电子科技大学
University of Electronic Science and Technology of China

LiDAR Data Synthesis with Denoising Diffusion Probabilistic Models

R2DM

R2DM

Introduction

- 使用 DDPM 框架构建
- 从三个方面设计整体模型：

1. Loss Function
2. Data Representation
3. Spatial Inductive Bias

- 使用 KITTI-360 以及 KITTI-RAW 两个 Datasets 进行了对应的 Ablation Study 以及 Evaluation

R2DM

Related Works

1. VAE + GAN
2. 关于 Range Image 中 ray-drop 噪声的研究

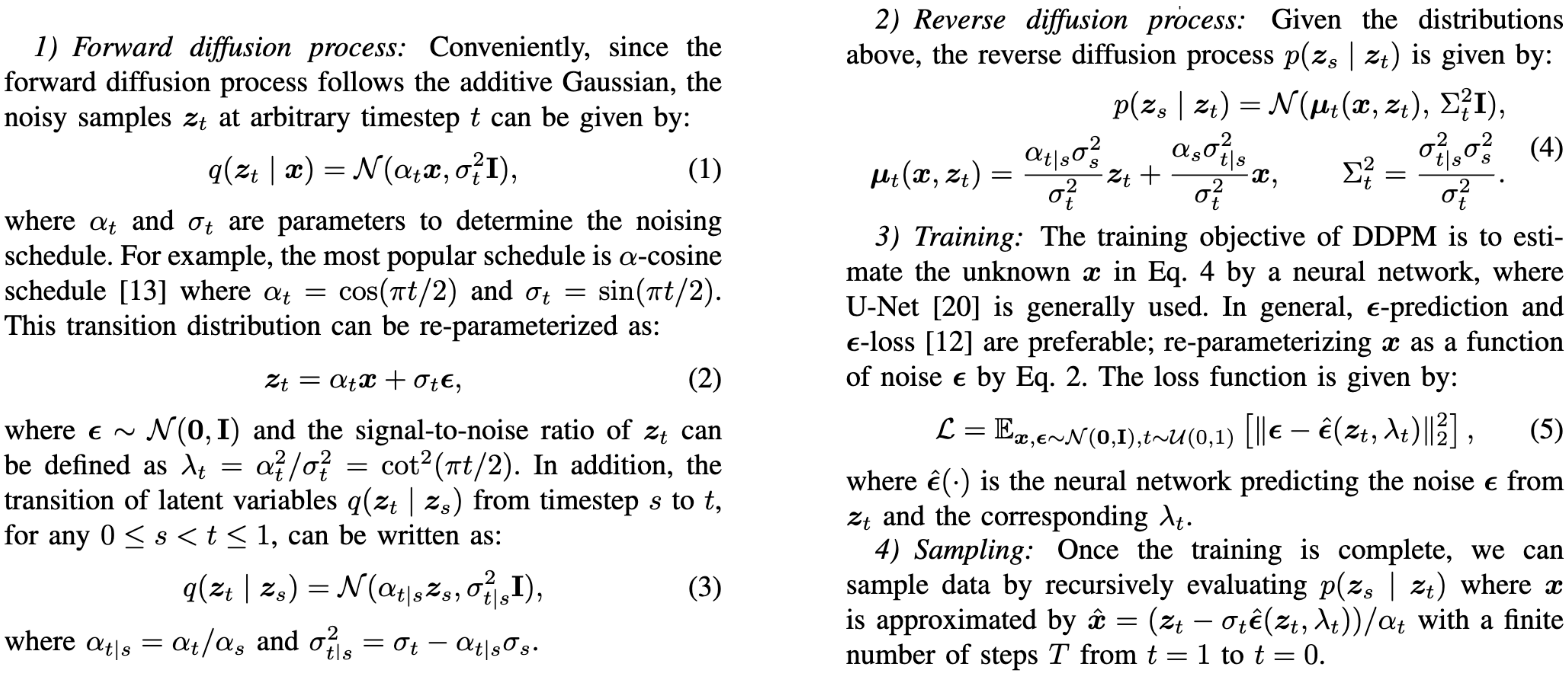
   Ray-drop 指的是一种离散丢失噪声，导致图像上出现离散的缺失点。这种噪声使得 range image 的数据完整性下降，影响后续处理效果
3. DUSty：基于 GAN 的模型，分离 range image 中的噪声部分，生成“去噪”版本的图像，同时估计缺失部分的丢失概率，帮助理解和模拟噪声的分布
4. LiDARGen：Score-based Diffusion Model，通过朗格纹动力学采样

