RadarSplat: Radar Gaussian Splatting for High-Fidelity Data Synthesis and 3D Reconstruction of Autonomous Driving Scenes

Supplementary Material

In the supplementary material, we first cover the radar sensing primer, radar spectral leakage modeling, and details of the radar used in this study (Sec. 1). Next, we provide additional method details and illustrations for noise detection, denoising, and occupancy grid mapping (Sec. 2). In Sec. 3, we elaborate on RadarSplat training and provide an overview figure. Sec. 4 presents evaluation details and additional quantitative and qualitative results. Finally, we discuss the limitations of our current method (Sec. 5).

1. Radar Sensing Primer

1.1. Frequency Modulated Continuous Wave (FMCW) Radar

In FMCW radar systems, the transmitted signal is a linear frequency-modulated chirp. The most common chirp is with the sawtooth pattern. The designed chirp slope is related to the bandwidth B and chirp duration T, where the chirp slope is $\frac{B}{T}$. See Figure 1 for more details [1].

Since the radar frequency changes over time, the wave travel time between the target and radar can be measured by the frequency difference between the emitted signal $s_{tx}(t)$ and the returned signal $s_{rx}(t)$, known as beat frequency $f_{beat} = f_{tx} - f_{rx}$, where f_{tx} and f_{rx} are frequencies of transmitted and received waves at time t. In practice, the beat frequency can be extracted from the intermediate frequency (IF) signal and a low pass filter (LPF).

Here, we consider a single target example to simplify the explanation. The IF signal is the complex mixing process between $s_{tx}(t)$ and $s_{rx}(t)$:

$$S_{IF}(t) = s_{tx}(t) \otimes s_{rx}(t) = \cos(2\pi f_{tx} t) \cdot \cos(2\pi f_{rx} t) = \frac{1}{2}\left[\cos\left[2\pi(f_{tx} - f_{rx})t\right] + \underbrace{\cos\left[2\pi(f_{tx} + f_{rx})t\right]}_{\text{later removed by LPF}}\right], \tag{1}$$

Next, the frequency-power signal, F, can be obtained by a Fast Fourier Transform (FFT), $\mathcal{F}$, also called a range FFT (Figure 1). Ideally, the beat frequency $\pm f_{beat}$ should have two impulses:

$$F(f) = \mathcal{F}\{S_{IF}(t)\} = \frac{1}{2}[\delta(f - f_{beat}) + \delta(f + f_{beat})] \tag{2}$$

In the end, the range-power signal is obtained by converting frequency f to distance D following:

$$D = \frac{cf}{2\mu} \tag{3}$$

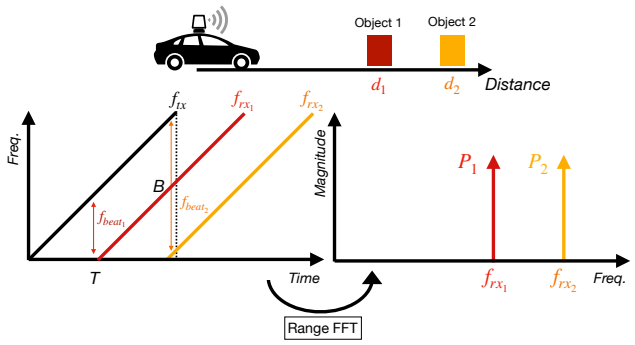

Figure 1. Radar imaging with range fast Fourier transform (FFT).

where $\mu = (B/T)$ is the radar chirp slope. The radar image we use as input and aim to synthesize consists of multiple range-power signals measured in different azimuth angles with a rotating mechanism.

1.2. Spectral Leakage Modeling

The spectral leakage is due to finite-time sampling of received signals, causing energy to spread across adjacent frequencies when doing range FFT. It transforms the target measurement into a Sinc function (Figure 2-b):

$$\mathcal{F}(f_{IF}(t)) = \text{sinc}(\omega T_s) \tag{4}$$

where $\omega = 2\pi f$ and T_s is the sampling duration of radar.

