Spectral-Geometric Neural Fields for Pose-Free LiDAR View Synthesis

Supplementary Material

This supplement contains the following contents:
- More ablation studies and analysis.
- More experimental results.
- More implementation details.

1. More Ablation Studies and Analysis

1.1. Efficacy of Cross-frame Consistency

In Sec. 3.4, line 318 of the main paper, we introduce the Cross-frame Consistency (CFC) strategy, which leverages discriminative supervision to enhance reconstruction quality. Here, we provide additional experiments to demonstrate its efficacy. We evaluate our method with and without the CFC strategy on the low-frequency KITTI-360 [7, 12] dataset in Table 1 . Fig. 1 provides qualitative comparisons of the reconstructed depth maps in two consecutive frames.

SG-NLF (w/o CFC) relies solely on pixel-level 2D depth image supervision [11, 13], as shown in Table 1, resulting in a decrease in metrics. While 2D depth image supervision can oversee photometric alignment for each frame, it focuses more on pixel-wise improvements and ignores local structural information, leading to geometric breaks (*e.g.*, incomplete telephone poles) and topological errors (*e.g.*, incorrectly connected leaves), as shown by the rectanglar boxes in Fig. 1. To overcome these issues, our CFC strategy introduces cross-frame structural discrimination. The discriminator is trained to identify geometric inconsistencies between aligned frames. When the model generates structurally inconsistent image pairs (*e.g.*, poles not present in the predicted images but present in the ground truth cross-frame images), the discriminator classifies them as false. The resulting consistency loss $\mathcal{L}_{con}$ forces NeRF to reconstruct more consistent geometry in these regions, shown in the third row of Fig. 1. These results demonstrate the effectiveness of our hybrid representation and cross-frame strategy in achieving robust reconstruction.

1.2. Ablation on Sampling Points

We conduct ablation experiments on the number of sampling points (M) as described in Sec. 3.2 of main paper. As shown in Fig. 2, we evaluate the reconstruction performance of depth RMSE and depth PSNR by varying M from 512 to 8192. Reconstruction quality (depth RMSE and PSNR) is compromised with only $M = 512$ points due to insufficient geometric coverage. Performance improves as M grows and stabilizes beyond 4096, indicating marginal gains from further increasing sampling density. Finally, we select the sampling points of 4096, which can adequately meet the requirements of LiDAR scene reconstruction.

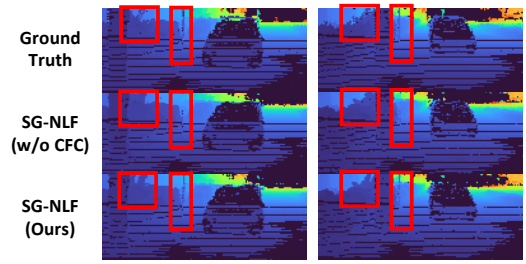

Figure 1. **Qualitative ablation of cross-frame consistency strategy.** We compare the reconstruction results of SG-NLF (w/o CFC) and SG-NLF (Ours). SG-NLF (w/o CFC) relies solely on pixel-level 2D depth image supervision. As shown in the rectangular boxes, while 2D depth image supervision ensures photometric alignment for each frame, it fails to preserve geometric details, resulting in incomplete telephone poles and incorrect leaf connections. By introducing cross-frame discrimination, our method learns cross-frame structure consistency representations, thus obtaining more complete and continuous reconstruction results.

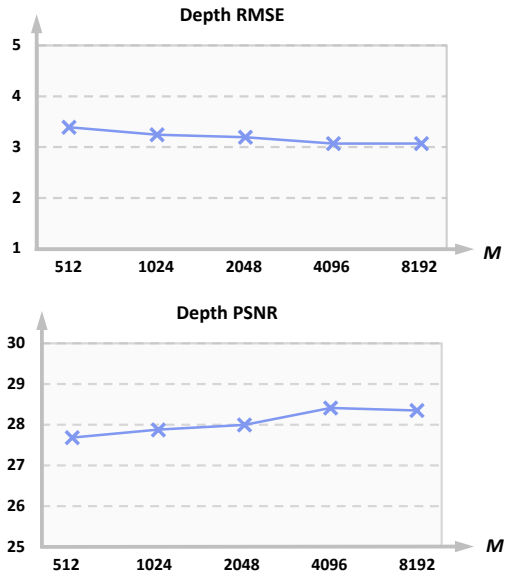

Figure 2. **Ablation experiments on the number of sampling points.** We evaluate the reconstruction performance of depth RMSE and depth PSNR by varying M from 512 to 8192. Depth RMSE and PSNR are compromised with only $M = 512$ points due to insufficient geometric coverage. Performance improves as M grows and stabilizes beyond 4096.

