Learning to Generate Realistic LiDAR Point Clouds – Supplementary Material

Vlas Zyrianov, Xiyue Zhu, and Shenlong Wang

University of Illinois at Urbana-Champaign, IL, USA
{vlasz2,xiyuez2,shenlong}@illinois.edu

Abstract. In the supplementary material, we first provide more qualitative and quantitative results and additional ablation analysis in Sec. 1. In addition, we report experimental results on the challenging NuScenes dataset in Sec. 2. Finally, we provide additional quantitative and qualitative results for posterior sampling in Sec. 3. The supplementary video "*LiDARGen-intro.mp4*" briefly introduces our method, demonstrates the diffusion process in detail on KITTI-360, and compares qualitative results with other methods.

Keywords: 3D generation, self-driving, generative models

1 Additional Analysis

Qualitative Ablation Study We conduct ablation studies to justify the design choice of our algorithm. We compare the same score function model in three different settings: with circular convolution and without coordinate encoding, with circular convolution and without coordinate encoding, and our final model, which uses circular convolution and a coordinate encoding. Qualitative results are shown in Fig. 2.

Without **circular convolution**, a discontinuity appears in the point cloud representation horizontally, starting from the origin. This discontinuity is most clearly seen in the sixth row of the second column in Fig. 2. This discontinuity is caused by the left and right edges of the range image lacking a receptive field between each other when using normal convolutions. To address this issue, we use Circular Convolutions. Qualitatively, this discontinuity is fixed with this change.

With the help of **coordinate encoding**, our approach generates more straight road layouts that appropriately reflect the real-world layout distribution in the urban driving environment.

Comparison to Point-based Backbone We also compare our model against the point-based score-matching model proposed in ShapeGF [3]. The original ShapeGF model was trained and tested in ShapeNet. We adapt their model on LiDAR generation by changing the point cloud size to be 50k points and changing the noise level schedule by setting the number of noise steps to be 15 and end noise sigma to be 0.001. We train its stage 1 autoencoding and stage 2 GAN model from scratch on the KITTI training set until the validation loss

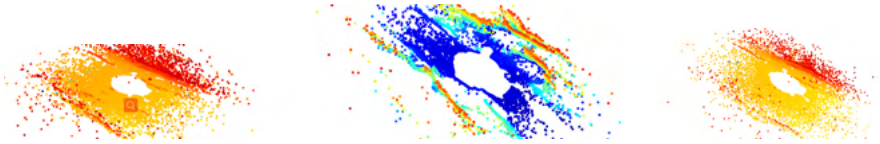

Fig. 1: Qualitative Results for ShapeGF [3] on KITTI.

converges. As shown in Fig. 1, ShapeGF cannot provide physically feasible results like equirectangular-based approaches. Further, despite working very well for ShapeNet-like objects, we find it has a strong mode collapse issue on complicated urban driving scenes.

2 nuScenes

Datasets The proposed LiDAR generation model is applicable across different datasets, geolocations and LiDAR sensors. To demonstrate this, we train and test our model on the **nuScenes dataset** [2]. nuScenes contains 297,737 LiDAR sweeps in the training set and 52,423 LiDAR sweeps in the testing and cross-validation set. The LiDAR sweeps were collected in the cities of Boston and Singapore. These locations present readings that are uniquely different compared to readings from the KITTI dataset [4], which have mostly been collected in the suburbs of Karlsruhe, Germany. In addition to the different environment, the LiDAR sensor used in nuScenes is different. The data was collected with a Velodyne HDL32E, which has 32 Beams and a $+10°$ to $-30°$ vertical Field of View. Compared to the sensor used in the KITTI dataset (Velodyne HDL-64E), the one used in nuScenes has lower vertical resolution, however has a higher vertical field of view.

Implementation Details We encode the raw nuScenes point cloud into an equirectangular view. Specifically, our range image resolution is set to be 32×1024, tailored for nuScenes LiDAR sensor's spatial resolution. And our Cartesian-to-range encoding is changed to the following: $\mathbf{z}_i = (\theta_i, \phi_i, d_i), r_i \colon \mathcal{I}(\lfloor \theta_i/s_\theta \rfloor, \lfloor \phi_j/s_\phi \rfloor) = \left(\frac{1}{6.5}\log_2(d_i+1), \frac{1}{31}r_i\right)$, ensuring the full range to be normalized to $[0, 1]$,

Baselines Following the main paper's experiments on KITTI-360. We also leverage two baselines for comparison. The first is ProjectedGAN [5]. This was the second-best performing model in the KITTI evaluation, so we include it in this evaluation too. The second baseline is Caccia et al.'s LiDAR VAE [1]. All models were trained with the same settings as for KITTI. Note that the GAN model described in Caccia et al. [1] does not converge after our hyper-parameter tuning, hence we omit it from this study.