In practice, radar manufacturers apply a windowing technique to obtain sharp frequency cutoff and lower sidelobes. The Navtech radar used in our experiments uses Hamming windowing, which has a range FFT result shown in Figure 2-c. The distribution after windowing is:

$$\mathcal{F}(W_{\text{Hamming}}(f(t))) = T_s\left[0.54\,\text{sinc}\left(\frac{\omega T_s}{2\pi}\right) - 0.23\sum_{k=\pm 1}\text{sinc}\left(\frac{\omega T_s + 2k\pi}{2\pi}\right)\right]. \tag{5}$$

We observe that the radar signal after windowing is similar to a Gaussian distribution. Therefore, we use a Gaussian distribution to approximate the blurred effect, as shown in Figure 2-d. The width of the Sinc function in the frequency domain is:

$$f_w = \frac{2\pi}{T_s} \tag{6}$$

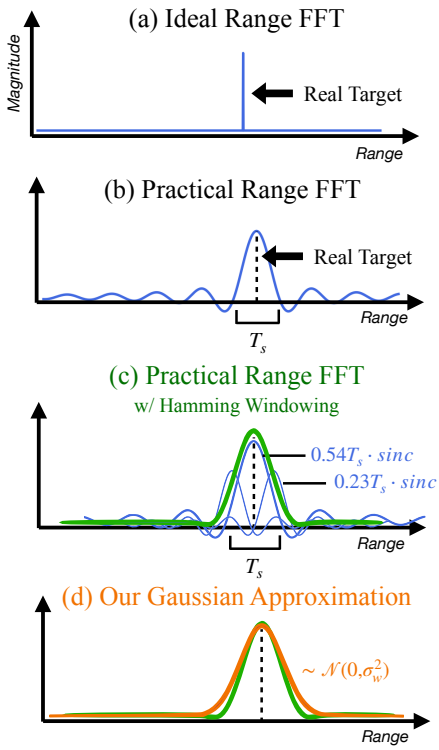

Figure 2. Modeling spectral leakage in the radar image. (a) Ideal range FFT. (b) Practical range FFT with spectral leakage. (c) The practical range FFT is sharpened using a Hamming window. The final distribution consists of three scaled and shifted sinc functions, as shown. (d) Proposed Gaussian approximation for Hamming-window-sharpened FFT.

where T_s is the sampling duration of the radar. The width of the Gaussian in meters d_w is derived with Eq. 3. We define the variance of the approximated Gaussian as $\sigma_w = (0.5 \times d_w)/3$ so that 99% of the Gaussian covers the width. Therefore the Gaussian approximation is $\sim \mathcal{N}(0, \sigma_w^2)$.

The radar used in this paper has $T_s = 565\mu$S and $\mu = 1.6 \times 10^{12}$. Therefore, we have $d_w \approx 1.04\ m$ and we set $\sigma_w = 0.17\ m$ in our experiments. Figure 3 illustrates the spectral leakage effect by visualizing overlapped radar image and LiDAR point cloud.

1.3. Scanning Radar Details

The radar used in this paper is Navtech CIR304-H from the Boreas Dataset [3]. The radar is operated at 4Hz scanning rate. The sequence we used has 0.0596 m range resolution and 0.9 horizontal resolution. The beam spread is 1.8° between -3 dB attenuation points horizontally and vertically. Additionally, the vertical antenna gain is designed with a cosec squared fill-in beam pattern, which enables a wider elevation field of view (FOV) to up to 40 degrees below the sensor plane. The radar antenna gain provided by the radar manufacturer is shown in Figure 4. The detailed numbers in dB is shown in Figure 5.

