensor simulation, where the driving agent can deviate significantly from the recorded trajectory.

To address these limitations, recent work introduces physics-based and data-driven priors (*e.g.*, additional regularization [11], supervision from pre-trained vision models [20], shared decoders [9], and generative models [11]) to stabilize the learned representation. Since these priors are not trained to handle reconstruction artifacts or representation-specific degradations, the gains are limited and often produce blurry results. Most recently, researchers propose to train *2D neural fixers* by fine-tuning diffusion models to correct artifacts at novel views by creating simulation and real pairs of held-out views [29], [35], which yields significant visual improvements. To further improve 3D consistency and use for simulation purposes, subsequent methods distill the visual improvements back into the underlying 3D representation [7], [30], [40], [17], [39]. However, despite the impressive results, these pipelines require hours of per-scene optimization and have difficulty scaling. In addition, the distilled representations remain fragile and usually generalize only to small synthesized viewpoint shifts where the fixer performs well, with significant degradation under larger extrapolations.

Towards this goal, we present *GenRe*, a diffusion-guided

generalizable enhancer for urban scene reconstruction. *GenRe* takes any pre-trained 3D Gaussian representation and fixes the deficiencies within a few minutes. At the heart of *GenRe* are two modules. A one-step diffusion neural fixer predicts view-conditioned residuals at novel views, guided by geometry and appearance cues. A generalizable 3D enhancer then updates Gaussian parameters to enforce multi-view and geometric consistency while preserving fidelity along both recorded and novel trajectories. The enhancer learns to transfer diffusion priors into iterative 3D-consistent updates by training across diverse scenes. *GenRe* produces stable renderings under meter-scale viewpoint shifts and lane changes, while plausibly completing unobserved or occluded regions.

Experiments on diverse urban scenes show that *GenRe* outperforms state-of-the-art scene reconstruction and neural fixer methods at challenging novel viewpoints while maintaining competitive performance along the recorded trajectories. We also show that *GenRe* benefits various downstream tasks, including higher-quality simulation of novel maneuvers, reduced domain gap for downstream perception tasks, and improved 3D object detection training with augmentation, unveiling the potential for robust and scalable sensor simulation.

II. Related Work

a) Urban scene reconstruction: Seminal works in differentiable rendering such as NeRF [16] and 3DGS [12] have driven rapid progress in urban scene reconstruction [37], [25], [34], [4], [9]. These methods represent the scene as either an implicit radiance field or a set of 3D Gaussians which can be differentiably rendered and supervised with reconstruction losses through per-scene optimization. To improve efficiency, recent approaches [1], [26] adopt a feed-forward paradigm, achieving substantial speedup in reconstruction. Despite these advances, both per-scene and generalizable methods achieve high quality primarily along the training trajectories, but their performance degrades significantly under large viewpoint shifts. Our work aims to address this limitation by harnessing diffusion models to enhance the rendering quality in a 3D-consistent and efficient manner.

b) Reconstruction with generative priors: To regularize the representation and improve visual plausibility at extrapolated views, one popular paradigm is to incorporate priors from large-scale generative models. A common strategy is to adapt the Score Distillation Sampling (SDS) loss [21], which guides the optimization of 3D representations by aligning rendered images with the gradients of a pre-trained diffusion model. While originally developed for object-level text-to-3D generation, recent methods have explored extending SDS to scene-level reconstruction [31]. In the driving domain, VEGS [11] adapts SDS for dynamic urban scenes and further introduce surface normal priors for regularization. However, despite these improvements, the optimization remains unstable due to competing and sometimes noisy losses, and remains scene-specific and computationally expensive, limiting its applicability to scalable simulation. Another line of work leverages diffusion models to directly generate 3D scenes conditioned on several input images [15], [22], [8], which typically sacrifices geometric and photometric accuracy.

c) Neural fixer of 3D scenes: Our work is closely related to the recent advances in *neural fixer* for 3D scenes, which aim to reduce artifacts such as holes, blurriness and distortions under extrapolated viewpoints. The representative work SplatFormer [3] trains a network on large-scale object-centric data [5] to refine pre-trained 3D Gaussian representations using supervision from extreme out-of-distribution views. However, this strategy is not applicable to urban driving scenes, where only one single pass is typically available and accurate supervision beyond the recorded trajectories is lacking. To overcome this limitation, recent works [29] employ generative models to generate the pseudo ground-truth at the challenging viewpoints (*e.g.*, 2-3m lateral shifts), and fine-tune the 3D representation with the generated images [30], [7], [17], [39]. However, they require costly per-scene optimization, and the distilled representations tend to overfit to synthesized views while exhibiting noticeable artifacts under larger extrapolation. In contrast, *GenRe* proposes a *generalizable enhancer*, providing significant speedups and improved robustness for urban scene reconstruction.

III. Robust Scene Reconstruction with Diffusion-Guided Generalizable Enhancer

Given multi-view camera ($\mathcal{I}_{\text{src}}$) and LiDAR ($\mathcal{P}$) observations from large-scale driving scenes, our goal is to reconstruct robust 3D scene representation at scale that handles large viewpoint shifts and occlusions for reliable re-simulation and downstream evaluation. Towards this goal, we propose a diffusion-guided generalizable 3D enhancer that improves the 3D Gaussian representation for robust rendering under challenging novel viewpoints. We first briefly review the 3DGS-based scene representation in Sec. III-A. We then introduce two key modules: a one-step diffusion-based neural fixer (*FNet*) that predicts view-conditioned residuals (Sec. III-B), and a generalizable enhancer network (*ENet*) that enforces multi-view consistency by refining Gaussian attributes (Sec. III-C) iteratively. Finally, we show how these two modules can be integrated into a robust generalizable reconstruction pipeline (*GenRe+*) for scalable urban scene simulation in Sec. III-D.