   存在的问题：

   1. 与前人工作提升较小
   2. 由于 time-step 过大，采样效率太低

R2DM

Proposed Method: Preliminary



1) Forward diffusion process: Conveniently, since the forward diffusion process follows the additive Gaussian, the noisy samples $\boldsymbol{z}_t$ at arbitrary timestep t can be given by:

$$q(\boldsymbol{z}_t \mid \boldsymbol{x}) = \mathcal{N}(\alpha_t \boldsymbol{x}, \sigma_t^2 \mathbf{I}), \tag{1}$$

where α_t and σ_t are parameters to determine the noising schedule. For example, the most popular schedule is α-cosine schedule [13] where $\alpha_t = \cos(\pi t/2)$ and $\sigma_t = \sin(\pi t/2)$. This transition distribution can be re-parameterized as:

$$\boldsymbol{z}_t = \alpha_t \boldsymbol{x} + \sigma_t \boldsymbol{\epsilon}, \tag{2}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and the signal-to-noise ratio of $\boldsymbol{z}_t$ can be defined as $\lambda_t = \alpha_t^2/\sigma_t^2 = \cot^2(\pi t/2)$. In addition, the transition of latent variables $q(\boldsymbol{z}_t \mid \boldsymbol{z}_s)$ from timestep s to t, for any $0 \leq s < t \leq 1$, can be written as:

$$q(\boldsymbol{z}_t \mid \boldsymbol{z}_s) = \mathcal{N}(\alpha_{t|s} \boldsymbol{z}_s, \sigma_{t|s}^2 \mathbf{I}), \tag{3}$$

where $\alpha_{t|s} = \alpha_t/\alpha_s$ and $\sigma_{t|s}^2 = \sigma_t - \alpha_{t|s}\sigma_s$.

2) Reverse diffusion process: Given the distributions above, the reverse diffusion process $p(\boldsymbol{z}_s \mid \boldsymbol{z}_t)$ is given by:

$$\begin{gathered} p(\boldsymbol{z}_s \mid \boldsymbol{z}_t) = \mathcal{N}(\boldsymbol{\mu}_t(\boldsymbol{x}, \boldsymbol{z}_t),\, \Sigma_t^2 \mathbf{I}), \\ \boldsymbol{\mu}_t(\boldsymbol{x}, \boldsymbol{z}_t) = \frac{\alpha_{t|s}\sigma_s^2}{\sigma_t^2} \boldsymbol{z}_t + \frac{\alpha_s \sigma_{t|s}^2}{\sigma_t^2} \boldsymbol{x}, \qquad \Sigma_t^2 = \frac{\sigma_{t|s}^2 \sigma_s^2}{\sigma_t^2}. \end{gathered} \tag{4}$$

3) Training: The training objective of DDPM is to estimate the unknown $\boldsymbol{x}$ in Eq. 4 by a neural network, where U-Net [20] is generally used. In general, $\boldsymbol{\epsilon}$-prediction and $\boldsymbol{\epsilon}$-loss [12] are preferable; re-parameterizing $\boldsymbol{x}$ as a function of noise $\boldsymbol{\epsilon}$ by Eq. 2. The loss function is given by:

$$\mathcal{L} = \mathbb{E}_{\boldsymbol{x}, \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), t \sim \mathcal{U}(0,1)} \left[\| \boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}}(\boldsymbol{z}_t, \lambda_t) \|_2^2 \right], \tag{5}$$

where $\hat{\boldsymbol{\epsilon}}(\cdot)$ is the neural network predicting the noise $\boldsymbol{\epsilon}$ from $\boldsymbol{z}_t$ and the corresponding λ_t.

4) Sampling: Once the training is complete, we can sample data by recursively evaluating $p(\boldsymbol{z}_s \mid \boldsymbol{z}_t)$ where $\boldsymbol{x}$ is approximated by $\hat{\boldsymbol{x}} = (\boldsymbol{z}_t - \sigma_t \hat{\boldsymbol{\epsilon}}(\boldsymbol{z}_t, \lambda_t))/\alpha_t$ with a finite number of steps T from $t = 1$ to $t = 0$.

