CaRaFe: Camera-Radar Radiance Fields for Scene Reconstruction

DAVID BORTS, ETH Zürich, Switzerland
JULIAN OST, Princeton University, USA and Torc Robotics, USA
SHAMIK BASU, INRIA, France
TIM BRÖDERMANN, ETH Zürich, Switzerland
ANDREA RAMAZZINA, Mercedes Benz, Germany and Technical University of Munich, Germany
CHRISTOS SAKARIDIS, ETH Zürich, Switzerland
MARIO BIJELIC, Princeton University, USA and Torc Robotics, USA
FELIX HEIDE, Princeton University, USA and Torc Robotics, USA

ACM Reference Format:
David Borts, Julian Ost, Shamik Basu, Tim Brödermann, Andrea Ramazzina, Christos Sakaridis, Mario Bijelic, and Felix Heide. 2026. CaRaFe : Camera-Radar Radiance Fields for Scene Reconstruction. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers (SIGGRAPH Conference Papers '26), July 19–23, 2026, Los Angeles, CA, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3799902.3811192

This supplementary document provides additional information about the method introduced in the main manuscript, CaRaFe, and the data used to validate it. We describe our pre-processing steps for both radar and camera training data, elaborate on specific architectural design choices, and detail our evaluation methodology. This document is organized as follows:

Contents

1 Datasets and Data Pre-processing

In this section, we describe the datasets used for our qualitative and quantitative experiments and detail key pre-processing steps - both for our sensor measurements and auxiliary depth supervision signals.

1.1 The Boreas Dataset

The Boreas dataset [Burnett et al. 2023] was captured using a custom-built sensor rig, driven through the same route in suburban Toronto over the course of 1 year. As such, it captures the same locales in a diverse array of lighting, weather, and road conditions. Boreas includes captures from a Navtech CIR-304-H FMCW Radar, a FLIR Blackfly S Camera with 81° horizontal and 71° vertical FOV, a Velodyne Alpha-Prime LiDAR with 360° horizontal FOV, and an Applanix POS LV GNSS-INS for accurate sensor poses. Example data is visualized in Fig. 1.

This dataset has several unique features that make it ideal for validating our method. Specifically, Boreas includes raw radar FFTs - not just processed point clouds - which are essential for high fidelity radar neural rendering. Boreas is also an outdoor automotive dataset, which aligns with our problem setting of unbounded scene reconstruction from narrow-baseline captures. The inclusion of dense LiDAR data with full horizontal FOV allows for the construction of a pseudo-ground truth geometry and the computation of 3D metrics. Moreover, the repeated driving routes allow for benchmarking performance on the same scene in varying weather conditions.

1.2 Radar Data Pre-processing

We use a radar pre-processing algorithm to filter out sensor artifacts and noise. Empirically, we find that this is an essential step; doing so helps distill a more powerful geometric signal from the radar data and prevents dramatic floaters in our scene reconstructions. Our approach largely follows [Kung et al. 2025], but with several modifications.

Like RadarSplat, we detect and filter out beams with receiver saturation or multipath effects by thresholding in the frequency domain. We apply the FFT to each radar beam, dropping those that have a constant power offset ratio above threshold C_{th} (receiver saturation), as well as those with a peak magnitude above threshold A_{th} and lower constant ratio threshold C_{th} (multipath effects). Next, we also denoise our measurements by performing windowed radar grid-mapping with a window size of 10, accumulating returns into an average power measurement. Before accumulating FFTs into this map, we apply gaussian smoothing to each beam and mask out the decay region around the strongest return per-beam.

Unlike [Kung et al. 2025], which emphasizes accurate radar noise re-simulation, we do not supervise CaRaFe with extracted multi-path components nor explicitly model noise. Therefore, we deviate from their pre-processing steps by using the grid-mapped average denoised signals as a noise mask, and directly supervising CaRaFe with denoised FFT measurements, not the original noisy signals. Specifically, we threshold the averaged grid-mapping output into a binary mask, dilate it, and discard all FFT data outside of the dilated mask. Empirically, we found that this significantly improved 3D reconstruction quality.

1.3 Radar Depth Supervision

Before applying our soft-matching radar depth supervision loss, we need to extract depth targets from the radar FFT signals. We apply a simple peak-finding algorithm as a pre-processing step to extract up to k peaks per radar beam, where we set $k = 2$.

Before running peak finding, we first blur the beam FFT with a spatially-varying gaussian kernel that grows over range. We then discard any peaks below a minimum prominence and magnitude threshold and return the k closest peaks to the sensor. If 2 valid peaks are less than ϵ meters apart, we discard the shorter peak. In the case where less than k valid peaks exist, we still keep whatever peaks are valid and return those. The result is a sparse 2D radar point cloud that we can use for depth supervision or camera depth processing.

