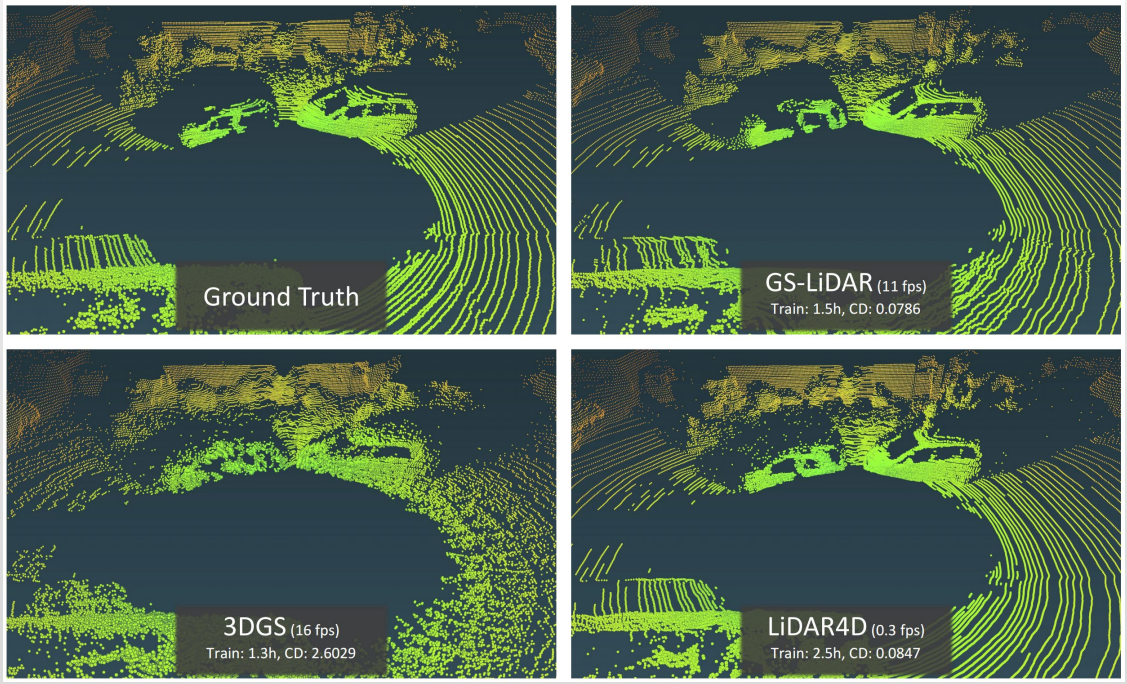
ICLR

GS-LiDAR: Generating Realistic LiDAR Point Clouds with Panoramic Gaussian Splatting

Junzhe Jiang, Chun Gu, Yurui Chen, Li Zhang
School of Data Science, Fudan University

Introduction



LiDAR novel view synthesis (NVS) has emerged as a novel task within LiDAR simulation, offering valuable simulated point cloud data from novel viewpoints to aid in autonomous driving systems.

However, existing LiDAR NVS methods typically rely on neural radiance fields (NeRF) as their 3D representation, which incurs significant computational costs in both training and rendering.