A. 3DGS-based Scene Representation

3D Gaussian Splatting (3DGS) [12] represents the scene as a set of anisotropic 3D Gaussians $\mathcal{G} = \{g_i\}_{i=1}^{M}$ that can be differentiably rasterized in real time. Each Gaussian $g_i = \{\boldsymbol{\mu}_i, \boldsymbol{s}_i, \boldsymbol{q}_i, o_i, \mathbf{c}_i\} \in \mathbb{R}^{14}$ consists of mean $\boldsymbol{\mu}_i \in \mathbb{R}^3$, scale vector $\mathbf{s}_i \in \mathbb{R}^3$, quaternion $\boldsymbol{q} \in \mathbb{R}^4$, opacity value $o_i \in [0, 1]$, and RGB color $\mathbf{c}_i \in [0, 1]^3$. For urban driving scenes, we decompose $\mathcal{G}$ into three subsets: a static background $\mathcal{G}_B$, dynamic actors $\mathcal{G}_A$, and a distant region $\mathcal{G}_D$ (*e.g.*, far-away buildings and sky). Foreground actors are tracked across frames using 3D bounding boxes that specify their size and location. The static background and dynamic actors are initialized from aggregated LiDAR points. A fixed number of Gaussians are placed at a large distance to represent $\mathcal{G}_D$.

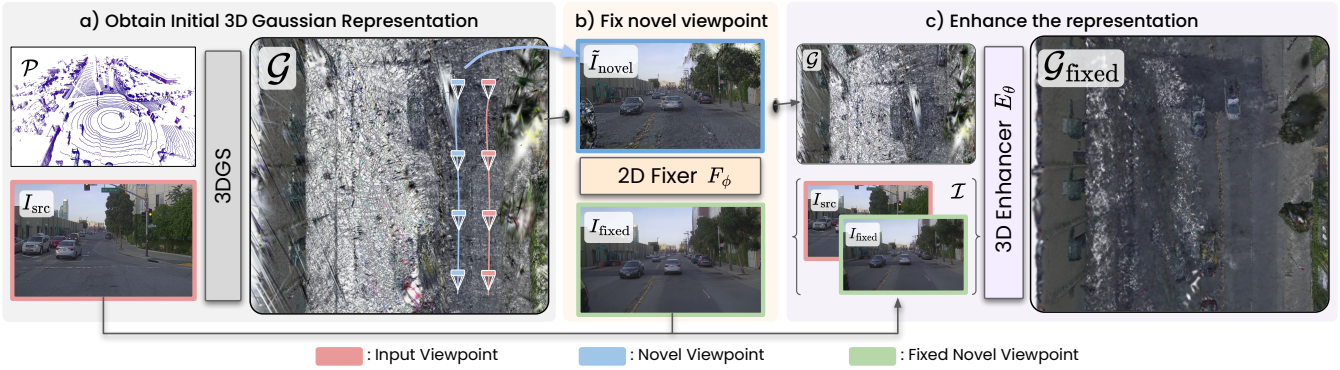

Fig. 2. ***GenRe* pipeline for urban scene reconstruction.** *GenRe* is composed of three steps. First, any 3DGS-based reconstruction methods are used to obtain an initial representation. Then, we render at novel viewpoint (e.g., 3m shifts) and adopt a diffusion-based neural fixer *FNet* (Sec. III-B) to fix the degraded artifacts. Finally, we leverage a generalizable enhancer *ENet* (Sec. III-C) that predicts per-Gaussian residuals to enhance the 3D representation.

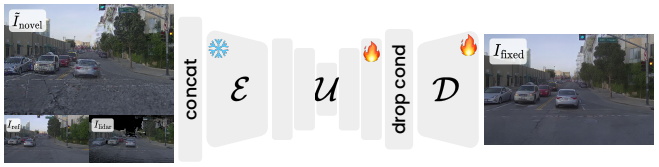

Fig. 3. **2D neural fixer (*FNet*) overview.** *FNet* takes a 3DGS-rendered view $\tilde{I}$, conditions on the reference image I_{ref} and the rendered LiDAR map I_{lidar}, and produces the fixed image I_{fixed}. We fine-tune *FNet* from the pre-trained single-step diffusion model *SD-Turbo* [23].

Given the camera projection matrix Π, the 3D Gaussians are projected onto the image plane and rasterized into per-ray fragments. After depth sorting along each ray, the color $\mathbf{C}$ of a pixel $\mathbf{p}$ is computed by front-to-back alpha compositing:

$$\alpha_i = o_i \exp\left(-\frac{1}{2}(\mathbf{p}-\hat{\boldsymbol{\mu}}_i)^T \hat{\boldsymbol{\Sigma}}_i^{-1}(\mathbf{p}-\hat{\boldsymbol{\mu}}_i)\right), \tag{1}$$

$$\mathbf{C} = \sum_{i=1}^{N} w_i\, \mathbf{c}_i, \quad w_i = \alpha_i \prod_{j<i}(1-\alpha_j), \tag{2}$$

where $\hat{\boldsymbol{\mu}}_i$ and $\hat{\boldsymbol{\Sigma}}_i$ are the projected mean and covariance of the i-th Gaussian, computed from its parameters $(\boldsymbol{\mu_i}, \boldsymbol{s_i}, \boldsymbol{q_i})$ and camera projection Π [12]. α_i is the transmittance and w_i is the weight. The image is rendered $\tilde{I} = f_{\text{render}}(\mathcal{G}; \Pi)$.