1.3. Comparison with Gaussian Splatting

SG-NLF also outperforms Gaussian-Splatting-based model GS-LiDAR [5] in the quality of LiDAR view synthesis. As

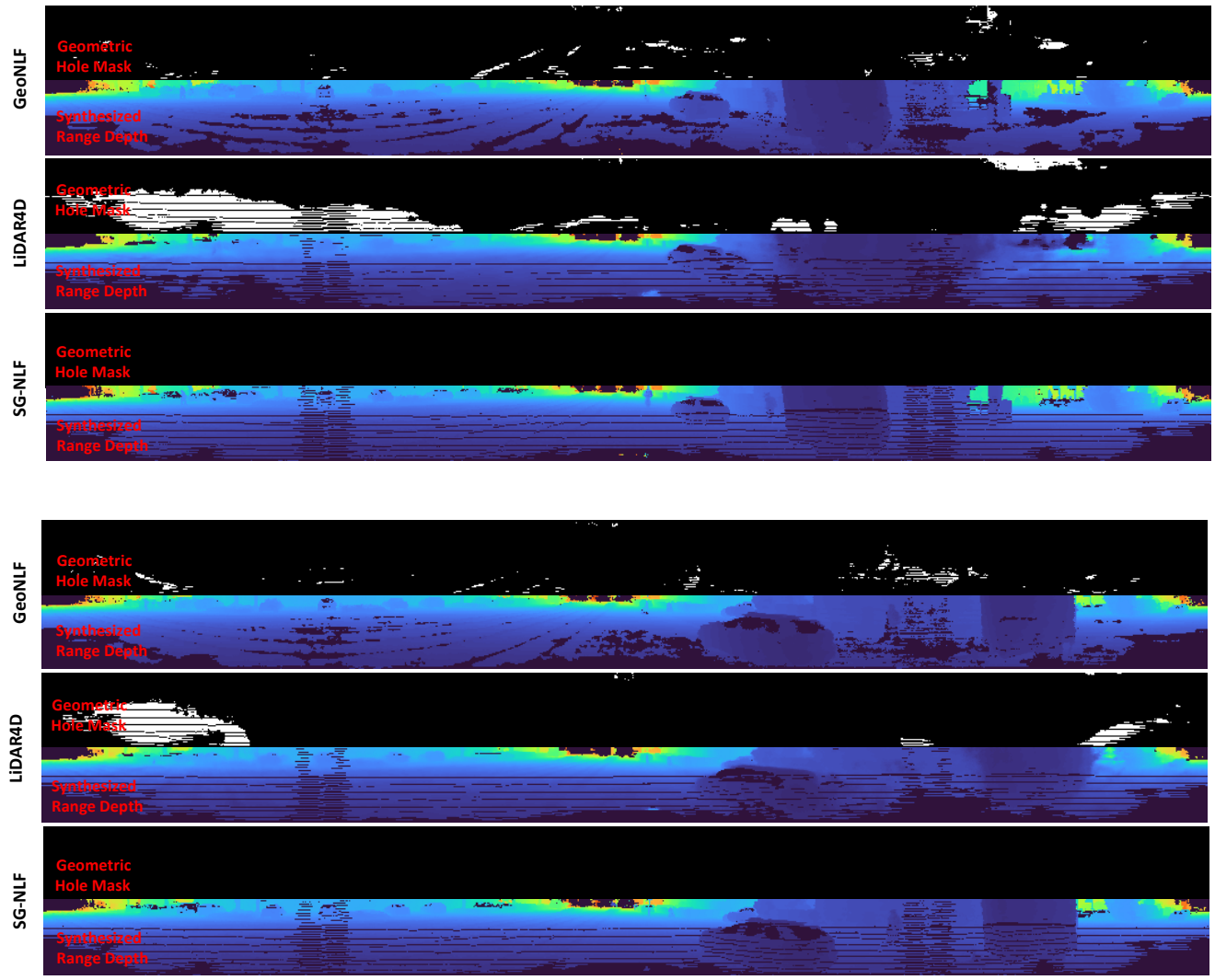

Figure 3. **Geometric Consistency Comparison.** The geometric hole mask is generated by comparing rendered opacity with ground truth LiDAR measurements. White regions highlight geometric holes where methods fail to reconstruct geometry. GeoNLF [12] and LiDAR4D [13] exhibit significant geometric holes, resulting in blurred details and structural artifacts in the synthesized range depth maps. In contrast, our method achieves geometrically consistent reconstruction results with clear boundary, demonstrating our effectiveness.

Table 1. **Ablate on cross-frame consistency strategy (CFC).** We evaluate our method with and without the CFC strategy on the low-frequency KITTI-360 [7, 12] dataset.

Method	2D Range	CFC	Point Cloud CD↓	Depth PSNR↑	Intensity PSNR↑
SG-NLF(w/o CFC)	✗	✔	0.1735	27.0587	18.2443
SG-NLF(Ours)	✔	✔	**0.1695**	**28.7068**	**19.2652**

discussed in the GS-LiDAR paper, current methods (*e.g.*, LiDAR-NeRF, GeoNLF, LiDAR4D, etc.) adopt NeRF-based frameworks owing to their "effective implicit representation and high-quality volumetric rendering". On the other hand, "3D Gaussian splatting struggles with geometric modeling and tends to overfit sparse views". GS-LiDAR represents an initial effort to apply Gaussian primitives to this task. However, limited by the above challenges, as shown in the table below, our method surpasses GS-LiDAR by 11.2% (Depth RMSE) and 7.9% (Intensity RMSE) on KITTI-360 dataset.