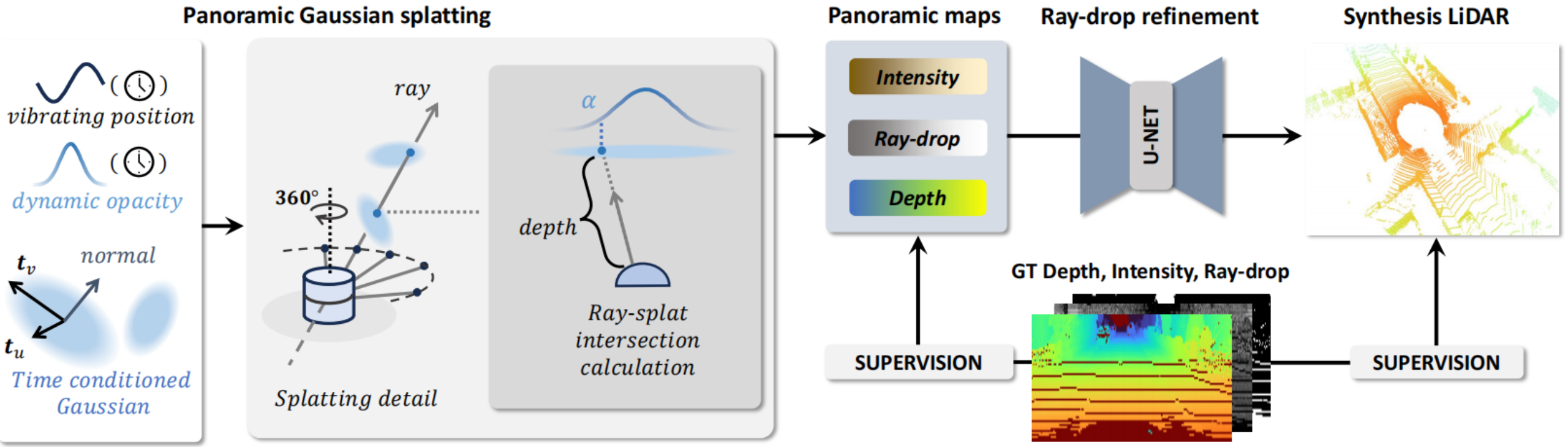
To address these challenges, we propose **GS-LiDAR**, a novel framework for generating realistic LiDAR point clouds with panoramic Gaussian splatting. Our approach employs 2D Gaussian primitives with periodic vibration properties, allowing for precise geometric reconstruction of both static and dynamic elements in driving scenarios. We further introduce a novel panoramic rendering technique with explicit ray-splat intersection, guided by panoramic LiDAR supervision.

Extensive experiments on KITTI-360 and nuScenes demonstrate the superiority of our method in terms of quantitative metrics, visual quality, as well as training and rendering efficiency

Pipeline

GS-LiDAR is based on 2D Gaussian primitives with periodic vibration properties, allowing for dynamic modeling of position and opacity along with accurate geometry. At a given timestamp, Gaussians query their states and utilize the proposed panoramic Gaussian splatting technique to render panoramic maps of depth, ray-drop, and intensity.

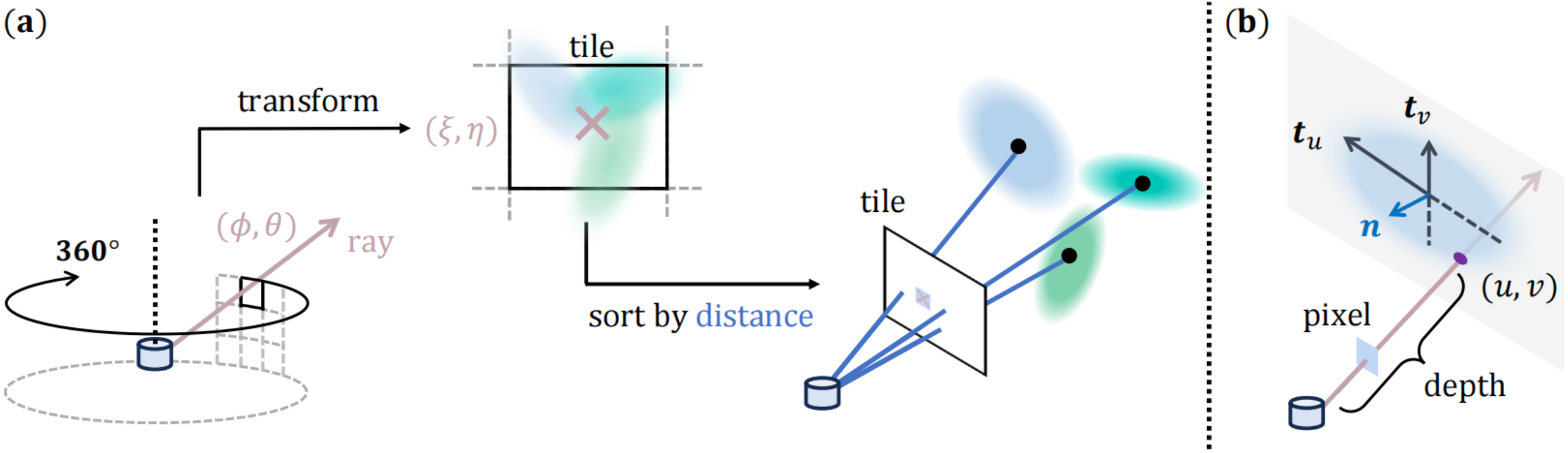
For each ray and Gaussian primitive, we calculate their intersection to obtain the depth and α, ensuring more geometrically consistent renderings. The results are subsequently refined by a well-trained U-Net to further enhance the quality of the point clouds



Panoramic Gaussian splatting

(a) Our method employs tile-based sorting and rendering. For panoramic ray maps, we first transform the spherical coordinate system into the pixel coordinate system. The pixel map is then divided into small tiles, and within each tile, Gaussian primitives are sorted based on their distance to the LiDAR origin.