B. Enhancing NVS with 2D Neural Fixer (FNet)

Although 3DGS-based reconstruction methods achieve high-quality rendering at the original or interpolated views, they often degrade significantly at viewpoints that deviate substantially from the recorded trajectories due to overfitting to the training views and the lack of supervision in unobserved regions. Inspired by recent works in 2D neural fixer [35], [30], we therefore learn an image-space, diffusion-based fixer that reduces artifacts under large viewpoint shifts.

a) 2D Fixer Network: Given an image $\tilde{I}$ rendered from a pre-trained 3DGS scene, we learn a neural fixer F_ϕ to fix the rendering artifacts and obtain a more photorealistic image. During training, we use paired data and supervise $F_\phi(\tilde{I})$ to match the ground-truth image I. Since $\tilde{I}$ is already close to I, we adopt a pre-trained single-step diffusion model *SD-turbo* for efficiency. Following [18], we encode $\tilde{I}$ into the latent space and feed it to the diffusion UNet to obtain a noisy latent. We then apply a single-step sampler to denoise the latent and decode it with the frozen VAE decoder to obtain the fixed image $I_{\text{fixed}} = F_\phi(\tilde{I})$. The fixer is supervised in image space using a photometric term and a perceptual term:

$$\mathcal{L}_{\text{fixer}} = \mathcal{L}_{\text{rgb}} + \mathcal{L}_{\text{lpips}}. \tag{3}$$

b) Appearance and geometry conditioning: Although the vanilla fixer with rendered-image-only improves fidelity, it still produces blurry or hallucinatory content and, as an image-based diffusion model, lacks temporal consistency. To improve fidelity and consistency, we augment the 2D fixer with additional appearance and geometry conditioning. In particular, we take as the reference view I_{ref} the training image whose camera pose is closest to the target view, and we render an accumulated LiDAR map I_{lidar} (*i.e.*, a 3DGS rendering of aggregated, colored LiDAR points at the target view) to provide explicit geometric cues (Fig. 3). Formally, let $\mathcal{E}$ and $\mathcal{D}$ denote the VAE encoder and decoder, τ be the noise timestep with schedule σ_τ, and $\mathcal{U}$ be the single-step diffusion UNet, we have

$$\mathbf{z}_{\text{render}} = \mathcal{E}(\tilde{I}), \quad \mathbf{z}_{\text{ref}} = \mathcal{E}(I_{\text{ref}}), \quad \mathbf{z}_{\text{lidar}} = \mathcal{E}(I_{\text{lidar}}), \tag{4}$$

$$\mathbf{z}_\tau = \mathbf{z}_{\text{render}} + \sigma_\tau \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{5}$$

$$\tilde{\boldsymbol{\epsilon}} = \mathcal{U}(\mathbf{z}_\tau, \tau | [\mathbf{z}_{\text{ref}}, \mathbf{z}_{\text{lidar}}]), \tag{6}$$

$$\tilde{\mathbf{z}} = \mathbf{z}_\tau - \sigma_\tau \tilde{\boldsymbol{\epsilon}}, \quad I_{\text{fixed}} = \mathcal{D}(\tilde{\mathbf{z}}), \tag{7}$$

where we encode $\tilde{I}$, I_{ref}, and I_{lidar} with a shared VAE encoder and feed their latents $\mathbf{z}$ through the UNet. Inspired by [30], [13], we stack latents along the view axis, and each UNet block applies a lightweight condition-mixing self-attention to share information across views before restoring the layout. This conditions denoising on appearance cues from the reference and geometric cues from LiDAR, improving view consistency and reducing blurriness, hallucination, and flickering under large viewpoint shifts. Fig. 3 shows the overview of *FNet*.

c) Implementation details: We initialize the VAE $(\mathcal{E}, \mathcal{D})$ and UNet $(\mathcal{U})$ from pre-trained *SD-Turbo* [23] and fine-tune $\mathcal{D}$ and $\mathcal{U}$ (with LoRA [10]). We remove the CLIP cross-attention layers and set the noise timestep to $\tau = 200$ following [30]. We train FNet at the resolution of 720×1280 for 20k steps using AdamW with a batch size of 8.

C. Generalizable 3D Enhancer Network (ENet)

While *FNet* improves novel-view renderings, its per-frame runtime limits real-time use, and image-space corrections alone do not guarantee multi-view consistency. For fast simulation, the improvements must reside in the 3D representation. Prior work address this by distilling corrected images back into 3D on a per-scene basis, which takes hours and often yields models that generalize only to small synthesized shifts with noticeable degradation under larger extrapolations. Motivated by this gap, we propose a generalizable 3D enhancer E_θ that updates 3DGS parameters in an iterative manner. Trained across diverse scenes, the 3D enhancer transfers the knowledge of the 2D fixer into a 3D-consistent and robust representation within a few feed-forward steps.

a) 3D Enhancer Network: Given a pre-trained 3DGS scene $\mathcal{G}$ and the fixer F_θ, we train a generalizable enhancer E_θ that predicts per-Gaussian residuals to update $\mathcal{G}$ into a higher-fidelity representation $\mathcal{G}_{\text{fixed}}$. Specifically, conditioned on camera poses Π and images $\mathcal{I} = \{\mathcal{I}_{\text{src}}, \mathcal{I}_{\text{fixed}}\}$ (a mixture of ground-truth frames I and fixer outputs $I_{\text{fixed}} = F_\phi(\tilde{I})$), the enhancer network outputs

$$\Delta\mathcal{G} = E_\theta(\mathcal{G}; \mathcal{I}, \Pi) = \{\Delta\boldsymbol{\mu}_i, \Delta\boldsymbol{s}_i, \Delta\boldsymbol{q}_i, \Delta o_i, \Delta\boldsymbol{c}_i\}_{i=1}^{M}. \quad (8)$$

To obtain the final representation, we apply the residuals to the initial 3D Gaussians:

$$\mathcal{G}_{\text{fixed}} = \mathcal{G} + \Delta\mathcal{G} = \{\Delta g_i + g_i\}_{i=1}^{M}. \quad (9)$$