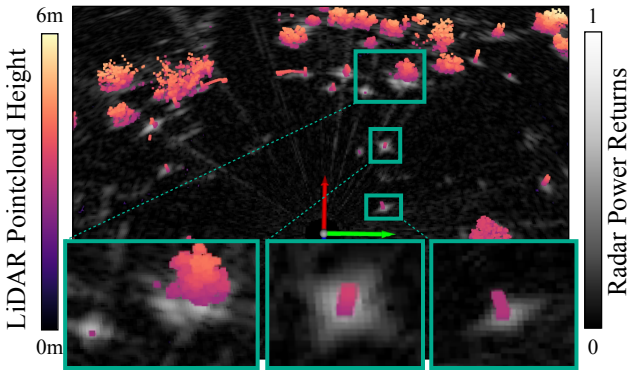

Figure 3. Illustration of blurred range-power signal due to spectral leakage. The overlapped LiDAR and radar data highlight the effect of radar spectral leakage.

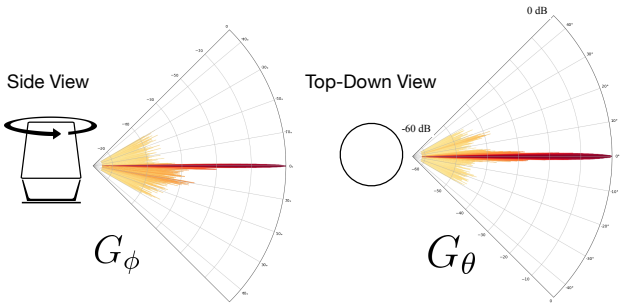

Figure 4. Visualize azimuth and elevation radar antenna gain.

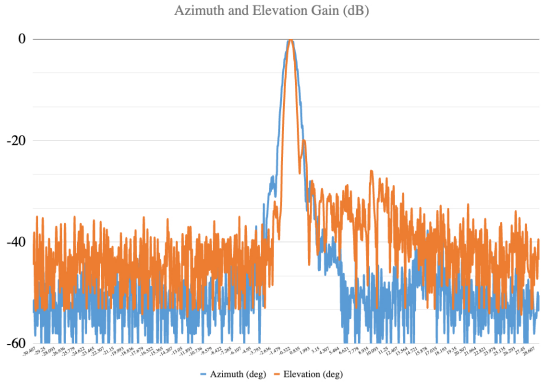

Figure 5. Antenna gain plot.

2. Noise Detection, Denoising, and Occupancy Mapping

2.1. Multipath and Saturation Noise Detection

Figure 6 shows the range-power signal and FFT of multipath, receiver saturation, and noise-free azimuth beams. The saturation beam is detected when constant ratio $C > C_{th}$. The multipath is detected when $C > C_{th}$ and $|X[k_m]| > A_{th}$.

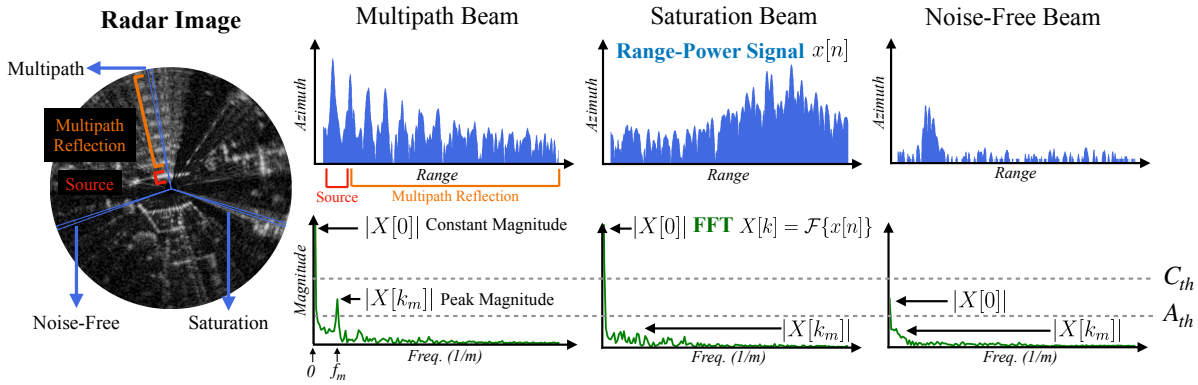

Figure 6. Illustration of the range-power signal of multipath, receiver saturation, and noise-free azimuth beam and their FFT. Also the constant and peak magnitude is shown along with the constant ratio threshold C_{th} for noise detection and with the peak magnitude threshold A_{th} for multipath detection.