Table 2. **Comparisons on the standard-frequency KITTI-360 [7] with challenging dynamic scenes [13].** We follow the setting of LiDAR4D [13] and compare SG-NLF with state-of-the-art pose-dependent methods [3, 6, 9, 11, 13] and pose-free model [12]. We color the best results as red and the second-best as orange.

Method	Point Cloud		Depth					Intensity				
	CD↓	F-score↑	RMSE↓	MedAE↓	LPIPS↓	SSIM↑	PSNR↑	RMSE↓	MedAE↓	LPIPS↓	SSIM↑	PSNR↑
LiDARsim [9]	3.2228	0.7157	6.9153	0.1279	0.2926	0.6342	21.4608	0.1666	0.0569	0.3276	0.3502	15.5853
PCGen [6]	0.4636	0.8023	5.6583	0.2040	0.5391	0.4903	23.1675	0.1970	0.0763	0.5926	0.1351	14.1181
NKSR [3]	1.8982	0.6855	5.8403	0.0996	0.2752	0.6409	23.0368	0.1742	0.0590	0.3337	0.3517	15.2081
LiDAR-NeRF [11]	0.1438	0.9091	4.1753	0.0566	0.2797	0.6568	25.9878	0.1404	0.0443	0.3135	0.3831	17.1549
LiDAR4D [13]	0.1089	0.9272	3.5256	0.0404	0.1051	0.7647	27.4767	0.1195	0.0327	0.1845	0.5304	18.5561
GeoNLF [12]	0.2872	0.8766	3.0218	0.0467	0.1343	0.8002	28.4861	0.1286	0.0402	0.2751	0.5025	17.8174
SG-NLF (Ours)	0.1047	0.9176	2.7705	0.0492	0.1007	0.8351	29.6267	0.1069	0.0320	0.2343	0.5894	19.4302

Table 3. **Comparisons on the standard-frequency KITTI-360 [7].**

Method	Point Cloud	Depth		Intensity	
	CD↓	RMSE↓	PSNR↑	RMSE↓	PSNR↑
GS-LiDAR (ICLR 2025)	0.1085	3.1212	28.4381	0.1161	18.7482
SG-NLF (Ours)	**0.1047**	**2.7705**	**29.6267**	**0.1069**	**19.4302**

2. More Experimental Results

2.1. Geometric Consistency Comparison

As illustrated in Fig. 2 and line 061 of our submitted manuscript, previous approaches [4, 11–13] typically rely on geometric interpolation for neural field rendering. However, due to the sparsity and irregularity of LiDAR data, such interpolated features often fail to reconstruct continuous surfaces, leading to geometric inconsistency. We provide further comparative visualization in Fig. 3. We obtain the geometric hole mask by comparing the opacity of the model rendering with the actual LiDAR measurements. The white regions show areas where these methods fail to reconstruct the geometry. As shown in Fig. 3, geometric interpolation-based methods such as GeoNLF [12] and LiDAR4D [13] exhibit significant geometric holes, resulting in blurred details and structural artifacts in the synthesized range depth maps. In contrast, our method achieves geometrically consistent reconstruction results with clear boundaries, demonstrating the effectiveness of our SG-NLF.

2.2. More Quantitative Results

In Sec. 4.2 and Table 3 of the submitted main manuscript, we evaluate SG-NLF on the standard-frequency KITTI-360 [7] dataset, demonstrating high-quality reconstruction and synthesis. Here, we extend our evaluation to challenging dynamic scenes, using the same settings as LiDAR4D [13] on the standard-frequency KITTI-360 dataset [7]. LiDAR4D incorporates a scene flow prior to enhance reconstruction consistency in dynamic environments. As shown in Table 2, even without employing the time prior for scene flow, SG-NLF improves depth SSIM, depth PSNR, intensity SSIM, and intensity PSNR by over 9.2%, 7.8%, 11.1%, and 4.7% than LiDAR4D [13], demonstrating the robustness and generalization of our framework.

2.3. SG-NLF with Ground Truth Pose Input

The advantages of our method compared to state-of-the-art pose-dependent methods [6, 9, 13] and pose-free models [2, 8, 12] are demonstrated through Figures 4 to 6 and Tables 1 to 4 in our submitted manuscript. As analyzed in Sec. 4.2, our pose-independent SG-NLF achieves competitive performance without relying on the ground truth pose input, even outperforming pose-dependent models in challenging scenarios. To further demonstrate the efficacy of our hybrid representation, we conduct additional experiments with ground truth poses as input. Table 4 and Table 5 provide comparative results on the low-frequency KITTI-360 [7, 12] and low-frequency nuScenes [1, 12] datasets. When ground truth poses are provided, our method achieves higher reconstruction accuracy, significantly surpassing the leading pose-dependent method LiDAR4D, demonstrating the effectiveness of our hybrid spectral-geometry design.