Let $\tilde{I}_{\text{src}}$ and $\tilde{I}_{\text{novel}}$ denote renders from the current 3DGS at recorded and extrapolated viewpoints. We train E_θ across many scenes by minimizing the following training objective:

$$\mathcal{L} = \mathcal{L}_{\text{src}}(\tilde{I}_{\text{src}}, I) + \lambda_{\text{novel}}\mathcal{L}_{\text{novel}}(\tilde{I}_{\text{novel}}, I_{\text{fixed}}), \quad (10)$$

$$\mathcal{L}_{\text{src}} = \mathcal{L}_{\text{rgb}} + \lambda_{\text{lpips}}\mathcal{L}_{\text{lpips}} + \lambda_{\text{ssim}}\mathcal{L}_{\text{ssim}} + \lambda_{\text{depth}}\mathcal{L}_{\text{depth}}, \quad (11)$$

$$\mathcal{L}_{\text{novel}} = \mathcal{L}_{\text{rgb\_novel}} + \lambda_{\text{lpips}}\mathcal{L}_{\text{lpips\_novel}} + \lambda_{\text{ssim}}\mathcal{L}_{\text{ssim\_novel}}, \quad (12)$$

where the source-view terms compare $\tilde{I}_{\text{src}}$ with ground-truth images I to preserve fidelity along the recorded trajectories, while the novel-view terms compare $\tilde{I}_{\text{novel}}$ with fixer targets I_{fixed} to improve robustness under challenging viewpoints.

b) Iterative refinement: Inspired by G3R [1], we unroll the enhancer for T iterations instead of using a single forward pass. At iteration $t \in \{0, \ldots, T-1\}$, the network E_θ takes the current Gaussians $\mathcal{G}_t$ (with $\mathcal{G}_0 = \mathcal{G}$) and per-point gradients $\nabla\mathcal{G}_t$ as guidance features, computed by backpropagating the loss in Eq. 10 with respect to the Gaussian parameters. Given the predicted residuals $\Delta\mathcal{G}_t$, we then update the representation to obtain $\mathcal{G}_{t+1}$. The weights of E_θ are shared across iterations and the enhanced Gaussians are $\mathcal{G}_{\text{fixed}} = \mathcal{G}_T$. This iterative refinement (also known as learned optimization) substantially improves visual quality under extrapolated viewpoints.

c) Implementation details: We use Sparse UNet as the 3D enhancer network. We set $T = 12$, and train the model for 2k iterations with a batch size of 8. We set $\lambda_{\text{rgb}} = 0.8$, $\lambda_{\text{lpips}} = 0.2$, $\lambda_{\text{ssim}} = 0.2$, $\lambda_{\text{depth}} = 0.01$, $\lambda_{\text{novel}} = 0.5$. We split each training sequence into 20-frame non-overlapping chunks, and optimize the 3DGS scene for each chunk to obtain $\mathcal{G}$ for training *ENet*.

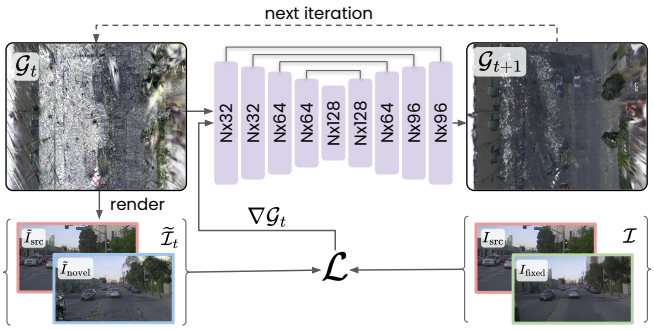

Fig. 4. **Generalizable 3D enhancer (*ENet*) overview.** *ENet* iteratively refines a 3DGS scene using rendering-guided gradients. At iteration t, *ENet* takes the current 3D Gaussians $\mathcal{G}_t$ and per-Gaussian gradients $\nabla\mathcal{G}_t$ (from rendering loss) and predicts residuals $\Delta\mathcal{G}_t$ to update the scene to $\mathcal{G}_{t+1}$. Source and novel views are compared with ground-truth I and fixed targets I_{fixed} to compute losses $\mathcal{L}_{\text{src}}(\tilde{I}_{\text{src}}, I)$ and $\mathcal{L}_{\text{novel}}(\tilde{I}_{\text{novel}}, I_{\text{fixed}})$, whose backprop gives $\nabla\mathcal{G}_{t+1}$. Unrolling T steps yields the enhanced scene $\mathcal{G}_{\text{fixed}}$.

D. GenRe (Enhancer) and GenRe+ (Reconstructor)

We now describe how to integrate the neural fixer *FNet* and the 3D enhancer *ENet* into a unified framework, *GenRe*, for robust urban scene reconstruction. As shown in Fig. 2, starting from an initial 3DGS representation $\mathcal{G}$, we first render novel viewpoints and correct artifacts with *FNet*. These fixed images $\mathcal{I}_{\text{fixed}}$ are then used by *ENet* together with source-view real images $\mathcal{I}_{\text{src}}$ to refine the underlying Gaussian representation, distilling the 2D corrections back into 3D. Formally, we have

$$\mathcal{I}_{\text{fixed}} = F_\phi(f_{\text{render}}(\mathcal{G}; \Pi_{\text{novel}})) \quad (13)$$

$$\mathcal{G}_{\text{fixed}} = E_\theta(\mathcal{G}, \{\mathcal{I}_{\text{src}}, \mathcal{I}_{\text{fixed}}\}; \{\Pi_{\text{src}}, \Pi_{\text{novel}}\}). \quad (14)$$

Although *ENet* is designed to enhance existing 3DGS scenes, its formulation as a learned optimizer makes it naturally amenable to inducing a 3DGS representation directly from data. We therefore adapt it as a robust and efficient generalizable reconstruction module that predicts scenes from raw sensory inputs, which we refer to as *GNet*. To further improve robustness under extrapolation, we include *FNet*-generated images as auxiliary supervision during training with a small weight. This preserves fidelity along recorded trajectories while strengthening stability at challenging novel viewpoints. Benefiting from the 2D fixer priors, *GNet* provides strong standalone reconstruction and achieves superior robustness compared to existing methods. We obtain *GNet* by fine-tuning the enhancer E_θ with the rendering objective in Eq. 10 (λ_{novel}=0.1). We unroll T=24 iterations to increase reconstruction capacity.