Experimental Results Fig. 3 depicts the qualitative comparison results. From this figure, we can see that our method still achieves superior results compared to both VAE [1] and projected GAN [5]. An AB test on a group of four human sub-

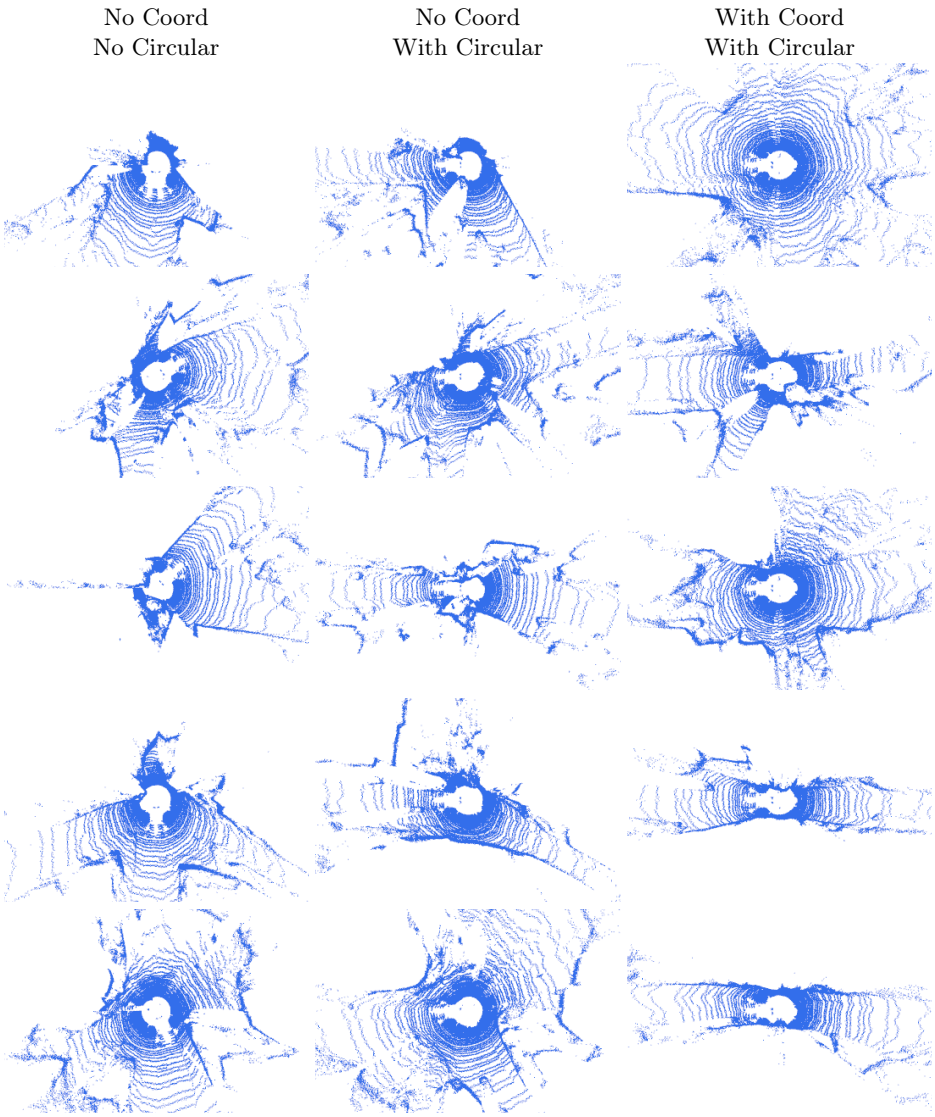

Fig. 2: Qualitative Ablation Comparison. Circular convolutions prevents a discontinuity that happens on a vector that starts at the origin and points left. The coordinate encoding encourages the network to generate straighter and more realistic roads.

jects suggests that our method is still significantly favored over other competing algorithms in 89% of cases.