2.4. More Qualitative Results

We show more visual comparisons in Fig. 4, 5, and 6. Fig. 4 provides qualitative comparisons for LiDAR depth reconstruction and synthesis, while Fig. 5 presents comparisons for LiDAR intensity reconstruction and synthesis. We further visualize the reconstructed point clouds in Fig. 6.

2.5. Model Efficiency

In Fig. 1 of the submitted main manuscript, we present the average rendering time per sequence on low-frequency KITTI-360 [7, 12] dataset. To further evaluate the model efficiency, we compare the model parameters of different methods [11–13] in Table 6. Our evaluation follows the standard protocol using official implementations of competing methods [11–13]. As shown in Table 6, LiDAR4D [13] exhibits significant computational overhead with 53.1M parameters and 242 ms runtime, primarily due to its complex planar-grid representation that introduces substantial memory. In contrast, our SG-NLF achieves a balance between performance and efficiency, with only 18.4M parameters

Table 4. **More comparisons with state-of-the-art methods on KITTI-360 [7] dataset with a low-frequency setting [12].** We conduct additional experiments with ground truth poses (GT Pose) as input. Compared to state-of-the-art pose-dependent methods [3, 6, 9, 13], our SG-NLF achieves better reconstruction performance when providing ground truth poses.

Type	Method	Point Cloud		Depth					Intensity				
GT Pose		CD↓	F-score↑	RMSE↓	MedAE↓	LPIPS↓	SSIM↑	PSNR↑	RMSE↓	MedAE↓	LPIPS↓	SSIM↑	PSNR↑
✔	LiDARsim [9]	11.0426	0.5975	10.1994	1.3881	0.5588	0.3888	17.9423	0.2089	0.1254	0.6463	0.0824	13.6127
✔	PCGen [6]	1.0356	0.7862	7.5672	0.7066	0.5333	0.3744	20.6219	0.2139	0.1063	0.5930	0.1070	13.4243
✔	NKSR [3]	1.3989	0.6753	8.8172	1.5896	0.5821	0.4135	19.2937	0.2133	0.1381	0.6099	0.0707	13.4516
✔	LiDAR4D [13]	0.2760	0.8843	4.7303	0.0785	0.3368	0.6197	24.7282	0.1459	0.0524	0.3883	0.3406	16.9512
✗	BARF-LN [8, 11]	3.1001	0.6156	7.5767	2.0583	0.5779	0.2834	22.5759	0.2121	0.1575	0.7121	0.1468	11.9778
✗	HASH-LN [2, 11]	2.6913	0.6082	6.3005	2.1686	0.5176	0.3752	22.6196	0.2404	0.1502	0.6508	0.1602	12.9286
✗	GeoTrans-LN [10, 11]	0.5753	0.8116	4.4291	0.2023	0.3896	0.5330	25.6137	0.2709	0.1589	0.5578	0.2578	12.9707
✗	GeoNLF [12]	0.2363	0.9178	4.0293	0.1009	0.3900	0.6272	25.2758	0.1495	0.1525	0.5379	0.3165	16.5813
✗	SG-NLF(Ours)	0.1695	0.9191	2.9514	0.0544	0.0701	0.9270	28.7068	0.1089	0.0368	0.2026	0.5751	19.2652
✔	SG-NLF(w/ pose)	**0.1180**	**0.9317**	**2.1510**	**0.0312**	**0.0424**	**0.9532**	**31.4257**	**0.1029**	**0.0363**	**0.1840**	**0.6185**	**19.7479**