(b) During pixel rendering, the α and depth are computed by calculating the intersection between the ray and the Gaussian primitive.



Experimental results

KITTI-360

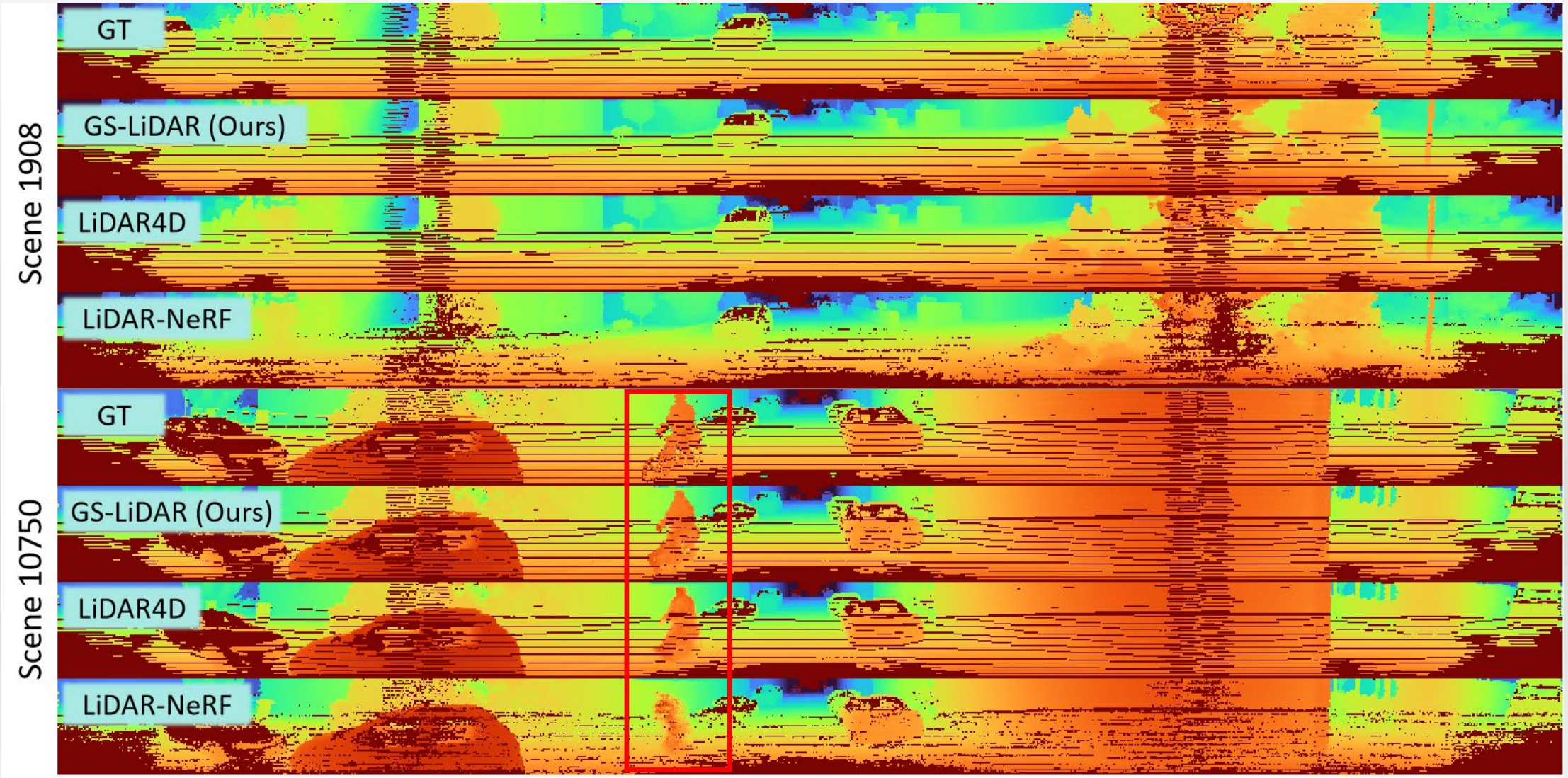
Method	Point Cloud		Depth					Intensity				
	CD↓	F-score↑	RMSE↓	MedAE↓	LPIPS↓	SSIM↑	PSNR↑	RMSE↓	MedAE↓	LPIPS↓	SSIM↑	PSNR↑
LiDARsim (Manivasagam et al., 2020)	3.2228	0.7157	6.9153	0.1279	0.2926	0.6342	21.4608	0.1666	0.0569	0.3276	0.3502	15.5853
NKSR (Huang et al., 2023)	1.8982	0.6855	5.8403	0.0996	0.2752	0.6409	23.0368	0.1742	0.0590	0.3337	0.3517	15.2081
PCGen (Li et al., 2023)	0.4636	0.8023	5.6583	0.2040	0.5391	0.4903	23.1675	0.1970	0.0763	0.5926	0.1351	14.1181
LiDAR-NeRF (Tao et al., 2023)	0.1438	0.9091	4.1753	0.0566	0.2797	0.6568	25.9878	0.1404	0.0443	0.3135	0.3831	17.1549
D-NeRF (Pumarola et al., 2021)	0.1442	0.9128	4.0194	0.0508	0.3061	0.6634	26.2344	0.1369	0.0440	0.3409	0.3748	17.3554
TiNeuVox-B (Fang et al., 2022)	0.1748	0.9059	4.1284	0.0502	0.3427	0.6514	26.0267	0.1363	0.0453	0.4365	0.3457	17.3535
K-Planes (Fridovich-Keil et al., 2023)	0.1302	0.9123	4.1322	0.0539	0.3457	0.6385	26.0236	0.1415	0.0498	0.4081	0.3008	17.0167
LiDAR4D (Zheng et al., 2024)	0.1089	0.9272	3.5256	0.0404	0.1051	0.7647	27.4767	0.1195	0.0327	0.1845	0.5304	18.5561
GS-LiDAR (Ours)	**0.1085**	**0.9231**	**3.1212**	**0.0340**	**0.0902**	**0.8553**	**28.4381**	**0.1161**	**0.0313**	**0.1825**	**0.5914**	**18.7482**