2.2. Denoising

We reconstruct an initial estimated occupancy map to guide training, following [2]. Instead of relying solely on a dynamic threshold, our noise detection enables more robust noise removal, leading to improved initial occupancy maps for training. We propose a denoising algorithm that removes noise across detected noisy azimuth angles, $\theta_{noise} \in \Theta_{sat} \cup \Theta_{multi}$. We first apply Gaussian smoothing on the range-power signal to reduce the impact of high-frequency noise.

$$G_{\sigma_s}(n) = \frac{1}{\sqrt{2\pi\sigma_s^2}} e^{-\frac{n^2}{2\sigma_s^2}} \tag{7}$$

$$x_{smooth}[n] = (x[n] * G_{\sigma_s}), \tag{8}$$

where the σ_s is the variance of Gaussian, which we set to 5 bins in our implementation. Next, a masking region selection algorithm is applied to $x_{smooth}[n]$. We find the range index with the maximum magnitude and search the decay region to generate a noise-free mask region (n_s, n_e). Every bin n outside (n_s, n_e) region is set to zero. The pseudo-code is shown in Algorithm. 1. The denoised image is then used to construct the initial occupancy map.

Algorithm 1 Denoising algorithm with Decay Regions in Radar Range-Power Data

Require: $P(n)$ (1D array of radar power values), σ_s (Gaussian smoothing parameter)
Ensure: $n_{\max}$ (Index of maximum power), $\mathcal{D} = (n_s, n_e)$ (Decay region start and end indices)

$P_{\text{smooth}}(n) \leftarrow \text{GaussianFilter1D}(P(n), \sigma_s)$
$n_{\max} \leftarrow \arg\max(P_{\text{smooth}}(n))$
$n_s \leftarrow n_{\max}$
while $n_s > 0$ **and** $P_{\text{smooth}}(n_s - 1) \leq P_{\text{smooth}}(n_s)$ **do**
    $n_s \leftarrow n_s - 1$
end while
$n_e \leftarrow n_{\max}$
while $n_e < \text{length}(P_{\text{smooth}}(n)) - 1$ **and** $P_{\text{smooth}}(n_e + 1) \leq P_{\text{smooth}}(n_e)$ **do**
    $n_e \leftarrow n_e + 1$
end while
return (n_s, n_e)

2.3. Occupancy Mapping

To the handle occlusion issue, we use a W frames window to reconstruct an occupancy map for each frame. We set $W = 10$ in our experiment. We compute the power mean of each grid to identify the free and occupied space. A power threshold $p_{th} = 0.15$ is chosen to obtain a binary grid map for training supervision.

3. RadarSplat Implementation Details

3.1. Input Format

We set the maximum range of the input radar image to 50 m, with an azimuth resolution of 0.9° and a range resolution of 0.0596 m, resulting in an input image size of (400, 839). To ensure accurate error computation, we mask out the closest 2.5 m of radar data, as these measurements primarily originate from the ego-vehicle.

3.2. RadarSplat

The RadarSplat rendering pipeline, illustrated in Figure 7, consists of three key stages: elevation projection, azimuth projection, and spectral leakage modeling. In practice, we set $Q = 10$ in the elevation projection step, which results in the rendered elevation-projected image I_{Elev} having a size of $(4000, 839)$. The azimuth projection is implemented using a 1D convolution along the azimuth axis with a stride size of Q and circular padding. As a result, I_{Azi} has a size of (400, 839). In the end, the spectral leakage modeling is applied with Gaussian variance $\sigma_w = 0.17\ m$ derived from

Sec. 1.2.