R2DM

Proposed Method: Loss Function

$$\mathcal{L} = \mathbb{E}_{\boldsymbol{x},\boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I}),t\sim\mathcal{U}(0,1)}\left[\|\boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}}(\boldsymbol{z}_t,\lambda_t)\|_2^2\right]$$

- 上图展示了使用 L2 范式的损失函数
- Monocular depth estimation using diffusion models 提出了 L1 范式的损失函数对较大的深度值和噪点有更强的鲁棒性，因此在单目深度估计任务中有更好的表现
- 本文提出了将 L1 范式和 L2 范式相结合的 Huber Loss

R2DM

Proposed Method: Data Representation

1. 使用 range view 的形式，将 range 和 reflectance intensity 从笛卡尔坐标映射到 equirectangular image 上

2. 对 range value 进行对数缩放

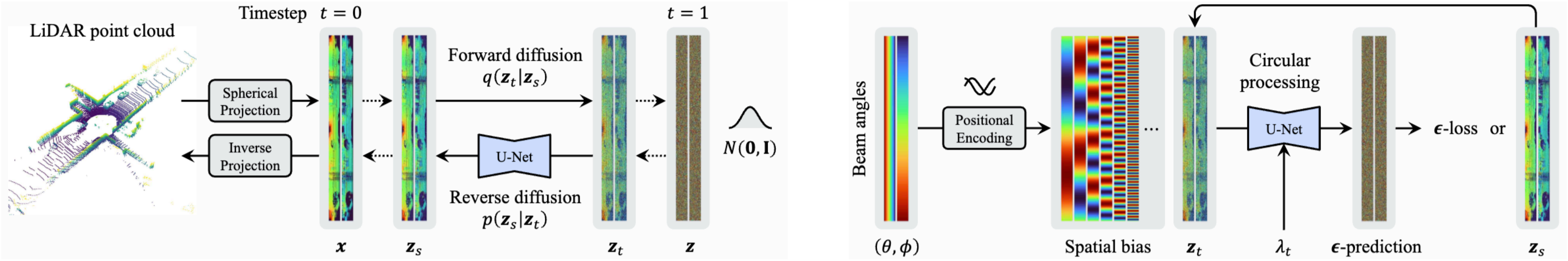
$$\boldsymbol{d}_{\log} = \frac{\log(\boldsymbol{d}+1)}{\log(d_{\max}+1)},$$

3. 同时测试了使用 Standard Metric Depth 以及 Inverse Depth 处理深度

R2DM

Proposed Method: Spatial inductive bias

1. LiDARGen 直接将笛卡尔坐标系的角度显式地作为空间归纳偏置 concat，文中称之为 identity function（恒等函数）
2. 作者认为单独有坐标值缺少水平上的连续性以及高频细节
3. 提出了两种 Positional Encoding 的方式：
   1. Spherical Harmonics：使用正交的球谐函数基函数表示笛卡尔坐标
   2. Fourier Features：使用 log2-spaced Scheme 将仰角和方位角扩展到二次方频率



(a) Range/reflectance image-based diffusion model

(b) Details of the reverse diffusion process

Fig. 2. **Overview of R2DM**. (a) The diffusion processes are performed on the range/reflectance image representation. (b) U-Net is trained to recursively denoise the latent variables z_t at $t > 0$, conditioned by the beam angle-based spatial bias and the scheduled signal-to-noise ratio λ_t.

