ViSE: A Systematic Approach to Vision-Only Street-View Extrapolation

KaiyuanTan Yingying Shen Haiyang Sun[†] Bing Wang
Guang Chen Hangjun Ye[✉]
Xiaomi EV

Abstract

Realistic view extrapolation is critical for closed-loop simulation in autonomous driving, yet it remains a significant challenge for current Novel View Synthesis (NVS) methods, which often produce distorted and inconsistent images beyond the original trajectory. This report presents our winning solution which ctook first place in the RealADSim Workshop NVS track at ICCV 2025. To address the core challenges of street view extrapolation, we introduce a comprehensive four-stage pipeline. First, we employ a data-driven initialization strategy to generate a robust pseudo-LiDAR point cloud, avoiding local minima. Second, we inject strong geometric priors by modeling the road surface with a novel dimension-reduced SDF termed 2D-SDF. Third, we leverage a generative prior to create pseudo ground truth for extrapolated viewpoints, providing auxilary supervision. Finally, a data-driven adaptation network removes time-specific artifacts. On the RealADSim-NVS benchmark, our method achieves a final score of 0.441, ranking first among all participants.

1. Introduction

Validating autonomous driving algorithms in the real world faces high cost and significant safety risks, making high-fidelity simulation a critical enabling technology for advancing AD systems. Traditional game-engine-based simulators support closed-loop testing but suffer from a substantial domain gap and require labor-intensive, manual scene construction. Conversely, replay from real-world driving logs offers high visual fidelity but is limited to pre-recorded scenarios, unable to support interactive, closed-loop evaluation.

Novel View Synthesis (NVS) presents a compelling pathway to bridge this gap, aiming to construct interactive, 4D driving environments directly from real-world captures. However, a critical obstacle remains: the inherent sparsity of driving logs means that most NVS methods excel at view interpolation but struggle severely with view extrapolation. Under extrapolated viewpoints, current techniques based on volumetric primitives like NeRF[7] or 3D Gaussian Splatting (3DGS)[4] often exhibit major geometric distortions and a collapse in texture realism.

The RealADSim challenge was established to directly address this critical gap. It provides a set of multi-traversal driving logs with a quantitative metric designed for evaluating extrapolation performance.

In this report, we introduce a holistic pipeline that achieves robust and geometrically consistent street view extrapolation. Our solution systematically tackles the core challenges through four key contributions:

1. A robust, LiDAR-free initialization strategy that generates a vision-based pseudo point cloud to provide a strong geometric prior and prevent overfitting.
2. A dimension-reduced 2D Signed Distance Function (2D-SDF) that enforces a locally planar prior for road surfaces, improving geometric consistency under large viewpoint changes.
3. An iterative pseudo-ground-truth framework that leverages a generative prior to provide supervision for unobserved, extrapolated viewpoints, effectively repairing artifacts.
4. A data-driven time-invariance adaptation network that removes transient, time-specific features, ensuring robust performance across logs captured under different conditions.

On the RealADSim benchmark, our team achieves a final score of 0.441, ranking 1st among all participants.

2. Preliminary

2.1. 3D Gaussian Splatting

3D Gaussian Splatting (3DGS) [4] represents scenes using 3D Gaussian primitives. Each primitive is defined by a position $\mu \in \mathbb{R}^3$, covariance matrix $\mathbf{\Sigma} \in \mathbb{R}^{3\times3}$, and opacity $\alpha \in \mathbb{R}^+$: $\mathcal{G}(\mathbf{x}) = \alpha \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)\right)$ The covariance matrix is factorized into scaling and rotation components: $\mathbf{\Sigma} = \mathbf{RSS}^\top\mathbf{R}^\top$.