For Gaussian Splatting, we initialize 2×10^4 Gaussians with occupancy probability $\alpha = 0.1$ and noise probability $\eta = 0.1$. The initial Gaussian size is set to 0.5 m with random initialization. The spherical harmonics level is set to 10.

3.3. Training Configuration

We set $\lambda_1 = 0.8, \lambda_2 = 0.2, \lambda_3 = 5, \lambda_4 = 10^2, \lambda_5 = 10^2$. RadarSplat is initialized with 2×10^4 Gaussians of size $s = 0.5\ m$ and trained for 3000 iterations. For multipath modeling, we set $C_{th} = 0.21, A_{th} = 0.3, C'_{th} = 0.2, r_{th} = 0.5m,$ and $\theta_{th} = 10°$. Gaussian rendering is set to $Q = 10$.

3.4. Cartesian-to-spherical Gaussian Conversion

The conversion is as follows:

$$\mu_{spherical} = \begin{bmatrix} r \\ \theta \\ \phi \end{bmatrix} = \begin{bmatrix} \sqrt{x^2+y^2+z^2} \\ \arctan 2(y,x) \\ \arcsin\left(\frac{z}{r}\right) \end{bmatrix} \tag{9}$$

$$\Sigma_{\text{spherical}} = J\Sigma J^T \tag{10}$$

where J is the Jacobian of Cartesian-to-spherical space conversion.

$$J = \begin{bmatrix} \frac{x}{r} & \frac{y}{r} & \frac{z}{r} \\ -\frac{y}{x^2+y^2} & \frac{x}{x^2+y^2} & 0 \\ -\frac{xz}{r^2\sqrt{r^2-z^2}} & -\frac{yz}{r^2\sqrt{r^2-z^2}} & \frac{\sqrt{r^2-z^2}}{r^2} \end{bmatrix} \tag{11}$$

4. Evaluation Details and Extra Evaluation

4.1. Scene Reconstruction Evaluation

We construct a LiDAR pointcloud map to obtain ground-truth geometry for evaluation. Similar to building an occupancy map, we use a W-frame window to reconstruct a local map for each frame, preventing occluded objects from being included in the ground-truth map. To further solve the occlusion problem and radar invisible objects in the scene (mostly tree leaves and sticks), we adopt a small radar power threshold of 0.1 to remove all the LiDAR points having corresponding radar measurements below the threshold. Also, the $1.8°$ elevation angle is applied when obtaining LiDAR ground truth for evaluation. Here are details about our scene reconstruction metrics:

Relative Chamfer Distance (R-CD). The Relative Chamfer Distance normalizes the Chamfer Distance by the maximum pairwise distance between the predicted point cloud, P, and the ground truth point cloud, Q. The Chamfer Distance is defined as:

$$CD(P,Q) = \frac{1}{|P|}\sum_{p\in P}\min_{q\in Q}\|p-q\|_2^2 + \frac{1}{|Q|}\sum_{q\in Q}\min_{p\in P}\|q-p\|_2^2 \tag{12}$$

The Relative Chamfer Distance is defined as:

$$R\text{-}CD(P,Q) = \frac{CD(P,Q)}{\max\limits_{q_i,q_j\in Q}\|q_i-q_j\|_2^2} \tag{13}$$

Accuracy. Accuracy is computed as the ratio of correctly matched points (both in precision and recall sense) over the total number of points in both clouds.

$$\text{Accuracy} = \frac{|\{p\in P \mid d(p,Q)<\tau\}| + |\{q\in Q \mid d(q,P)<\tau\}|}{|P|+|Q|} \tag{14}$$

where τ is the distance threshold and

$$d(p,Q) = \min_{q\in Q}\|p-q\|_2 \tag{15}$$

is the nearest neighbor distance from each predicted point to the ground truth. We set $\tau = 0.5$ in practice. The accuracy can be divided into precision and recall:

Precision. Precision measures the fraction of reconstructed points that are within τ of the ground truth:

$$\text{Precision} = \frac{|\{p\in P \mid d(p,Q)<\tau\}|}{|P|} \tag{16}$$

Recall. Recall measures the fraction of ground truth points that have a corresponding reconstructed point within τ:

$$\text{Recall} = \frac{|\{q\in Q \mid d(q,P)<\tau\}|}{|Q|} \tag{17}$$

Here we provide a more detailed evaluation with precision and recall, as shown in Table 1. The results show that we have significant improvement in both precision and recall compared to baseline. However, we observed that precision is relatively lower than recall, indicating that our occupancy estimation has more false positives than false negatives. We hypothesize that the primary reason for this is the reconstruction of certain structures that are occluded in the LiDAR point cloud but visible in the radar image.

Method	Scene Reconstruction				
	RMSE↓	R-CD↓	Acc.↑	**Precision↑**	**Recall↑**
Radar Fields	3.03	0.29	0.59	0.46	0.61
Ours	**1.81**	**0.04**	**0.91**	**0.71**	**0.94**

Table 1. Scene reconstruction evaluation with precision and recall metrics.

4.2. Scene-Separated Evaluation

In Table 2, we show RadarSplat's results in different weather and lighting conditions. The consistent results across diverse weather and lighting conditions also show robustness of our method.

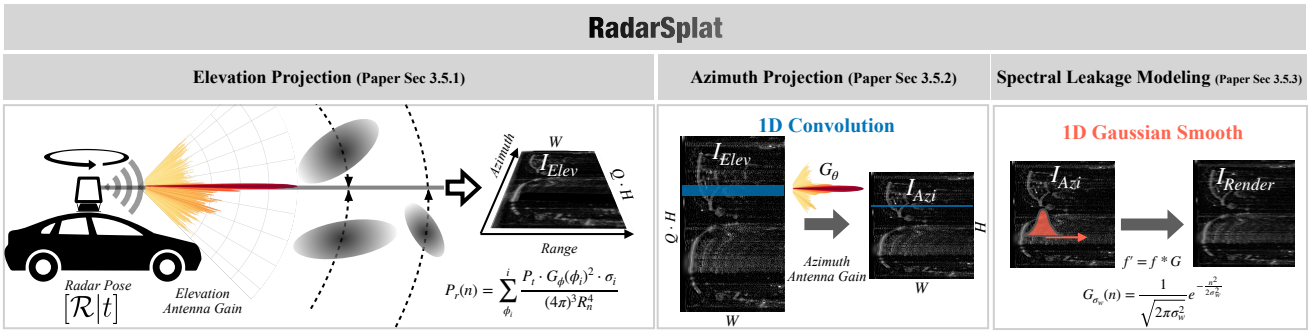

Figure 7. RadarSplat rendering. We first project Gaussians to 2D using the elevation antenna gain, then apply the azimuth antenna gain via a 1D convolution along the azimuth axis. Radar spectral leakage is modeled using 1D Gaussian smoothing along the range axis.

Scenes	Image Synthesis			Scene Reconstruction		
	PSNR↑	SSIM↑	LPIPS↓	RMSE↓	R-CD↓	Acc.↑
Sunny	25.81	0.50	0.37	1.78	0.04	0.90
Snowy	25.79	0.52	0.36	–	–	–
Rainy	26.59	0.49	0.39	1.63	0.03	0.93
Night	26.69	0.51	0.37	2.12	0.06	0.92

Table 2. RadarSplat Scene-separated evaluation

4.3. Ablation Studies for Multipath Modeling on Scenes With and Without Multipath Reflections

Table 3 show separate ablations on urban scenes with many multipath effects and natural scenes without multipath effects to quantify the contribution of multipath modeling.