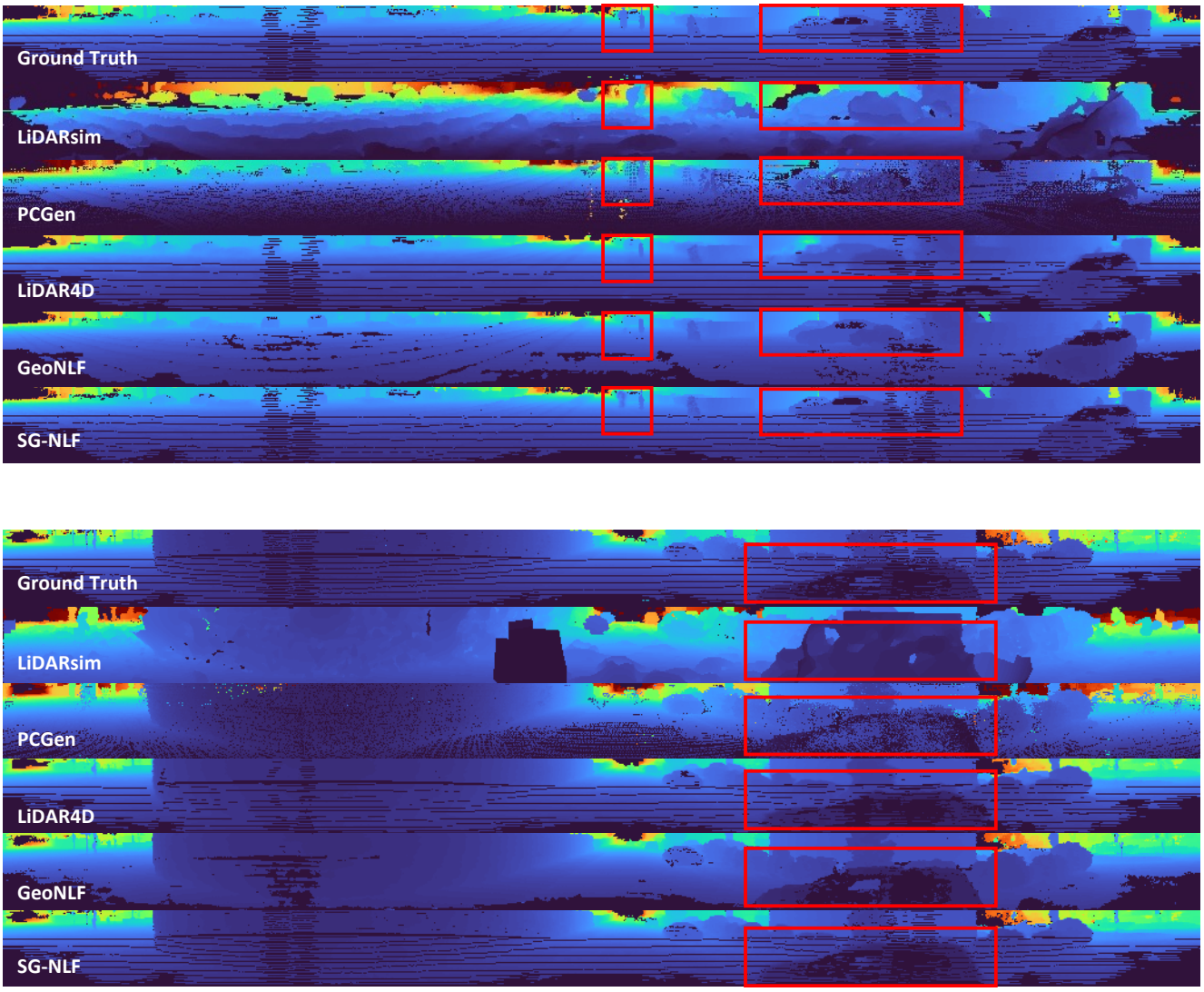

Figure 4. **Qualitative comparisons for LiDAR range depth reconstruction and synthesis.** Both pose-dependent [3, 6, 13] and pose-free methods [12] are compared. Regions with obvious differences are highlighted in the rectangular boxes.

Table 5. **More comparisons with state-of-the-art methods on nuScenes [1] dataset with a low-frequency setting [12].** We conduct additional experiments with ground truth poses (GT Pose) as input. Compared to state-of-the-art pose-dependent methods [3, 6, 9, 13], our SG-NLF achieves better reconstruction performance when providing ground truth poses.

Type	Method	Point Cloud		Depth					Intensity				
GT Pose		CD↓	F-score↑	RMSE↓	MedAE↓	LPIPS↓	SSIM↑	PSNR↑	RMSE↓	MedAE↓	LPIPS↓	SSIM↑	PSNR↑
✔	LiDARsim [9]	16.7623	0.4308	12.3483	1.9971	0.3125	0.3889	16.2553	0.0858	0.0355	0.1713	0.2897	21.3989
✔	PCGen [6]	2.2608	0.6139	12.6586	0.7055	0.2365	0.4389	12.6586	0.0865	0.0235	0.1530	0.3707	21.3643
✔	NKSR [3]	2.4444	0.5955	13.2809	3.0977	0.3957	0.3373	15.6226	0.0910	0.0409	0.2079	0.2627	20.8796
✔	LiDAR4D [13]	0.5668	0.7444	11.1964	0.0847	0.0685	0.5983	17.0920	0.0599	0.0180	0.0639	0.5401	24.4753
✗	BARF-LN [8, 11]	1.2695	0.7589	8.2414	0.1123	0.1432	0.6856	20.8900	0.3920	0.0144	0.1023	0.6119	26.2330
✗	HASH-LN [2, 11]	0.9691	0.8011	7.8353	0.0441	0.1190	0.6543	20.6244	0.0459	0.0135	0.0954	0.6279	26.8870
✗	GeoTrans-LN [10, 11]	4.1587	0.2993	10.7899	2.1529	0.1445	0.3671	17.5885	0.0679	0.0256	0.1149	0.3476	23.6211
✗	GeoNLF [12]	0.2408	0.8647	5.8208	0.0281	0.0727	0.7746	22.9472	0.0378	0.0100	0.0774	0.7368	28.6078
✗	SG-NLF(Ours)	0.1545	0.9097	3.0706	0.0278	0.0191	0.9398	28.4094	0.0299	0.0078	0.0349	0.8679	30.4987
✔	SG-NLF(w/ pose)	**0.1078**	**0.9354**	**2.2764**	**0.0238**	**0.0107**	**0.9467**	**31.2547**	**0.0235**	**0.0056**	**0.0327**	**0.8964**	**31.0764**