For rendering, 3D Gaussians are transformed to cam-

[†] Project leader [✉] Corresponding Author

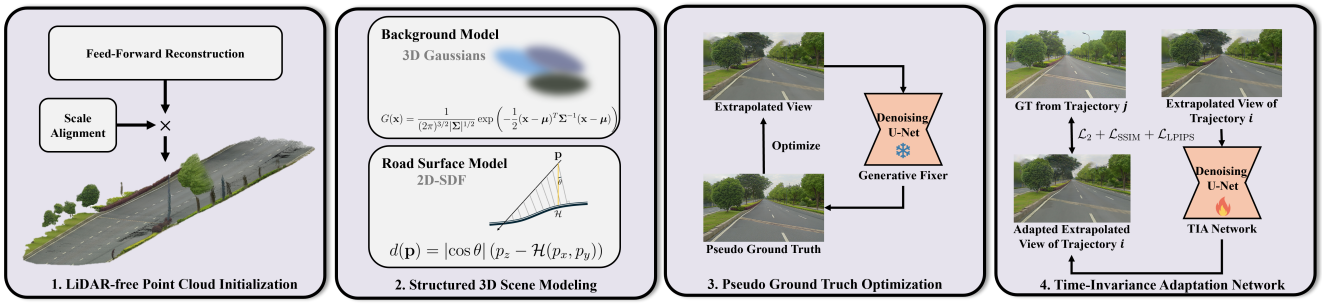

Figure 1. **Proposed Pipeline.** We introduce a four-stage framework for consistent novel-view synthesis. The process begins with (1) generating a robust, vision-based point cloud for initialization. This enables (2) detailed 3D reconstruction, where a 2D-SDF provides strong geometric priors for the road. To address remaining artifacts in unobserved areas, we employ (3) an iterative pseudo-GT strategy using a generative prior. Finally, (4) a time-invariance adaptation network ensures the output is temporally invariant across different driving logs.

era coordinates via world-to-camera matrix $\mathbf{W}$ and projected to 2D using local affine approximation $\mathbf{J}$: $\mathbf{\Sigma}' = \mathbf{JW\Sigma W}^\top\mathbf{J}^\top$. By discarding the third dimension, we obtain 2D Gaussians $\mathcal{G}^{2D}$ with covariance $\mathbf{\Sigma}^{2D}$. The final color is computed via front-to-back alpha blending:

$$\mathbf{c}(\mathbf{x}) = \sum_{k=1}^{K} \mathbf{c}_k, \alpha_k, \mathcal{G}^{2D} k(\mathbf{x}) \prod j = 1^{k-1}(1-\alpha_j, \mathcal{G}_j^{2D}(\mathbf{x})) \tag{1}$$

where $\mathbf{c}_k$ represents view-dependent color. This fully differentiable pipeline enables optimization through standard photometric losses.

2.2. Rendering Signed Distance Functions

A common approach to representing a surface is through the zero-level set of a Signed Distance Function (SDF), where the function's value indicates the distance to the nearest surface, with the sign denoting whether the point lies inside or outside the surface. NeuS [12] introduced a volume rendering technique for optimizing neural SDFs from posed images.

Given a ray $\mathbf{r}$, with sampled points $\mathbf{p}(t_i)_{i=0}^N$, the opacity α_i at location $\mathbf{p}(t_i)$ is computed as:

$$\alpha_i = \max\left(\frac{\Phi_s(d(\mathbf{p}(t_i))) - \Phi_s(d(\mathbf{p}(t_{i+1})))}{\Phi_s(d(\mathbf{p}(t_i)))}, 0\right) \tag{2}$$

Where $\Phi_s(x) = 1/(1+\exp(-sx))$ is the Sigmoid-based cumulative distribution function, with s controlling the sharpness of the transition and typically treated as a learnable parameter. Then, the color of each pixel is rendered using alpha-blending:

$$\mathbf{C}(\mathbf{r}) = \sum_{i=1}^{N} T_i \alpha_i \mathbf{c}_i, \quad T_i = \prod_{j=1}^{i-1}(1-\alpha_j) \tag{3}$$

3. Method

Problem Formulation Given a driving log comprising images $I_{i i=1}^N$ and their associated ego-vehicle poses $P_{i i=1}^N$, our objective is to learn a 3D scene representation that maps a 6-DoF pose to an image, $\mathcal{S} : SE(3) \to \mathbb{R}^{H\times W\times 3}$. This work focuses on the challenging task of synthesizing novel views from extrapolated viewpoints beyond the original trajectory.

We address four core challenges through a systematic, four-stage pipeline, illustrated in Figure 1.

3.1. 3D Scene Initialization without LiDAR

The absence of LiDAR data poses a significant challenge for initializing 3D Gaussians. Conventional Structure-from-Motion (SfM) [10] points, often combined with random initialization, typically yield poor geometry due to the sparsity and limited viewing directions in driving logs. This frequently leads to convergence in a local minimum that fails under viewpoint extrapolation.