Image Synthesis		Urban Scene			Natural Scene		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
RadarSplat	w/o Multipath	25.25	0.49	0.37	**27.27**	**0.52**	**0.37**
	Full Method	**26.06**	**0.51**	**0.37**	**27.27**	**0.52**	**0.37**

Table 3. In urban scenes, where multipath reflections are prominent, the modeling improves results. In contrast, in natural scenes with only trees surrounding the area, where multipath effects are minimal, the multipath modeling has negligible impact.

4.4. Ablation Studies on Occupancy Maps

We validate the impact of the proposed denoised occupancy map in proposed RadarSplat and Radar Fields. Table 4 shows both Radar Fields and our method benefit from the proposed occupancy map.

Scene Reconstruction		RMSE↓	R-CD↓	Acc.↑
Radar Fields	w/ RF Occ. Map	3.03	0.29	0.59
	w/ Proposed Occ. Map	2.68	0.11	0.72
RadarSplat	w/ RF Occ. Map	1.83	0.05	0.90
	w/ Proposed Occ. Map	**1.81**	**0.04**	**0.91**

Table 4. Comparing effect of ours and Radar Fields' occupancy map.

4.5. Ablation Studies on Initialized Gaussians

Table 5 and 6 show performance improves with increasing number and size of initialized Gaussians, saturating at the chosen 20k Gaussians and 0.5 m size.

Gaussians Num	5k	10k	20k	30k
PSNR ↑	25.31	25.79	26.06	26.05
Acc. ↑	0.88	0.90	0.91	0.91

Table 5. Gaussian number ablation.

Gaussians Size	0.1	0.3	0.5	0.7
PSNR ↑	21.73	25.14	26.06	26.20
Acc. ↑	0.65	0.90	0.91	0.91

Table 6. Gaussian size ablation.

4.6. Additional Results

In Figure 8, we present additional results, including the ground-truth camera view and the rendered reflectance. RadarSplat achieves superior image synthesis and occupancy estimation compared to the baseline [2]. For reflectance rendering, we take advantage of explicit Gaussian representation to segment out Gaussian that has low occupancy probability and high noise probability, resulting in a clearer object reflectance map. In addition, the videos of radar 3D reconstruction compared with ground-truth LiDAR are provided in the supplementary materials zip file.

5. Limitations

Occlusion Problem. Although radar provides bird-eye-view (BEV) power images with radar waves penetrating and bouncing off to see through occluded objects, occlusion can still happen in the radar image if the objects have high reflectivity. Figure 9 illustrates that a region behind the corner is occluded in most training views, resulting in wrong occupancy estimation.