While achieving superior human performance, we notice that our current nuScenes model has a noticeable weakness on the nuScenes dataset. In particular,

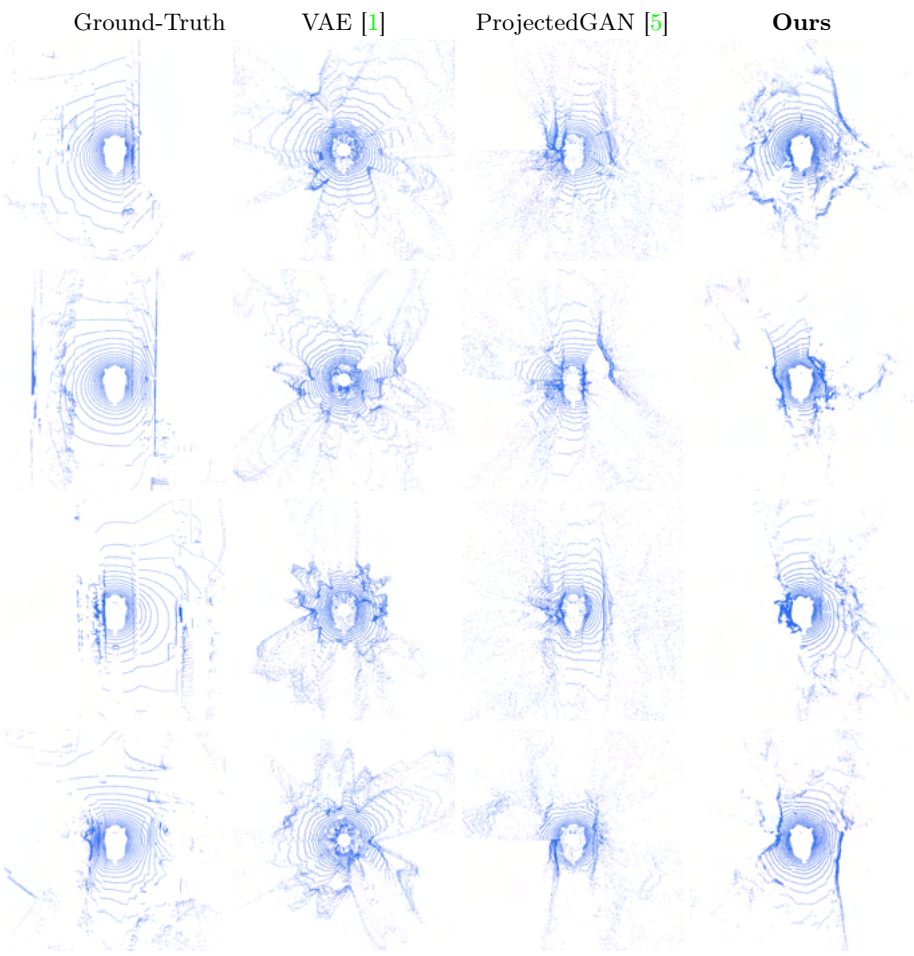

Fig. 3: Qualitative Results on the nuScenes dataset.

our method tends to generate point clouds that concentrate their mass closer to the viewpoint. As a result, despite superior visual quality, our MMD score at BEV is worse than VAE and Projected GAN (2e-3 vs. 1.1e-3 and 6e-5). In the nuScenes experiment, we directly adopt the same starting and ending noise level (150 and 0.01) as in KITTI, which might be too large for nuScenes. We believe better-tuned noise parameters will resolve this issue.

3 Additional Posterior Sampling Results

Densification We demonstrate additional densification results in Fig. 5 and Fig. 6. From the figure, we can see our produced diversified point clouds are both highly-realistic and have high fidelity and consistency to the input.