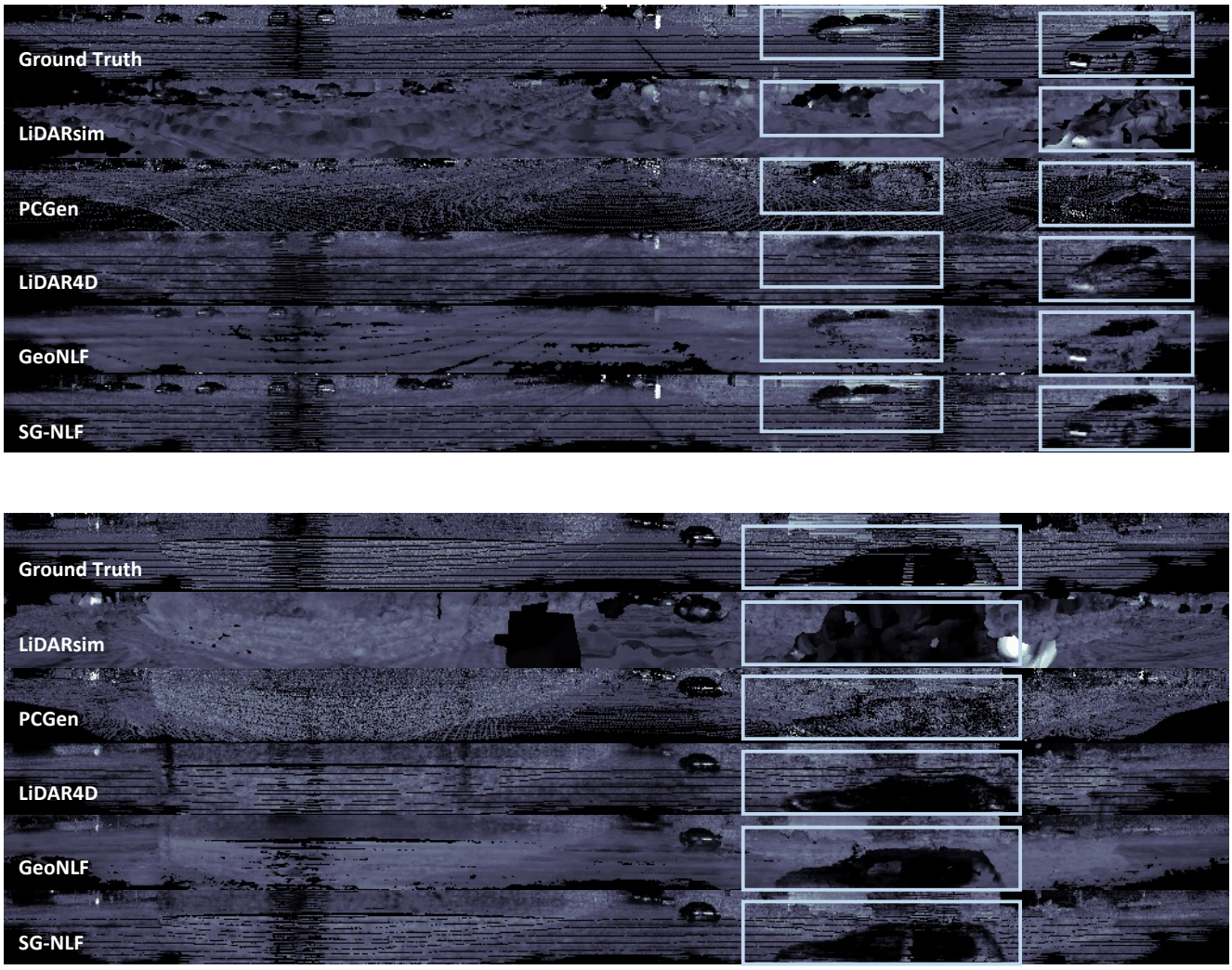

Figure 5. **Qualitative comparisons for LiDAR range intensity reconstruction and synthesis.** Both pose-dependent [3, 6, 13] and pose-free methods [12] are compared. Regions with obvious differences are highlighted in the rectangular boxes.

(65% reduction compared to LiDAR4D [13]) and 20 ms average inference time (over 12 times faster).

2.6. Training time

Our SG-NLF is efficient without a heavy training burden. We test on a 4090 GPU. (1) SG-NLF takes 1.7 hours for