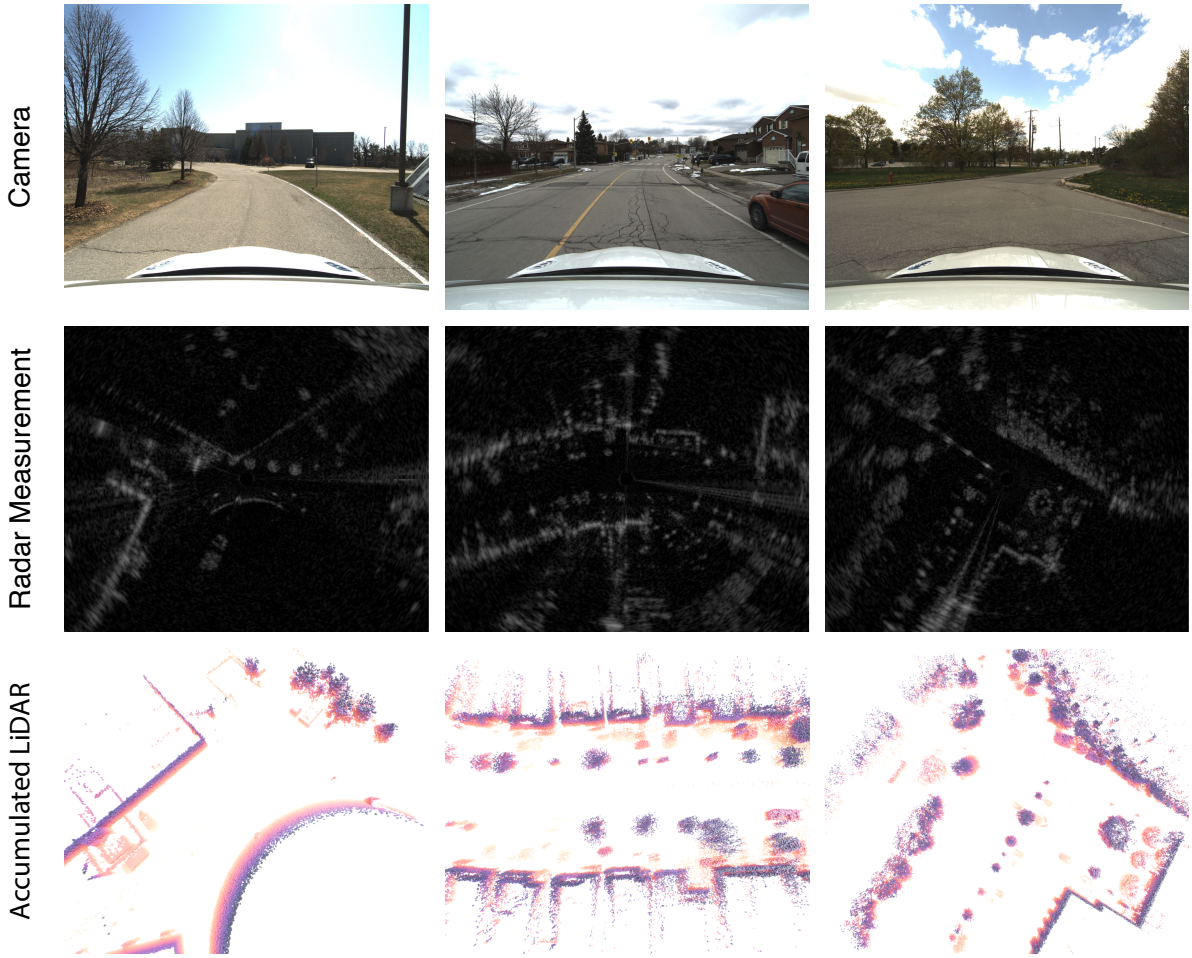

Fig. 1. Boreas Dataset Visualization. We show example captures from the dataset across sequences, including camera (top), radar FFTs (middle), and accumulated LiDAR point clouds (bottom), color-coded by height.

1.4 Monocular Depth Estimation and Radar-informed Pre-processing

CaRaFe supports additional depth supervision, both from dense camera or sparse radar. For the former, we leverage pre-trained camera monocular depth estimation models as a rich geometric prior for per-pixel predicted depth. However, these models can output warped or inconsistent depth predictions, especially on out of distribution scenes or camera intrinsics. They process each frame in a data sequence independently, without temporal modeling: supervising CaRaFe with these raw depth predictions can therefore degrade performance instead of improving it.