Finally, we show that combining *GNet* (generalizable reconstruction), *FNet* (2D fixer), and *ENet* (3D enhancer) yields a more robust and scalable pipeline, *GenRe+* (*GNet* $\rightarrow$ *FNet* $\rightarrow$ *ENet*), for urban scene reconstruction. Specifically, *GNet* reconstructs the base scene representation from sensory data; *FNet* corrects artifacts at novel-view rendering; and *ENet* distills these corrections back into the 3D representation.

IV. Experiments

We evaluate against state-of-the-art (SoTA) urban scene reconstruction approaches and per-scene optimization with

TABLE I

COMPARISON TO STATE-OF-THE-ART RECONSTRUCTION METHODS ON INTERPOLATED AND EXTRAPOLATED VIEWS.

Methods	*Interpolation*		*Extrapolation* (Moderate)			*Extrapolation* (Hard)		Recon time
	PSNR↑	SSIM↑	FID@1m↓	FID@2m↓	FID@3m↓	FID@4m↓	FID@5m↓	Minute↓
Standalone reconstruction								
3DGS [12]	23.45	0.707	82.63	128.10	169.69	205.09	231.70	41.90
StreetGS [34]	23.14	0.693	**68.99**	97.05	127.06	153.76	176.43	47.62
SplatAD [9]	**24.93**	**0.768**	84.21	122.56	160.43	188.00	210.24	113.62
G3R [1]	23.28	0.673	89.75	114.94	147.50	174.64	191.33	**0.90**
Ours (*GNet*)	23.56	0.689	70.04	**86.43**	**106.07**	**124.65**	**138.30**	0.96
Reconstruction with neural fixers								
StreetCrafter [35]	23.33	0.690	**59.44**	81.14	97.09	118.94	141.14	127.33
Difix3D [30]	**23.34**	**0.705**	60.29	83.35	102.16	137.80	167.28	35.32
Ours (*GenRe+*)	23.28	0.692	60.69	**74.50**	**88.04**	**102.73**	**114.19**	**2.77**

Fig. 5. Qualitative comparison to state-of-the-art neural reconstruction methods under large extrapolation. Our method yields higher realism, fewer artifacts.

neural fixers. The performance is measured on both recorded trajectories and extrapolated viewpoints (*i.e.,* lateral shifts). We then show that our generalizable enhancer *GenRe* plugs into different 3DGS-based methods in a zero-shot manner, demonstrating its versatility and robustness. Finally, we showcase *GenRe+* benefits various downstream tasks including simulation, perception evaluation and augmented training.

A. Experiment Details

a) Experiment setup: We evaluate on PandaSet [33], a self-driving dataset with diverse, large-scale urban scenes. PandaSet contains 103 sequences captured by six 1080p cameras and a 64-beam LiDAR at 10Hz. We follow the split of [1], using 93 sequences for training, and 10 for testing. We assess performance on both in-trajectory (*interpolation*) and out-of-trajectory (*extrapolation*) views. In all experiments, we subsample every fourth frame as input (25% of views) to reconstruct the scene representation and use the remaining 75% for interpolation evaluation. For extrapolation, we synthesize lateral shifts of 1–5m (1-3m: *moderate*; 4–5m: *hard*) and report FID [19]. For methods with neural fixers, we apply the fixers only at 3m, where prior work reports reasonable fidelity without pronounced consistency issues or hallucination [35] and evaluate up to 5m to test the robustness under larger extrapolations.

b) Baselines: We compare *GenRe* against SoTA urban scene reconstruction in two settings: (1) *standalone reconstruction*, including per-scene methods 3DGS [12], StreetGS [34] and SplatAD [9], as well as the generalizable method G3R [1]; and (2) r*econstruction with neural fixers*, including StreetCrafter [35] and Difix3D [30], where diffusion-based 2D image fixers refine novel views and the improvements are distilled back into the 3D representation through per-scene optimization. We train Difix3D on PandaSet following the official repository[1], and use the publicly released model checkpoint trained on PandaSet for StreetCrafter[2].

B. Experimental Results

a) Comparison to SoTA reconstruction methods: Table I reports the quantitative results. When comparing to standalone reconstruction approaches, *GNet* surpasses all baselines by a large margin in FID across nearly all lateral offsets while remaining competitive under original trajectories. This indicates that our generalizable enhancer can be efficiently adapted as a standalone reconstruction method, producing high-quality, robust 3D representations. In the *reconstruction*

[1]Difix3D official repository
[2]StreetCrafter official model weights

Fig. 6. Qualitative comparison to state-of-the-art 2D neural fixers.

TABLE II

3D ENHANCER PLUGS INTO DIFFERENT 3DGS-BASED METHODS.