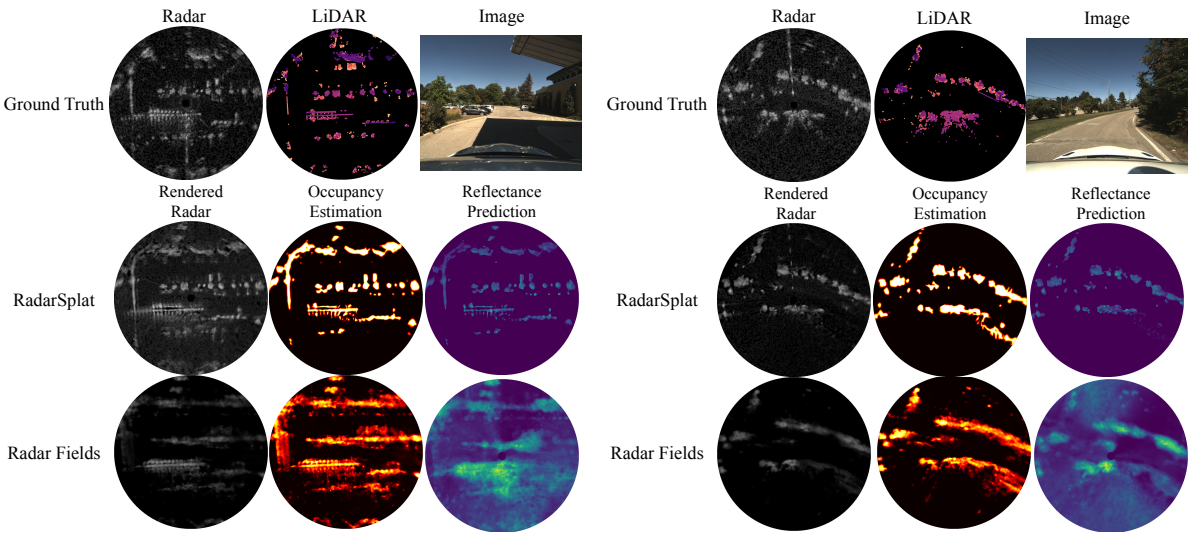

Figure 8. Additional results on Boreas Dataset.

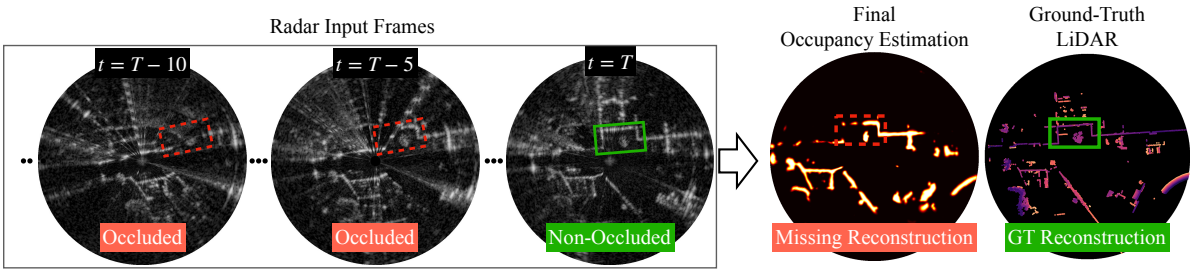

Figure 9. Wrong occupancy estimation caused by the occlusion problem.

Dynamic Objects. Similar to [2], our method does not consider dynamic object modeling. However, the Gaussian scene graph approach proposed in [4, 5] can be incorporated to model moving objects separately as individual Gaussian splats. These dynamic splats can then be combined with static Gaussian splats to construct a complete dynamic scene representation.

In the future, we plan to overcome these limitations by integrating occlusion modeling and the Gaussian scene graph into our method.

References

[1] Donald E Barrick. *FM/CW radar signals and digital processing*. Environmental Research Laboratories, 1973. 1

[2] David Borts, Erich Liang, Tim Broedermann, Andrea Ramazzina, Stefanie Walz, Edoardo Palladin, Jipeng Sun, David Brueggemann, Christos Sakaridis, Luc Van Gool, et al. Radar fields: Frequency-space neural scene representations for fmcw radar. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–10, 2024. 3, 5, 6

[3] Keenan Burnett, David J Yoon, Yuchen Wu, Andrew Z Li, Haowei Zhang, Shichen Lu, Jingxing Qian, Wei-Kang Tseng, Andrew Lambert, Keith YK Leung, et al. Boreas: A multi-season autonomous driving dataset. *The International Journal of Robotics Research*, 42(1-2):33–42, 2023. 2

[4] Yunzhi Yan, Haotong Lin, Chenxu Zhou, Weijie Wang, Haiyang Sun, Kun Zhan, Xianpeng Lang, Xiaowei Zhou, and Sida Peng. Street gaussians for modeling dynamic urban scenes. *arXiv preprint arXiv:2401.01339*, 2024. 6

[5] Xiaoyu Zhou, Zhiwei Lin, Xiaojun Shan, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. Drivinggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21634–21643, 2024. 6