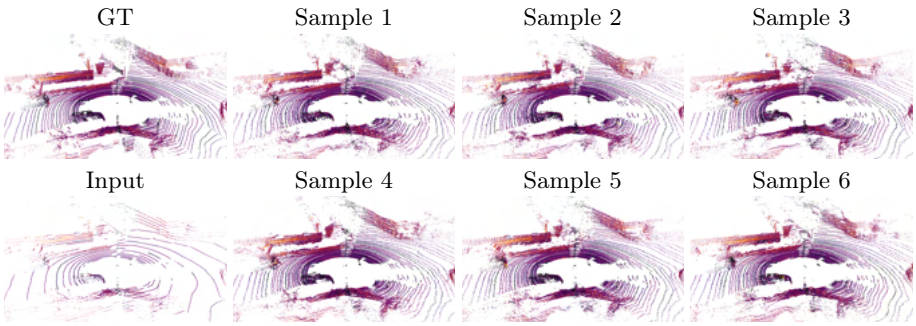

Fig. 4: Multiple samples of posterior sampling conditioned on the same sparse input. Notice the diversity of the shapes and intensity values of the car on the left, as well as the structure of the wall.

Fig 4 provides additional results demonstrating multiple samples given the same sparse point input. The figure shows that our posterior sampling approach produces multiple plausible resulting point clouds, further demonstrating the advantage of tackling such a task in a probabilistic fashion.

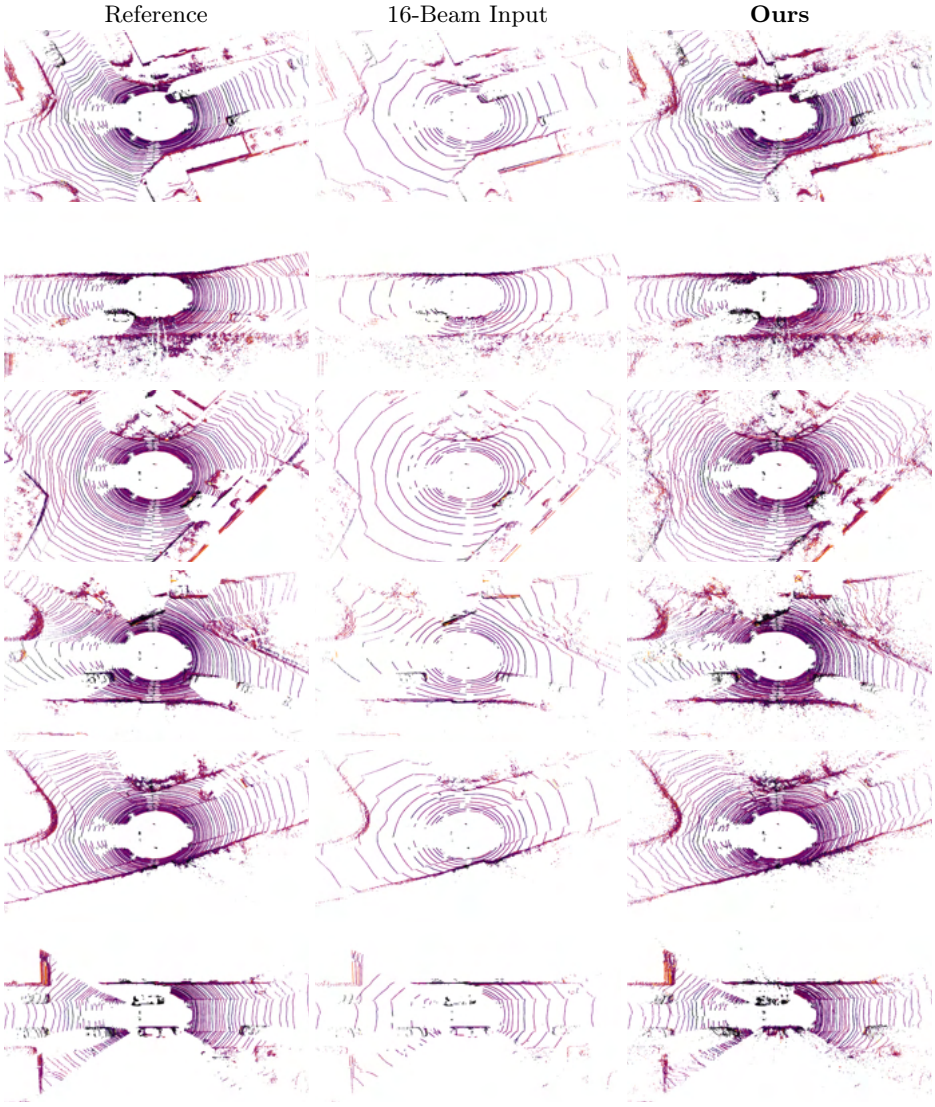

Fig. 5: Additional Results for Unsupervised LiDAR Densification.