Methods	FID@0m↓	FID@1m↓	FID@2m↓	FID@3m↓
3DGS [12]	61.74	82.45	117.05	154.21
+ *GenRe*	**57.32**	**69.62**	**85.02**	**99.69**
G3R [1]	70.62	80.48	104.75	132.46
+ *GenRe* (zero-shot)	65.75	74.86	89.22	100.67
+ *GenRe* (fine-tune)	**55.31**	**65.90**	**79.41**	**91.12**

with neural fixers setting, our full pipeline, *GenRe*, which integrates reconstruction with a diffusion-based 2D fixer and a generalizable 3D enhancer, achieves the best performance at challenging views while being substantially more efficient ($100\times$). It outperforms SoTA methods StreetCrafter and Difix3D especially under large extrapolations (*hard*). Qualitative results in Fig. 5 further show more complete reconstructions and fewer view-dependent artifacts for *GenRe* at extreme viewpoints.

b) Generalizable enhancements on varied 3DGS-based methods: We then demonstrate that our generalizable enhancer *GenRe* is generic and can easily plug into different 3DGS-based methods to enhance the representation. As shown in Table II, our enhancer, when trained on vanilla 3DGS, is able to correct the deficiencies in representation, and significantly boost the rendering quality at both original trajectory and extrapolated views. We also apply the pre-trained enhancer to directly refine the 3DGS representation produced by G3R in a zero-shot manner, achieving substantial improvements across novel viewpoints without retraining or altering the backbone, which demonstrates its versatility and robustness. A brief fine-tuning stage on the G3R representation further brings additional improvements.

c) 2D neural fixer comparison: Table III compares our 2D neural fixer (*FNet*) with the SoTA baselines. StreetCrafter-V [35], the fine-tuned video diffusion model in StreetCrafter, conditions on the reference view (*ref*) and the 3DGS rendering at the target view from colored LiDAR points (*lidar*). Difix3D [30] conditions on the rendered source image (*src*) and the reference image. In contrast, *FNet* leverages all three signals: *src* + *ref* + *lidar*. It achieves the lowest FID across all lateral shifts, demonstrating the advantage of pairing appearance cues with an explicit geometry-backed render. Qualitatively results in Fig. 6 show *FNet* produces less stylized, more photorealistic results than StreetCrafter-V and maintains stronger geometric consistency than Difix3D.

TABLE III

COMPARISON TO STATE-OF-THE-ART 2D NEURAL FIXERS.

Methods	**Input**			**FID ↓**			**Inference**
	src	*ref*	*lidar*	@1m	@2m	@3m	Time↓
StreetCrafter (V)		✓	✓	65.60	80.46	92.98	15.26 s/frame
Difix3D (Fixer)	✓	✓		59.07	75.33	91.11	**0.57 s/frame**
Ours (*FNet*)	✓	✓	✓	**50.12**	**65.55**	**80.27**	0.83 s/frame

TABLE IV

ABLATION STUDY ON *GenRe* COMPONENTS.

Methods	**Extrapolation FID ↓**				
	@1m	@2m	@3m	@4m	@5m
GNet	70.04	86.43	106.07	124.65	138.30
GNet→*FNet*→*GNet*	64.56	78.58	91.20	105.25	117.43
GNet→*FNet*→*ENet*	**60.69**	**74.50**	**88.04**	**102.73**	**114.19**

TABLE V

RUNTIME ANALYSIS ON DIFIX3D AND *GenRe* (IN MINUTES).

Methods	Reconstruction	Neural Fixer	Distillation	Total
Difix3D [30]	12.43	**0.383**	22.50	35.32
GenRe+	**0.967**	0.550	**1.25**	**2.77**

TABLE VI

RE-SIMULATION EVALUATION WITH DIFFERENT BEHAVIORS.

Methods	Brake	Accelerate	Change Lane	Swerve
3DGS [12]	225.40	234.65	132.28	129.90
Difix3D [30]	182.40	192.08	84.48	85.99
GenRe+	**152.38**	**143.03**	**77.69**	**78.49**

d) Ablation study: We decompose *GenRe* into *GNet* (base reconstruction), *FNet* (2D image fixer), and *ENet* (3D enhancer that distills fixes back into the scene representation) and quantify their contributions in Table IV. Correcting *GNet* artifacts with *FNet* and then rerunning reconstruction consistently lowers extrapolation FID, validating the effectiveness of the 2D fixer at novel views. Replacing the final *GNet* pass with *ENet* yields further gains in quality and efficiency, indicating distilling 2D fixes into 3D is more effective than rerunning reconstruction.

e) Runtime analysis: We provide a detailed runtime analysis compared to Difix3D in Table V. On average, *GenRe+* reconstructs a scene in 2.8 minutes, yielding a $10\times$ speedup over Difix3D (35.3 minutes). The speedup mainly comes from the distillation process, where Difix3D relies on costly per-scene reconstruction whereas our approach leverages a efficient generalizable 3D enhancer. Notably, our 3D enhancer *GenRe* (*FNet* → *ENet*) takes only 1.8 minutes to correct deficiencies in a 3D scene.

C. Downstream Applications

a) Realistic re-simulation with different behaviors: To test whether more robust reconstruction benefits downstream simulation, we emulate open-loop re-simulation by branching from the recorded trajectory and rendering along perturbed ego paths. We consider four behaviors: *braking*, *acceleration*, *lane change*, and *swerving* (changing to another lane and then

Fig. 7. Qualitative results on re-simulation in *swerving* behavior.

Fig. 8. *GenRe+* can support diverse variants for reactive log replay, such as dynamic actor removals, actors insertions, and actors manipulation.

back). Each rollout starts from a lateral offset of 3 m, and all synthetic scenarios are manually vetted for plausibility. We report image quality (FID) against baselines. As shown in Tab. VI and Fig. 7, *GenRe+* yields lower FID and provides higher quality rendering for all behaviors compared to baselines, showing its performance in downstream simulation. Moreover, Fig. 8 shows that *GenRe+* is able to support diverse variants for reactive log replay, such as dynamic actor removals, actors insertions, and actors manipulation, with high-quality, realistic rendering, which is desired for safety-critical evaluation of autonomy system.