To overcome this, we generate a vision-based pseudo-LiDAR point cloud for robust initialization. We adopt VGGT [11] for its favorable balance of quality and simplicity. As this method lacks absolute scale, we recover it by aligning the predicted camera poses with the ground-truth poses. The recovered scale is then multiplied to the predicted depth maps, which are subsequently unprojected and aggregated using the ground-truth poses to form a unified point cloud.

Although this pseudo-LiDAR point cloud can be noisy, it provides critical geometric initialization that enables faster 3DGS convergence, recovers finer details, and prevents implausible geometric distortions during extrapolation.

3.2. Geometry-Aware 3D Scene Reconstruction

2D-SDF for Road Surface Representation Accurately modeling the road surface is paramount for driving simula-

tion, as it contains critical visual elements like road signs and lane markings. However, this is challenging due to weak textures and large viewpoint changes in extrapolated scenarios. Methods based on 3DGS [4] and NeRF [7] often render severe distortions for the road surface under extrapolated trajectories.

While several methods [5, 19, 20] explored constraining 3D Gaussians for a flat road surface, we consider an alternative representation. We introduce a novel 2D Signed Distance Function (2D-SDF) that incorporates the strong geometric prior that road surfaces are *locally planar* and exhibit *globally smooth slope transitions*. This allows us to compress the representation from a general 3D SDF to two efficient 2D fields defined over the horizontal plane, modeling surface height and local slope, respectively. Under the local planar assumption, the signed distance from a 3D point $\mathbf{p}$ simplifies to:

$$d(\mathbf{p}) = |\cos\theta| \left(p_z - \mathcal{H}(p_x, p_y)\right), \tag{4}$$

where $\mathcal{H} : \mathbb{R}^2 \to \mathbb{R}$ represents the surface height at horizontal coordinates (p_x, p_y), and θ is the angle between the surface normal and the vertical axis. Both $\mathcal{H}$ and $|\cos\theta|$ depend solely on (p_x, p_y) and are parameterized using neural networks. This leads to our 2D-SDF formulation:

$$\begin{cases} d(\mathbf{p}) = \text{MLP}_{\text{slope}}(\mathbf{p})\,[p_z - \text{MLP}_{\text{elevation}}(\mathbf{p})] \\ \mathbf{c}(\mathbf{p}, \mathbf{v}) = \text{MLP}_{\text{color}}(\mathbf{p}, \mathcal{F}(\mathbf{v})) \end{cases} \tag{5}$$

Constrained to produce a smooth surface, the 2D-SDF ensures strong geometric consistency under extrapolation. Note that This dimension-reduced SDF also improves optimization efficiency over vanilla SDFs, converging in approximately 15 minutes per scene. Following previous SDF-based rendering methods (Section 2.2), the 2D-SDF is optimized alongside the rest of the scene using reconstruction losses alone.

Sky Modeling and Composition We represent all above-ground objects using 3D Gaussians, following prior work [1, 2, 15, 21]. The sky, being infinitely far away, is represented by an optimizable environment texture map, as in [1, 2]. To composite these representations, we assume a layering order where the road surface lies behind all above-ground objects, and the sky lies behind the road. The final image is rendered as:

$$I = O_{\text{gs}} I_{\text{gs}} + (1 - O_{\text{gs}}) O_{\text{road}} I_{\text{road}} + (1 - O_{\text{gs}})(1 - O_{\text{road}}) I_{\text{sky}}, \tag{6}$$

where $I_{\text{gs}}, I_{\text{road}}, I_{\text{sky}}$ are the rendered colors from the 3D Gaussians, the road model, and the sky environment map, respectively, and $O_{\text{gs}}, O_{\text{road}}$ are their accumulated opacities.

We use MLP to denote networks for brevity; in practice, we use multi-level hash grids as in [6].

3.3. Iterative Pseudo Ground Truth Generation

While our 2D-SDF effectively regularizes the road, non-road objects (e.g., vegetation, curbs, buildings) lack a universal geometric prior. These objects often exhibit severe distortions or floating artifacts under extrapolation due to insufficient observations.