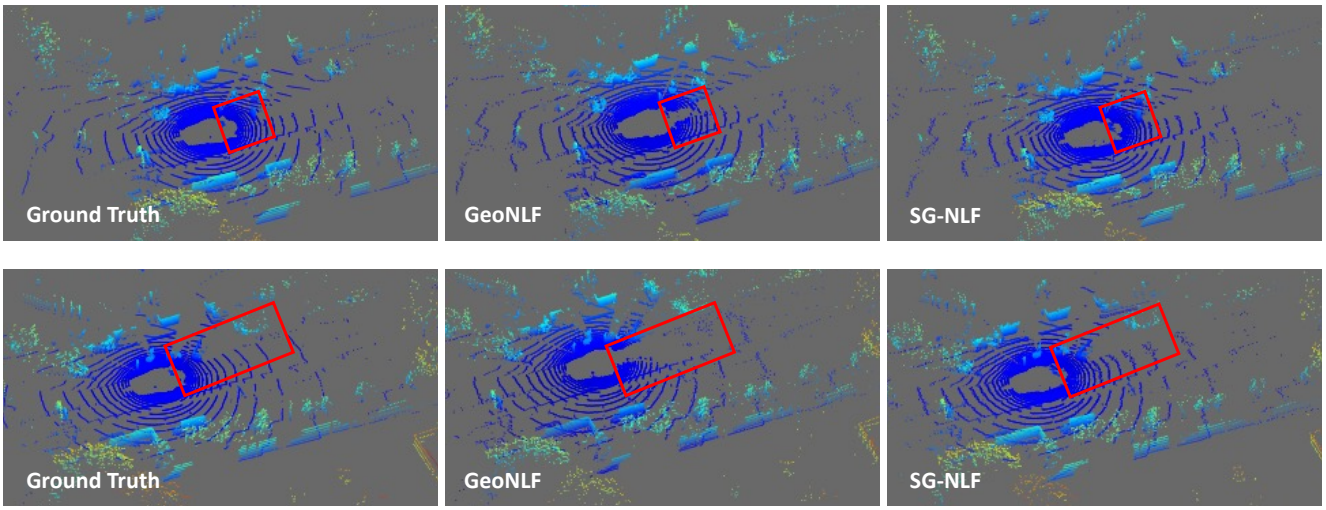

Figure 6. **Qualitative novel view LiDAR point cloud synthesis results on nuScenes [1, 12] dataset.** The point clouds from each frame are visualized from a top-down (Bird's-Eye View) perspective for analysis. Comparative results demonstrate that our SG-NLF achieves more complete road reconstruction than GeoNLF [12], as highlighted by the red rectangular boxes.

Table 6. **Model efficiency.** We report model parameters (M) and runtime (ms) of different methods [11–13]. Our evaluation follows the standard protocol using official implementations of competing methods [11–13]. Runtime are evaluated on the KITTI-360 [7, 12] dataset, reporting the average rendering time per sequence.

	LiDAR4D [13]	LiDAR-NeRF [11]	GeoNLF [12]	SG-NLF(Ours)
Params (M)	53.1	13.6	18.3	18.4
Time (ms)	242	55	20	25

training per scene, comparable to baselines like LiDAR4D (2.1 hours) and GeoNLF (1.6 hours). (2) Original NeRF consumes 82% of training time. Additional regularizations and networks only take up 18%.

3. More Implementation Details

Following Sec. 3.3, Sec. 3.4, and Sec. 4.1 of the submitted main manuscript, we present more detailed descriptions of our framework and implementations.

3.1. Hybrid Spectral-Geometric Representation

For geometric encoding, we use a hash grid with 20 levels, each level displaying 2-dimensional features, resulting in the same 40-dimensional features. The grid is mapped to a 2^{19} hash table. For spectral embedding, we use a 3-layer MLP for spherical mapping and another 3-layer MLP to learn the $K = 8$ eigenfunctions. We perform area-weighted sampling and select $M = 4096$ points for the spectral loss computation. Geometric encoding and spectral embedding constitute a 48-dimensional latent vector.

The fused spectral-geometric features are fed into a 2-layer 64-dimensional MLP, producing a 20-dimensional intrinsic feature vector and density value. The intrinsic features, combined with 12-band frequency-coded viewpoint information, are then fed into two independent 3-layer 64-dimensional MLPs to predict intensity values and ray-drop probabilities, respectively. The expectation of the density integrated along the ray serves as the depth value.

3.2. Global Pose Optimization

Beyond hybrid spectral-geometric representation for scene reconstruction, we additionally design a dynamically constructed graph for pose optimization. The edge inclusion threshold θ increases linearly from 0.5 to 0.9 during training to enforce progressively stricter edge selection. Similarly, the distance preservation threshold τ_d increases from 0.1 to 0.2 to accommodate improving alignment quality. For each edge $(i, j) \in \mathcal{E}$ in the pose graph, we compute the Chamfer Distance between the transformed point clouds:

$$\mathcal{L}_{\text{cd}}^{ij} = \frac{1}{|\mathcal{S}_i|} \sum_{\mathbf{x} \in \mathcal{S}_i} \min_{\mathbf{y} \in \mathcal{S}_j} \|\mathbf{x}-\mathbf{y}\|_2^2 + \frac{1}{|\mathcal{S}_j|} \sum_{\mathbf{y} \in \mathcal{S}_j} \min_{\mathbf{x} \in \mathcal{S}_i} \|\mathbf{x}-\mathbf{y}\|_2^2 , \quad (1)$$

where $\mathcal{S}_i$ and $\mathcal{S}_j$ represent the point clouds transformed by the optimized poses. The final pose graph loss combines the weighted Chamfer Distance across all edges:

$$\mathcal{L}_{\text{graph}} = \sum_{(i,j) \in \mathcal{E}} \alpha^{ij} \cdot \mathcal{L}_{\text{cd}}^{ij} . \quad (2)$$