b) Domain gap evaluation for perception: To evaluate how well the reconstruction methods can be used to test existing perception systems at challenging viewpoints, we measure domain gap via the *perception-agreement* metric under novel camera synthesis. Specifically, each scene is reconstructed using only the *front* camera, then rendered from *front-left* camera viewpoints. We run off-the-shelf object detection and instance segmentation models [32] on real and corresponding simulated renders. For each matched instance, we compute the AP, Recall, and IoU between predictions on real vs. simulated images (boxes and masks) and average over instances. This cross-camera agreement reflects how faithfully simulation preserves cues used by perception models under viewpoint transfer. As shown in Table VII, *GenRe+* achieves the highest agreement for both detection and instance segmentation when compared to other baselines. Fig 9 also illustrates these metrics, indicating *GenRe+*'s minimal domain gap and robust reconstruction.

c) 3D object detection with simulated data: Finally, to study whether simulation data improves 3D detection training, we augment PandaSet training set with renders generated by

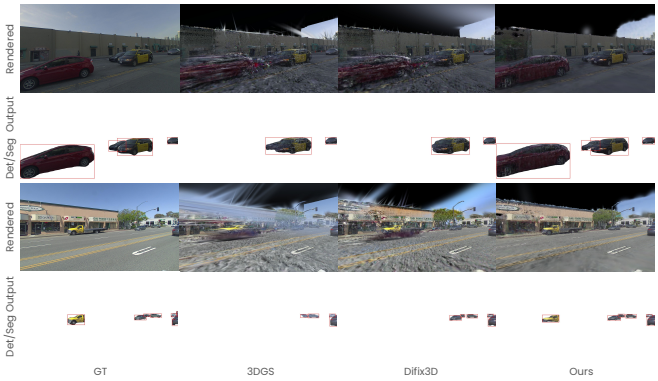

Fig. 9. *GenRe+* shows minimal detection and segmentation domain gap.

TABLE VII

DOWNSTREAM DOMAIN GAP EVALUATION.

Methods	Detection			Segmentation		
	AP ↑	Recall ↑	IoU ↑	AP ↑	Recall ↑	IoU ↑
3DGS [12]	0.560	0.376	0.505	0.558	0.375	0.501
Difix3D [30]	0.670	0.434	0.611	0.670	0.434	0.598
GenRe+	**0.785**	**0.607**	**0.728**	**0.768**	**0.596**	**0.723**

TABLE VIII

DOWNSTREAM TRAINING WITH DATA AUGMENTATION.

Methods	mAP↑	AP@1m↑	AP@2m↑	AP@4m↑
Real	0.256	0.085	0.247	0.437
Real + Sim (3DGS)	0.258	0.097	0.246	0.430
Real + Sim (ours)	**0.277**	**0.105**	**0.272**	**0.453**

3DGS and *GenRe+* at novel views: lateral shifts of 3 m while maintaining scene content unchanged. We retrain BEVformer-tiny [36] on the union of real and simulated images and evaluate on real held-out data. As reported in Table 9, *GenRe*-based augmentation yields clear improvements in average precision, whereas augmentation with vanilla 3DGS renders provides no noticeable gain. These findings indicate that high-fidelity, extrapolation-stable simulation is important for effective data augmentation in perception.

V. LIMITATIONS

GenRe has several limitations. First, although *GenRe* produces robust 3DGS representations, it still exhibits noticeable artifacts under extreme extrapolations (*e.g.*, bird-eye viewpoints in Fig. 2). Second, *GenRe* does not achieve 360° shape completion for background or objects [38]: geometry and appearance in heavily occluded or unobserved regions remain under-constrained. Addressing these challenges is an important direction towards fully unconstrained view synthesis and complete scene reconstruction.

VI. CONCLUSION

We introduce *GenRe*, a diffusion-guided generalizable enhancer for urban scene reconstruction. *GenRe* takes as input any pre-trained 3D Gaussian representation and fixes the deficiencies within 2 minutes in a generalizable manner. At the heart of *GenRe* are two modules: a one-step diffusion neural fixer that fixes degraded rendered images and a generalizable enhancer that predicts per-Gaussian residuals to enhance the

representation at novel views. Additionally, we also show that by adapting the enhancer for scene reconstruction from scratch, we obtain a generalizable reconstruction model that can robustly reconstruct the scene within 60s. Experiments show that *GenRe* outperforms existing methods in both quality and efficiency and benefits various downstream tasks, enabling robust and scalable sensor simulation for autonomous driving.

REFERENCES

[1] Yun Chen, Jingkang Wang, Ze Yang, Sivabalan Manivasagam, and Raquel Urtasun. G3r: Gradient guided generalizable reconstruction. In *ECCV*, 2025.

[2] Yurui Chen, Chun Gu, Junzhe Jiang, Xiatian Zhu, and Li Zhang. Periodic vibration gaussian: Dynamic urban scene reconstruction and real-time rendering. *arXiv*, 2023.

[3] Yutong Chen, Marko Mihajlovic, Xiyi Chen, Yiming Wang, Sergey Prokudin, and Siyu Tang. Splatformer: Point transformer for robust 3d gaussian splatting. In *ICLR*, 2025.

[4] Ziyu Chen, Jiawei Yang, Jiahui Huang, Riccardo de Lutio, Janick Martinez Esturo, Boris Ivanovic, Or Litany, Zan Gojcic, Sanja Fidler, Marco Pavone, et al. Omnire: Omni urban scene reconstruction. In *ICLR*, 2024.

[5] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, Eli VanderBilt, Aniruddha Kembhavi, Carl Vondrick, Georgia Gkioxari, Kiana Ehsani, Ludwig Schmidt, and Ali Farhadi. Objaverse-xl: A universe of 10m+ 3d objects. *arXiv*, 2023.

[6] Alexey Dosovitskiy, Germán Ros, Felipe Codevilla, Antonio M. López, and Vladlen Koltun. CARLA: an open urban driving simulator. In *CoRL*, 2017.