To provide explicit supervision for these unobserved regions, we employ an iterative pseudo ground truth (GT) generation strategy. Following recent work [3, 8, 13, 16–19], we leverage a pre-trained diffusion model as a generative fixer. We adopt the off-the-shelf Difix3D+ [14] model for this purpose. At scheduled intervals during training, we synthesize pseudo-GT images for progressively extrapolated camera poses. These poses are generated by interpolating between the nearest training views and the target extrapolated test views. We first render the current 3D scene at these target poses and then use the generative fixer to refine the noisy renderings into pseudo-GT images, which are added back to the training set (see Figure 1, Part 3).

The pseudo-GT supervision loss is defined as:

$$\mathcal{L}_{\text{pseudo}} = \lambda_{\text{LPIPS}} \cdot \mathcal{L}_{\text{LPIPS}} + \lambda_{\text{L1}} \cdot \mathcal{L}_{\text{L1}}. \tag{7}$$

A naïve application of this supervision introduces a trade-off: while it effectively reduces floating artifacts (improving LPIPS), it can also hallucinate unrealistic details such as over-saturated textures or spurious objects, which harm pixel-level metrics (PSNR/SSIM). To balance this, we significantly down-weight λ_{L1} relative to λ_{LPIPS}. This prioritizes perceptual quality while mitigating the impact of pixel-level inaccuracies in the pseudo-GT.

3.4. Time-Invariance Adaptation Network

A unique challenge in this competition is the need for extrapolation not only in space but also in time. Different lanes were recorded at different times, leading to variations in illumination, weather, and transient objects (e.g., puddles, shadows). A model that naïvely memorizes these time-specific details produces inconsistent results when rendering a scene in the style of a different log. We found that factoring out these time-varying artifacts is critical for achieving realistic and consistent view synthesis. For instance, time-specific elements like reflections on asphalt, puddles, and cloud patterns should be suppressed, while dynamic textures like foliage should be smoothed to avoid hallucinated details.

To achieve temporal invariance, we introduce a lightweight Time-Invariance Adaptation Network (TIA-Net). This network acts as a post-processing module that learns to remove time-specific artifacts from a rendered image:

$$I' = \text{TIA}_\theta(I), \tag{8}$$

where θ represents the learnable parameters.

We train TIA-Net in a data-driven manner using multi-traversal data from the public portion of the Para-Lane [9] dataset. Specifically, we reconstruct a 3D scene from one trajectory (e.g. LOG_i) using our full pipeline. We then render this scene using the camera poses from a different trajectory, LOG_j, which was captured at a different time. The rendered images are passed through TIA-Net, and the output is compared to the real images from LOG_j.using photometric losses.

This objective forces TIA-Net to learn a mapping that effectively removes time-dependent features (e.g., specific lighting, reflections) while preserving the underlying scene structure, resulting in a consistent and generalizable appearance.

We train TIA-Net by fine-tuning the Difix3D+ [14] model, as it provides a fast, one-step image-to-image refinement framework that is well-suited for this adaptation task.

4. Experiments

Rank	Team	PSNR↑	SSIM↑	LPIPS↓
1	XiaomiEV Team	**18.228**	**0.514**	**0.288**
2	Qualcomm AI Research	17.887	0.492	0.289
3	R2D2	18.009	0.496	0.361
4	MeowAndDoggy	17.857	0.490	0.371
5	aowei	16.72	0.484	0.401

Table 1. **Results on the RealADSim-NVS Benchmark.** We only copy results for the top-5 teams. The full results are available at https://huggingface.co/spaces/XDimLab/ICCV2025-RealADSim-NVS.

4.1. Main Results

We present results on the RealADSim-NVS benchmark, which is specifically designed to assess the challenging task of novel view extrapolation in autonomous driving scenarios. As shown in Table 1, our full pipeline, designated as **XiaomiEV Team**, achieves state-of-the-art performance, ranking first among all participants. This demonstrates the effectiveness of our structured vision-based pipeline in generating realistic and geometrically consistent images at extrapolated views.

4.2. Ablation Study

To validate the contribution of each component in our pipeline, we conduct a comprehensive ablation study. The results are summarized in Table 2

② **2D-SDF Road Prior:** Introducing a strong geometric prior for the road surface via our 2D-SDF model provides a crucial foundation. It preserves PSNR/SSIM while significantly improving LPIPS (0.500 vs. 0.513). This indicates that the model successfully mitigates geometric distortions under extrapolation, leading to more structurally coherent and realistic outputs.