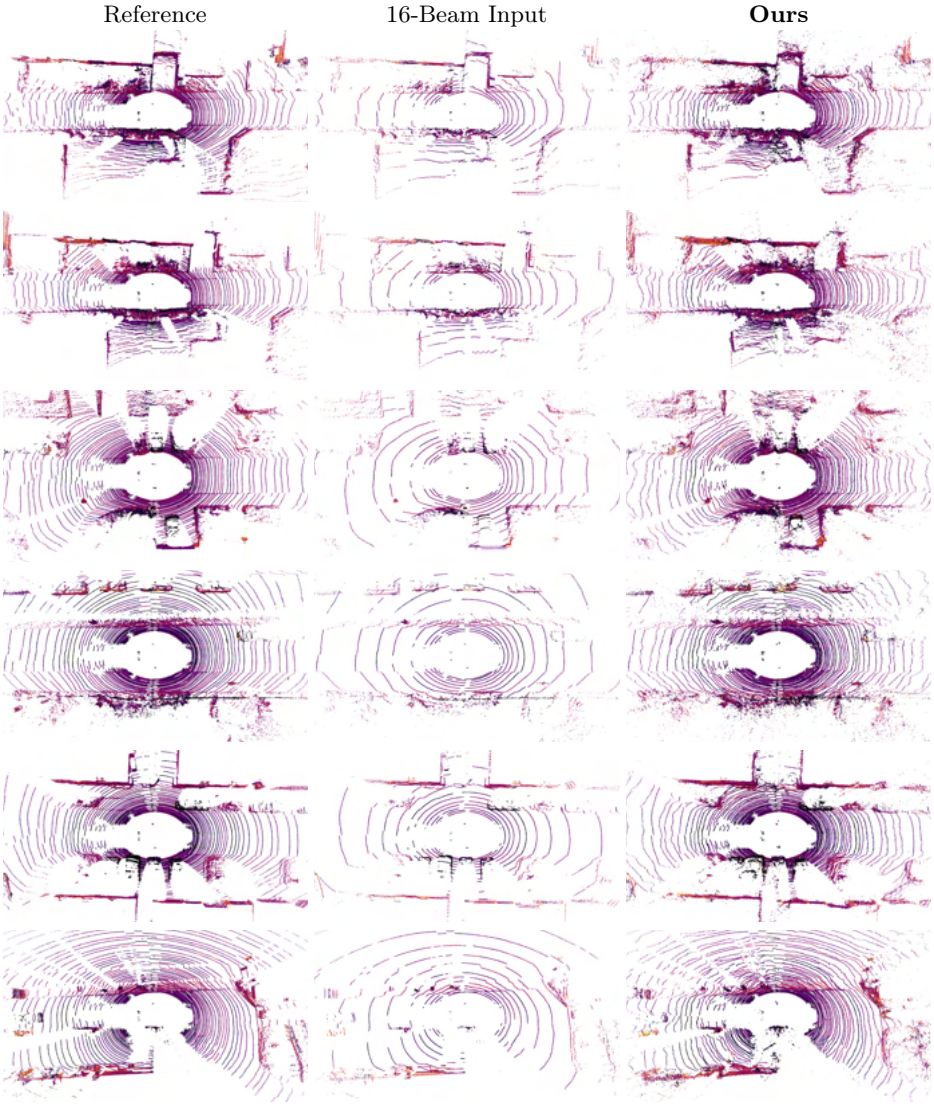

Fig. 6: Additional Results for Unsupervised LiDAR Densification (continued).

References

1. Caccia, L., van Hoof, H., Courville, A.C., Pineau, J.: Deep generative modeling of lidar data. IROS pp. 5034–5040 (2019) 2, 4
2. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. arXiv preprint arXiv:1903.11027 (2019) 2
3. Cai, R., Yang, G., Averbuch-Elor, H., Hao, Z., Belongie, S., Snavely, N., Hariharan, B.: Learning gradient fields for shape generation. In: European Conference on Computer Vision. pp. 364–381. Springer (2020) 1, 2
4. Liao, Y., Xie, J., Geiger, A.: KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. arXiv preprint arXiv:2109.13410 (2021) 2
5. Sauer, A., Chitta, K., Müller, J., Geiger, A.: Projected gans converge faster. In: Advances in Neural Information Processing Systems (NeurIPS) (2021) 2, 4