[7] Lue Fan, Hao Zhang, Qitai Wang, Hongsheng Li, and Zhaoxiang Zhang. Freesim: Toward free-viewpoint camera simulation in driving scenes. In *CVPR*, 2025.

[8] Jiazhe Guo, Yikang Ding, Xiwu Chen, Shuo Chen, Bohan Li, Yingshuang Zou, Xiaoyang Lyu, Feiyang Tan, Xiaojuan Qi, Zhiheng Li, and Hao Zhao. Dist-4d: Disentangled spatiotemporal diffusion with metric depth for 4d driving scene generation. *arXiv*, 2025.

[9] Georg Hess, Carl Lindström, Maryam Fatemi, Christoffer Petersson, and Lennart Svensson. Splatad: Real-time lidar and camera rendering with 3d gaussian splatting for autonomous driving. In *CVPR*, 2025.

[10] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022.

[11] Sungwon Hwang, Min-Jung Kim, Taewoong Kang, Jayeon Kang, and Jaegul Choo. Vegs: View extrapolation of urban scenes in 3d gaussian splatting using learned priors. In *ECCV*, 2024.

[12] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D gaussian splatting for real-time radiance field rendering. In *TOG*, 2023.

[13] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. In *CVPR*, 2024.

[14] Sivabalan Manivasagam, Shenlong Wang, Kelvin Wong, Wenyuan Zeng, Mikita Sazanovich, Shuhan Tan, Bin Yang, Wei-Chiu Ma, and Raquel Urtasun. Lidarsim: Realistic lidar simulation by leveraging the real world. In *CVPR*, 2020.

[15] Jiageng Mao, Boyi Li, Boris Ivanovic, Yuxiao Chen, Yan Wang, Yurong You, Chaowei Xiao, Danfei Xu, Marco Pavone, and Yue Wang. Dreamdrive: Generative 4d scene modeling from street view images. In *ICRA*, 2025.

[16] 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. In *ECCV*, 2020.

[17] 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. *arxiv*, 2024.

[18] Gaurav Parmar, Taesung Park, Srinivasa Narasimhan, and Jun-Yan Zhu. One-step image translation with text-to-image models. *arXiv*, 2024.

[19] Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu. On aliased resizing and surprising subtleties in gan evaluation. In *CVPR*, 2022.

[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 *CVPR*, 2025.

[21] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*, 2023.

[22] Xuanchi Ren, Yifan Lu, Hanxue Liang, Jay Zhangjie Wu, Huan Ling, Mike Chen, Francis Fidler, Sanja annd Williams, and Jiahui Huang. Scube: Instant large-scale scene reconstruction using voxsplats. In *NeurIPS*, 2024.

[23] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. *arXiv*, 2023.

[24] Shital Shah, Debadeepta Dey, Chris Lovett, and Ashish Kapoor. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and service robotics*, 2018.

[25] Adam Tonderski, Carl Lindström, Georg Hess, William Ljungbergh, Lennart Svensson, and Christoffer Petersson. NeuRAD: Neural rendering for autonomous driving. In *CVPR*, 2024.

[26] Jingkang Wang, Henry Che, Yun Chen, Ze Yang, Lily Goli, Sivabalan Manivasagam, and Raquel Urtasun. Flux4d: Flow-based unsupervised 4d reconstruction. In *NeurIPS*, 2025.

[27] Jingkang Wang, Sivabalan Manivasagam, Yun Chen, Ze Yang, Ioan Andrei Bârsan, Anqi Joyce Yang, Wei-Chiu Ma, and Raquel Urtasun. Cadsim: Robust and scalable in-the-wild 3d reconstruction for controllable sensor simulation. In *CoRL*, 2022.

[28] Jingkang Wang, Ava Pun, James Tu, Sivabalan Manivasagam, Abbas Sadat, Sergio Casas, Mengye Ren, and Raquel Urtasun. Advsim: Generating safety-critical scenarios for self-driving vehicles. In *CVPR*, 2021.

[29] Qitai Wang, Lue Fan, Yuqi Wang, Yuntao Chen, and Zhaoxiang Zhang. Freevs: Generative view synthesis on free driving trajectory. In *ICLR*, 2025.

[30] 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 *CVPR*, 2025.

[31] Rundi Wu, Ben Mildenhall, Philipp Henzler, Keunhong Park, Ruiqi Gao, Daniel Watson, Pratul P Srinivasan, Dor Verbin, Jonathan T Barron, Ben Poole, et al. Reconfusion: 3d reconstruction with diffusion priors. In *CVPR*, 2024.

[32] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

[33] Pengchuan Xiao, Zhenlei Shao, Steven Hao, Zishuo Zhang, Xiaolin Chai, Judy Jiao, Zesong Li, Jian Wu, Kai Sun, Kun Jiang, et al. Pandaset: Advanced sensor suite dataset for autonomous driving. In *ITSC*, 2021.

[34] 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 *ECCV*, 2024.

[35] 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 *CVPR*, 2025.

[36] Chenyu Yang, Yuntao Chen, Haofei Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Y. Qiao, Lewei Lu, Jie Zhou, and Jifeng Dai. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. *ArXiv*, 2022.

[37] 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 *CVPR*, 2023.

[38] Ze Yang, Jingkang Wang, Haowei Zhang, Sivabalan Manivasagam, Yun Chen, and Raquel Urtasun. Genassets: Generating in-the-wild 3d assets in latent space. In *CVPR*, 2025.

[39] 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. *arxiv*, 2025.

[40] Yingshuang Zou, Yikang Ding, Chuanrui Zhang, Jiazhe Guo, Bohan Li, Xiaoyang Lyu, Feiyang Tan, Xiaojuan Qi, and Haoqian Wang. Mudg: Taming multi-modal diffusion with gaussian splatting for urban scene reconstruction. *arXiv*, 2025.