③ **Pseudo-LiDAR Initialization:** Initializing 3DGS with a vision-based pseudo point cloud yields a further boost across all metrics. This step alleviates overfitting to the training viewpoints and helps the model escape poor local minima, resulting in more accurate geometry for both distant scenes and near-field objects.

④ **Pseudo GT with Generative Prior:** This stage delivers a substantial leap in performance, most notably a 20% relative improvement in LPIPS (0.396 vs. 0.47). By providing explicit supervision for unobserved regions, the generative prior effectively "inpaints" plausible scene content and eliminates floating artifacts, dramatically enhancing the visual plausibility of extrapolated views.

⑤ **TIA Network:** The final addition of our Time-Invariance Adaptation Network achieves highest improvement. By learning to remove transient, time-specific elements (e.g., illumination changes, shadows, and moving clouds), the TIA network produces a stable, time-agnostic scene representation. This is critical for robust performance under the spatio-temporal extrapolation required by the benchmark.

Exp. ID	Method	PSNR↑	SSIM↑	LPIPS↓
①	baseline	16.154	0.480	0.513
②	① + 2D-SDF	16.170	0.484	0.500
③	② + Pseudo LiDAR	16.369	0.493	0.47
④	③ + Pseudo GT	17.286	0.507	0.396
⑤	④ + TIA Network	**18.228**	**0.514**	**0.288**

Table 2. **Ablation of Key Steps**

5. Conclusion

This report presents our solution for robust street view extrapolation that earned 1st place in the RealADSim-NVS challenge. Our method systematically addresses key challenges through LiDAR-free initialization, a novel 2D-SDF road prior, generative pseudo-GT supervision, and temporal adaptation. Extensive experiments validate each component's contribution to our state-of-the-art performance.

References

[1] Yurui Chen, Chun Gu, Junzhe Jiang, Xiatian Zhu, and Li Zhang. Periodic vibration gaussian: Dynamic urban scene reconstruction and real-time rendering. *arXiv:2311.18561*, 2023. 3

[2] 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 *The Thirteenth International Conference on Learning Representations*, 2025. 3

[3] Lue Fan, Hao Zhang, Qitai Wang, Hongsheng Li, and Zhaoxiang Zhang. Freesim: Toward free-viewpoint camera simulation in driving scenes, 2024. 3

[4] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42 (4), 2023. 1, 3

[5] Mustafa Khan, Hamidreza Fazlali, Dhruv Sharma, Tongtong Cao, Dongfeng Bai, Yuan Ren, and Bingbing Liu. Autosplat: Constrained gaussian splatting for autonomous driving scene reconstruction, 2024. 3

[6] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

[7] 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. 1, 3

[8] 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. 2024. 3

[9] Ziqian Ni, Sicong Du, Zhenghua Hou, Chenming Wu, and Sheng Yang. Para-lane: Multi-lane dataset registering parallel scans for benchmarking novel view synthesis, 2025. 4

[10] Johannes L. Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, 2016. 2

[11] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 2

[12] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *NeurIPS*, 2021. 2

[13] Qitai Wang, Lue Fan, Yuqi Wang, Yuntao Chen, and Zhaoxiang Zhang. Freevs: Generative view synthesis on free driving trajectory. *arXiv preprint arXiv:2410.18079*, 2024. 3

[14] 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 Computer Vision and Pattern Recognition Conference*, pages 26024–26035, 2025. 3, 4

[15] 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. 3

[16] 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)*, 2025. 3

[17] Zeyu Yang, Zijie Pan, Yuankun Yang, Xiatian Zhu, and Li Zhang. Drivex: Driving view synthesis on free-form trajectories with generative prior. In *ICCV*, 2025.

[18] 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. 2024.

[19] 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. 2025. 3

[20] Hongyu Zhou, Longzhong Lin, Jiabao Wang, Yichong Lu, Dongfeng Bai, Bingbing Liu, Yue Wang, Andreas Geiger, and Yiyi Liao. Hugsim: A real-time, photo-realistic and closed-loop simulator for autonomous driving. *arXiv preprint arXiv:2412.01718*, 2024. 3

[21] Hongyu Zhou, Jiahao Shao, Lu Xu, Dongfeng Bai, Weichao Qiu, Bingbing Liu, Yue Wang, Andreas Geiger, and Yiyi Liao. Hugs: Holistic urban 3d scene understanding via gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21336–21345, 2024. 3