R2DM

Proposed Method: Noise Prediction Model

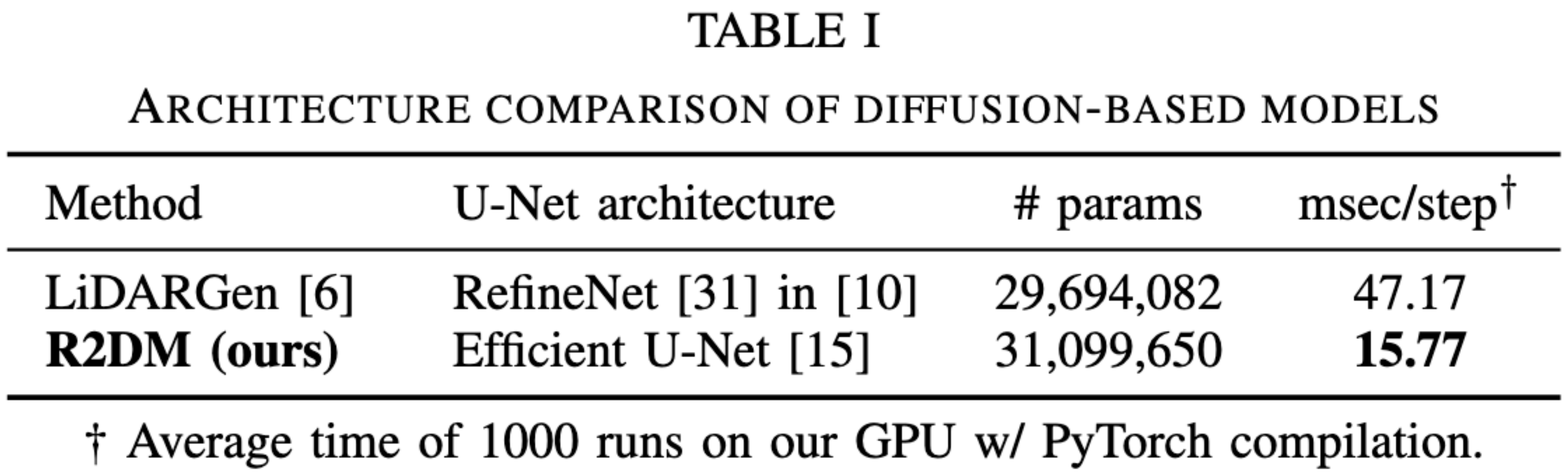
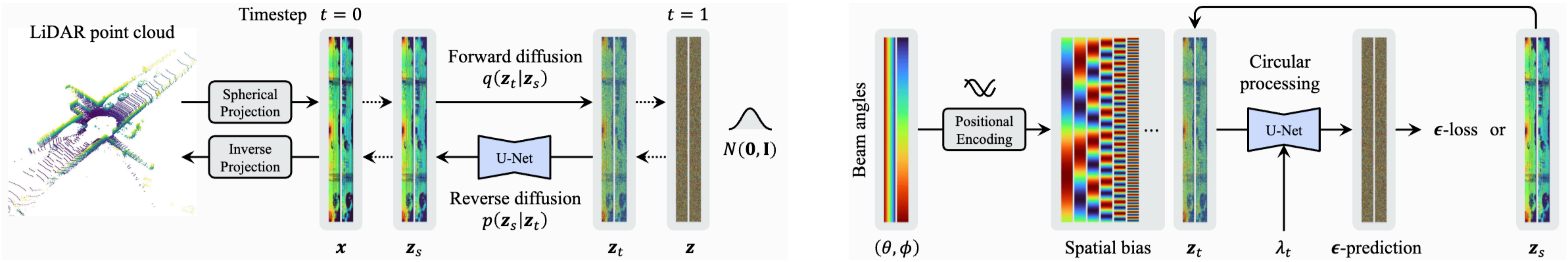


TABLE I

ARCHITECTURE COMPARISON OF DIFFUSION-BASED MODELS

Method	U-Net architecture	# params	msec/step[†]
LiDARGen [6]	RefineNet [31] in [10]	29,694,082	47.17
R2DM (ours)	Efficient U-Net [15]	31,099,650	**15.77**

† Average time of 1000 runs on our GPU w/ PyTorch compilation.



(a) Range/reflectance image-based diffusion model

(b) Details of the reverse diffusion process

Fig. 2. **Overview of R2DM**. (a) The diffusion processes are performed on the range/reflectance image representation. (b) U-Net is trained to recursively denoise the latent variables z_t at $t > 0$, conditioned by the beam angle-based spatial bias and the scheduled signal-to-noise ratio λ_t.

R2DM

Experiments: Compared with LiDARGen

1. 在 64 beam 的 KITTI-360 数据集上进行，每个 LiDAR 数据都被投影到 64 ✖ 1024 的 range view image 上
2. 消融实验变量设置为 3 个：
    1. Loss Function
    2. Range Representation
    3. Positional Encoding
4. 在 NVIDIA A6000 GPU 上使用 300k 步数训练了 20 个 GPU hours，以 1024 步数采样 10k 个样本消耗了 30 个 GPU hours