nuScenes

Method	Point Cloud		Depth					Intensity				
	CD↓	F-score↑	RMSE↓	MedAE↓	LPIPS↓	SSIM↑	PSNR↑	RMSE↓	MedAE↓	LPIPS↓	SSIM↑	PSNR↑
LiDARsim Manivasagam et al. (2020)	12.1383	0.6512	10.5539	0.3572	0.1871	0.5653	17.7841	0.0659	0.0115	0.1160	0.5170	23.7791
NKSR Huang et al. (2023)	11.4910	0.6178	9.3731	0.5763	0.2111	0.5637	18.7774	0.0680	0.0119	0.1290	0.5031	23.4905
PCGen Li et al. (2023)	2.1998	0.6341	8.8364	0.4011	0.1792	0.5440	19.2799	0.0768	0.0147	0.1308	0.4410	22.4428
LiDAR-NeRF Tao et al. (2023)	0.3225	0.8576	7.1566	0.0338	0.0702	0.7188	21.2129	0.0467	0.0076	0.0483	0.7264	26.9927
D-NeRF Pumarola et al. (2021)	0.3296	0.8513	7.1089	0.0368	0.0789	0.7130	21.2594	0.0467	0.0080	0.0492	0.7180	26.9951
TiNeuVox-B Fang et al. (2022)	0.3920	0.8627	7.2093	0.0290	0.1549	0.6873	21.0932	0.0462	0.0080	0.1294	0.7107	26.8620
K-Planes Fridovich-Keil et al. (2023)	0.2982	0.8887	6.7960	0.0209	0.1218	0.7258	21.6203	0.0438	0.0076	0.1127	0.7364	27.4227
LiDAR4D (Zheng et al., 2024)	0.2443	0.8915	6.7831	0.0258	0.0569	0.7396	21.7189	0.0426	0.0071	0.0459	0.7498	27.7977
GS-LiDAR (Ours)	**0.2382**	**0.9055**	**5.8925**	**0.0198**	**0.0708**	**0.8394**	**22.2482**	**0.0415**	**0.0067**	**0.0627**	**0.7291**	**27.7420**

Experimental results

Range map



Experimental results

ICLR

Thanks!

Project page:

https://github.com/fudan-zvg/GS-LiDAR