3.3. Cross-frame Consistency

For cross-frame consistency supervision, the projected range images are fed into a 4-layer convolutional discriminator that outputs a 2D patch-wise authenticity map. The discriminator uses 64 base channels with a kernel size of

4 and stride of 2, doubling the number of channels in each layer. The discriminator is trained to distinguish between geometrically consistent reconstructions and inconsistent ones by analyzing local patches at multiple scales, enabling it to capture both fine-grained details and global structural coherence. Furthermore, we supervise the neural LiDAR fields using the 2D range image loss [13]:

$$\mathcal{L}_{ran}(\mathbf{r}) = \sum_{\mathbf{r}\in R} \lambda_d \|\hat{D}(\mathbf{r}) - D(\mathbf{r})\|_1 + \sum_{\mathbf{r}\in R} \lambda_i \|\hat{I}(\mathbf{r}) - I(\mathbf{r})\|_2^2 + \sum_{\mathbf{r}\in R} \lambda_p \|\hat{P}(\mathbf{r}) - P(\mathbf{r})\|_2^2 , \tag{3}$$

where $\hat{D}$, $\hat{I}$, and $\hat{P}$ represent the predicted depth, intensity, and ray-drop probability respectively, while D, I, and P denote the corresponding ground truth values.

3.4. Optimization Details

Following prior pose-free methods [12], we adopt an alternating optimization strategy to achieve stable convergence. The process alternates between optimizing the neural LiDAR field parameters (for reconstruction) and refining the frame poses (for registration). After every m_1 epochs of neural LiDAR fields optimization, we perform m_2 epochs of pose optimization, with the ratio m_2/m_1 decreasing from 10 to 1 during training [12]. The learning rate for pose optimization is synchronized with the neural field optimizer, following the same schedule as GeoNLF [12]. The training objective for neural LiDAR fields combines consistency loss, range image loss, and spectral loss with the following weights: $\lambda_d = 20$ for depth, $\lambda_i = 0.5$ for intensity, $\lambda_n = 0.1$ and $\lambda_o = 10$ for spectral components, and 1 for other terms. The overall spectral loss weight increases progressively from 0 to 1 during training. Other unmentioned optimization details are basically in line with GeoNLF [12].

References

[1] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11621–11631, 2020. 3, 5, 6

[2] Hwan Heo, Taekyung Kim, Jiyoung Lee, Jaewon Lee, Soohyun Kim, Hyunwoo J Kim, and Jin-Hwa Kim. Robust camera pose refinement for multi-resolution hash encoding. In *International Conference on Machine Learning*, pages 13000–13016. PMLR, 2023. 3, 4, 5

[3] Jiahui Huang, Zan Gojcic, Matan Atzmon, Or Litany, Sanja Fidler, and Francis Williams. Neural kernel surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4369–4379, 2023. 3, 4, 5

[4] Shengyu Huang, Zan Gojcic, Zian Wang, Francis Williams, Yoni Kasten, Sanja Fidler, Konrad Schindler, and Or Litany. Neural lidar fields for novel view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18236–18246, 2023. 3

[5] Junzhe Jiang, Chun Gu, Yurui Chen, and Li Zhang. GS-LiDAR: Generating realistic lidar point clouds with panoramic gaussian splatting. *arXiv preprint arXiv:2501.13971*, 2025. 1

[6] Chenqi Li, Yuan Ren, and Bingbing Liu. PCGen: Point cloud generator for lidar simulation. *arXiv preprint arXiv:2210.08738*, 2022. 3, 4, 5

[7] Yiyi Liao, Jun Xie, and Andreas Geiger. Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3292–3310, 2022. 1, 2, 3, 4, 6

[8] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5741–5751, 2021. 3, 4, 5

[9] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11167–11176, 2020. 3, 4, 5

[10] Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, and Kai Xu. Geometric transformer for fast and robust point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11143–11152, 2022. 4, 5

[11] Tang Tao, Longfei Gao, Guangrun Wang, Yixing Lao, Peng Chen, Hengshuang Zhao, Dayang Hao, Xiaodan Liang, Mathieu Salzmann, and Kaicheng Yu. Lidar-nerf: Novel lidar view synthesis via neural radiance fields. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 390–398, 2024. 1, 3, 4, 5, 6

[12] Weiyi Xue, Zehan Zheng, Fan Lu, Haiyun Wei, Guang Chen, et al. GeoNLF: Geometry guided Pose-Free Neural LiDAR Fields. *Advances in Neural Information Processing Systems*, 37:73672–73692, 2024. 1, 2, 3, 4, 5, 6, 7

[13] Zehan Zheng, Fan Lu, Weiyi Xue, Guang Chen, and Changjun Jiang. Lidar4d: Dynamic neural fields for novel space-time view lidar synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5145–5154, 2024. 1, 2, 3, 4, 5, 6, 7