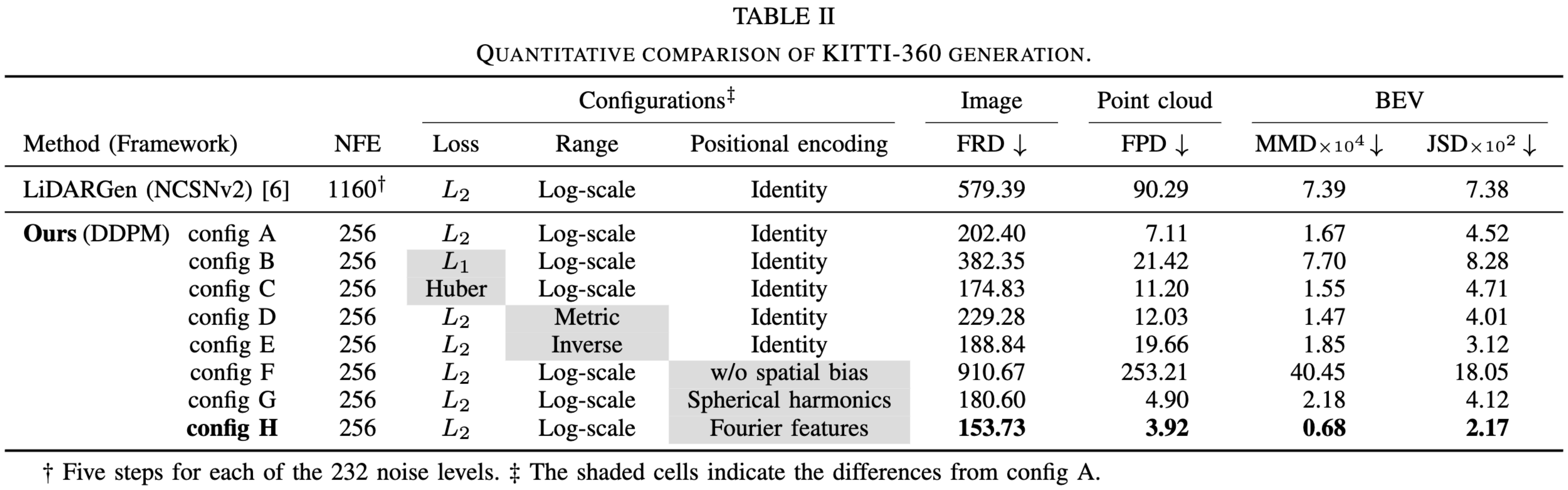


TABLE II
Quantitative comparison of KITTI-360 generation.

		Configurations‡			Image	Point cloud	BEV	
Method (Framework)	NFE	Loss	Range	Positional encoding	FRD ↓	FPD ↓	MMD$_{\times 10^4}$ ↓	JSD$_{\times 10^2}$ ↓
LiDARGen (NCSNv2) [6]	1160†	L_2	Log-scale	Identity	579.39	90.29	7.39	7.38
Ours (DDPM) config A	256	L_2	Log-scale	Identity	202.40	7.11	1.67	4.52
config B	256	L_1	Log-scale	Identity	382.35	21.42	7.70	8.28
config C	256	Huber	Log-scale	Identity	174.83	11.20	1.55	4.71
config D	256	L_2	Metric	Identity	229.28	12.03	1.47	4.01
config E	256	L_2	Inverse	Identity	188.84	19.66	1.85	3.12
config F	256	L_2	Log-scale	w/o spatial bias	910.67	253.21	40.45	18.05
config G	256	L_2	Log-scale	Spherical harmonics	180.60	4.90	2.18	4.12
config H	256	L_2	Log-scale	Fourier features	**153.73**	**3.92**	**0.68**	**2.17**

† Five steps for each of the 232 noise levels. ‡ The shaded cells indicate the differences from config A.

R2DM

Experiments: Compared with LiDARGen

1. Range View 模态：FRD 指标

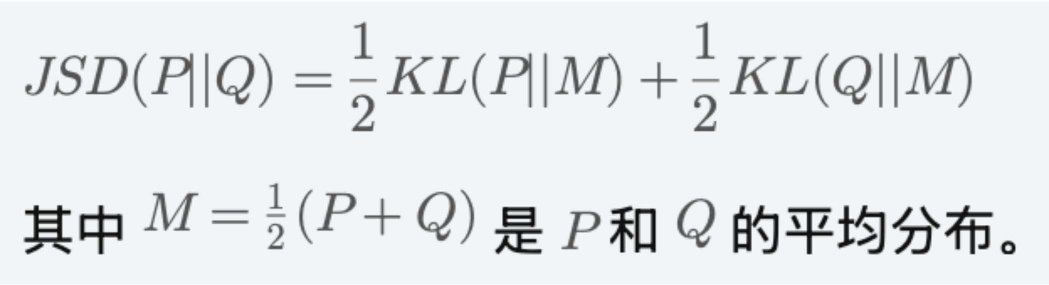
   在 RangeNet-53 的特征空间上计算生成的 range view 与真实的 rang view 分布之间的 Frechet Distance

2. Point Cloud 模态：FPD 指标

   在 PointNet 的特征空间上计算生成的 range view 与真实的 rang view 分布之间的 Frechet Distance

3. BEV 模态：JSD & MMD

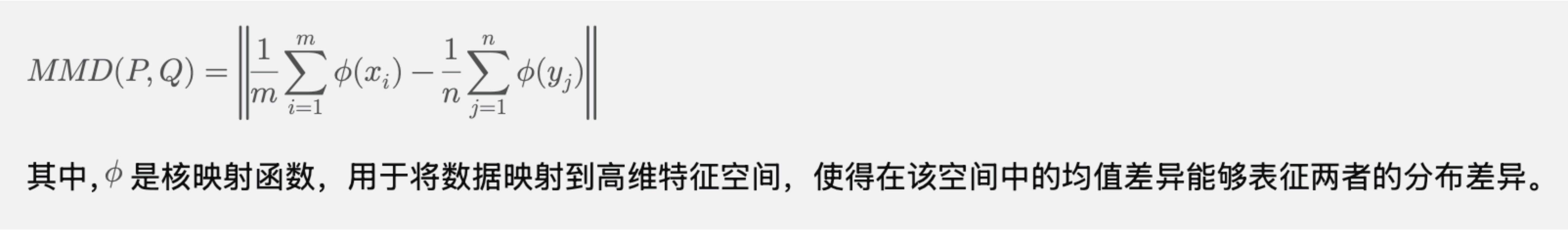
   1. JSD

      $$JSD(P||Q) = \frac{1}{2}KL(P||M) + \frac{1}{2}KL(Q||M)$$

      其中 $M = \frac{1}{2}(P+Q)$ 是 P 和 Q 的平均分布。

   2. MMD

      $$MMD(P,Q) = \left\| \frac{1}{m}\sum_{i=1}^{m}\phi(x_i) - \frac{1}{n}\sum_{j=1}^{n}\phi(y_j) \right\|$$

      其中，ϕ 是核映射函数，用于将数据映射到高维特征空间，使得在该空间中的均值差异能够表征两者的分布差异。

R2DM

Experiments: Compared with LiDARGen

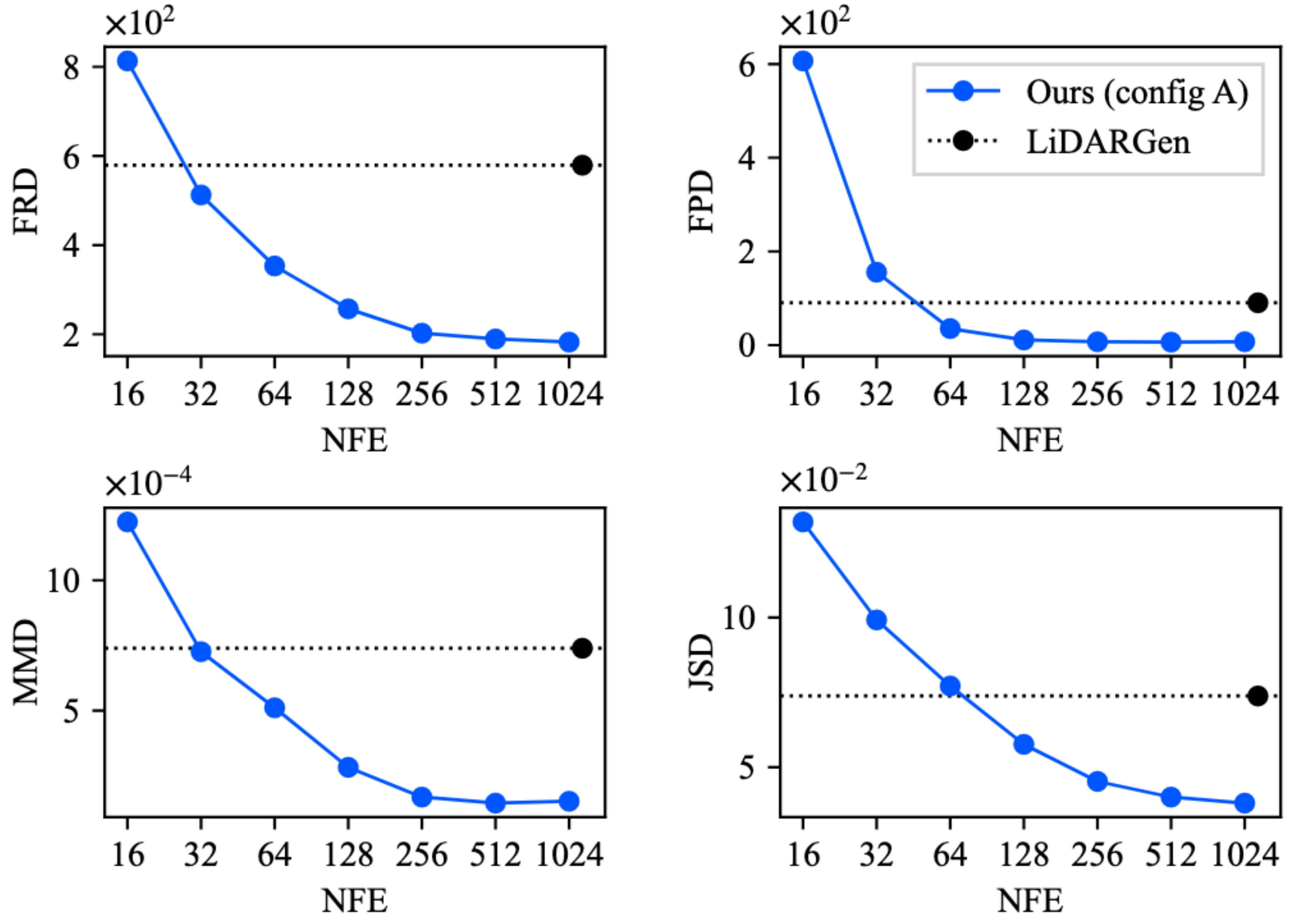


Fig. 4. **Comparison of diffusion-based methods**. For overall metrics, our method achieved better scores with the significantly lower number of function evaluations (NFE), against 1160 steps of LiDARGen [6].

R2DM

Experiments: Compared with GAN Method

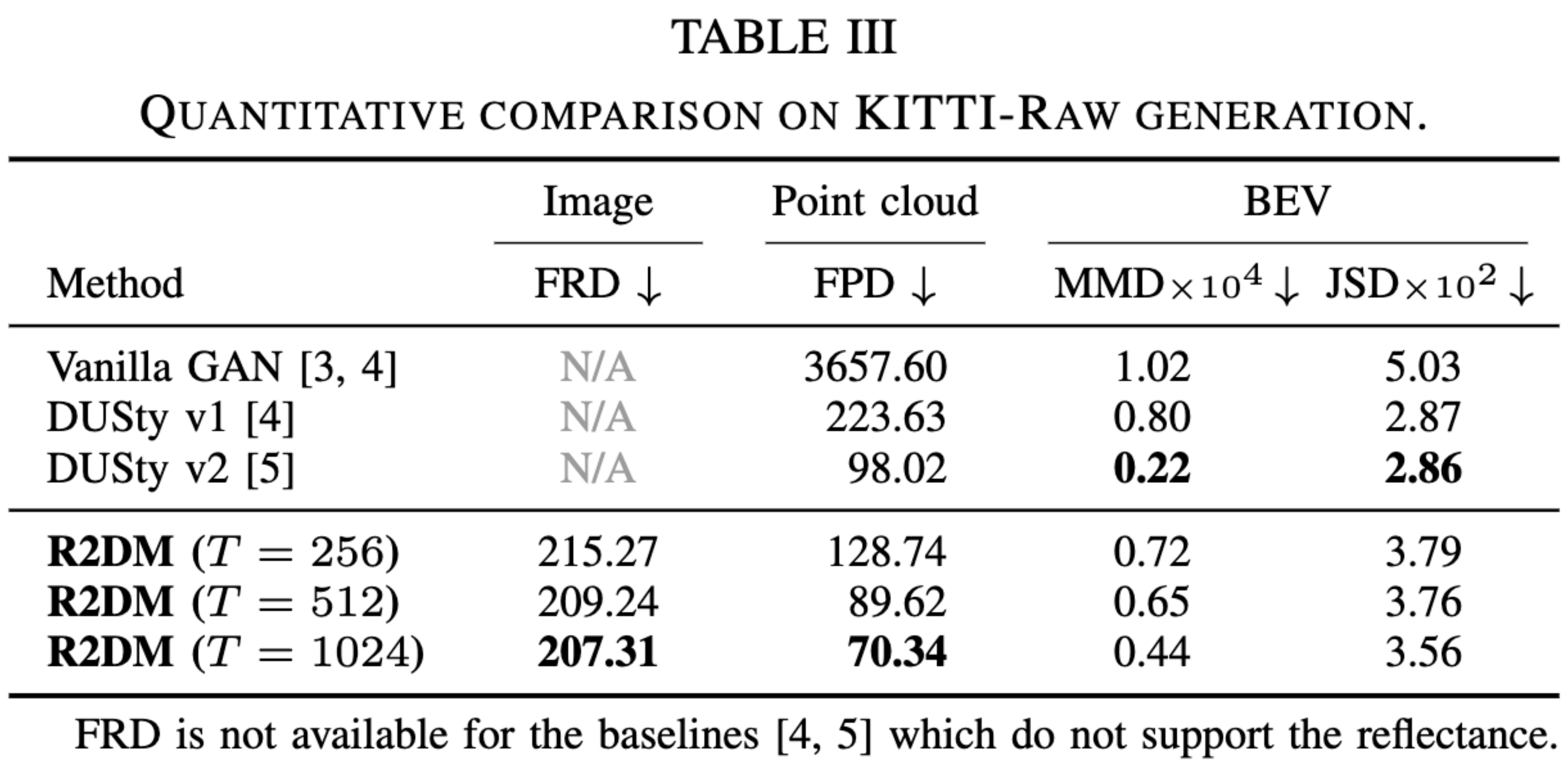
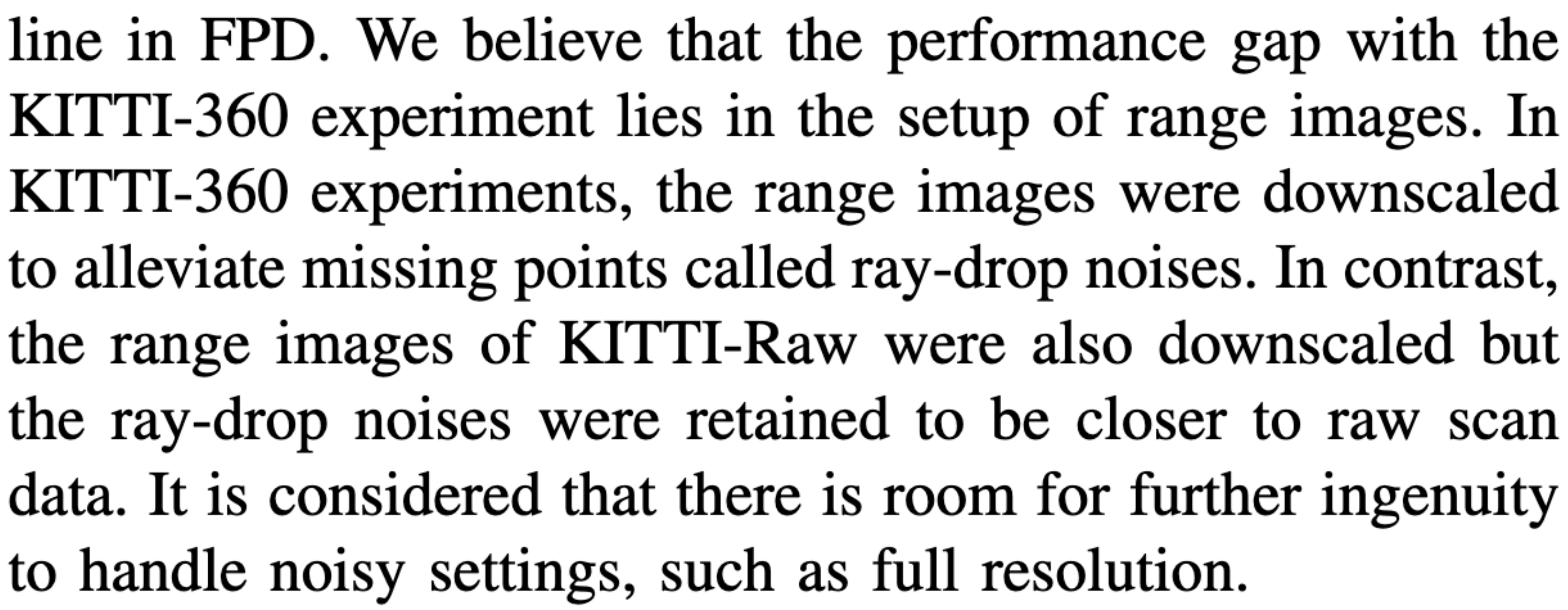


TABLE III

QUANTITATIVE COMPARISON ON KITTI-RAW GENERATION.

	Image	Point cloud	BEV	
Method	FRD ↓	FPD ↓	MMD $\times 10^4$ ↓	JSD $\times 10^2$ ↓
Vanilla GAN [3, 4]	N/A	3657.60	1.02	5.03
DUSty v1 [4]	N/A	223.63	0.80	2.87
DUSty v2 [5]	N/A	98.02	**0.22**	**2.86**
R2DM ($T = 256$)	215.27	128.74	0.72	3.79
R2DM ($T = 512$)	209.24	89.62	0.65	3.76
R2DM ($T = 1024$)	**207.31**	**70.34**	0.44	3.56

FRD is not available for the baselines [4, 5] which do not support the reflectance.



line in FPD. We believe that the performance gap with the KITTI-360 experiment lies in the setup of range images. In KITTI-360 experiments, the range images were downscaled to alleviate missing points called ray-drop noises. In contrast, the range images of KITTI-Raw were also downscaled but the ray-drop noises were retained to be closer to raw scan data. It is considered that there is room for further ingenuity to handle noisy settings, such as full resolution.