Since we have access to radar data, we leverage its robust metric depth to correct and filter out distorted camera depth estimates. We leverage the radar depth points computed in the previous section, accumulating a local radar point cloud within the region spanned by each camera frame's predicted depths. We follow [Rim et al. 2025] and compute a polynomial depth correction function per-frame, but estimate the polynomial coefficients analytically

rather then predict them with a learned model. We use a 3-degree polynomial, which we compute by projecting the predicted camera depth points onto the radar sensing plane and solving a least-squares correspondence between the radar depth points and the camera points. After correcting each camera frame's depth values individually, we pool them together for a final filtering step. We filter any remaining outliers by discretizing the scene into voxels, and mask out all camera points inside all voxels with fewer than 100 points total. The chosen side length of our voxels is 20 cm. As we found the error in many of these depth foundation models to increase with range, we exclude any predicted depth estimates greater than 50 meters from this process.

2 Additional Implementation Details

In this section, we elaborate on our architectural, representational, and hyperparameter choices for CaRaFe.

2.1 Architecture

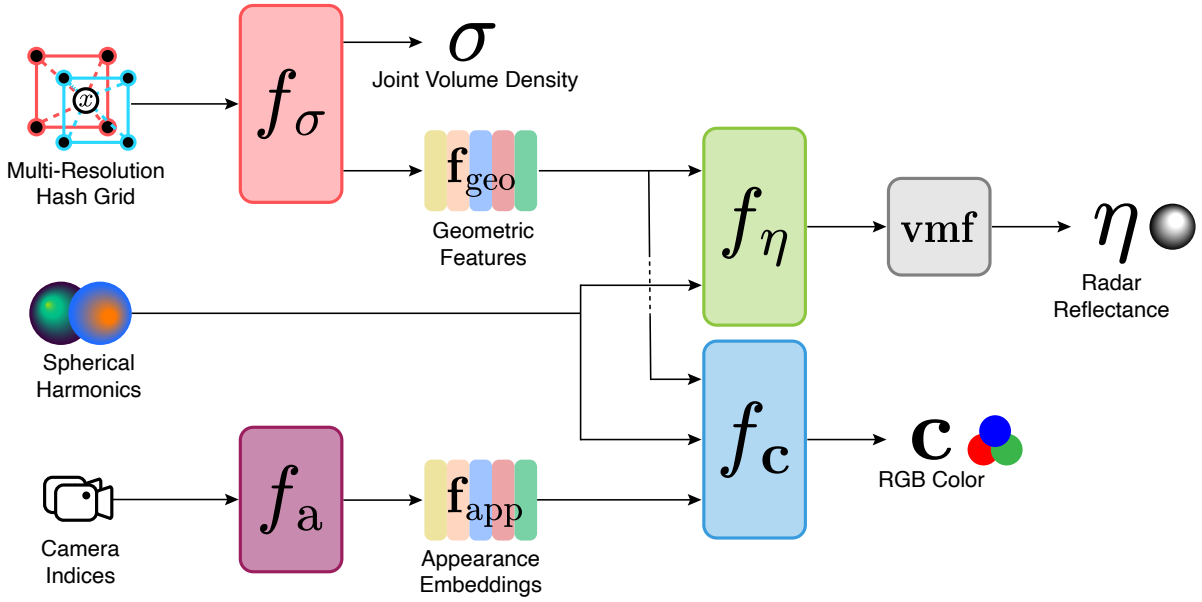

Fig. 2. CaRaFe core architecture. We represent the shared density head and modality-specific radiance heads with MLPs, while we use a simple embedding matrix for f_a.

We parametrize CaRaFe with 3 primary light-weight MLPs, offloading much of the representational capacity to the joint multi-resolution hash grid. We use the hash grid to compute positional embeddings on all 3D points before querying the density network, while for view direction we use a lightweight non-parametric spherical harmonics parametrization. In addition to the main MLPs, we also jointly learn 2 proposal sampler MLPs, each of which has its own lower resolution hash grid.

Camera Frame Appearance Embeddings. As shown in Fig. 1, the camera used for the Boreas dataset has automatic exposure compensation, leading to dramatic variations in lighting and color, even on the scale of individual frames. These artifacts pose the risk of degrading reconstruction fidelity if not adequately modeled, as the color variations will otherwise be absorbed into degenerate volume density throughout the scene. Therefore, we follow

[Yang et al. 2023] and introduce an optional camera frame appearance embedding into our architecture. While many variants exist, we opted for the simplest, which is a direct function of camera frame index alone. The resulting per-frame appearance embeddings $\mathbf{f}_{\text{app}}$ are learned jointly and concatenated with view direction and $\mathbf{f}_{\text{geo}}$ before being passed into the camera color head.

Scene Contraction. We also follow [Barron et al. 2022; Yang et al. 2023] in supporting query points from unbounded urban scenes by introducing scene contraction into our representation. We define an initial tight scene box, estimated from camera extrinsics, and pass any query points outside this box through an asymptotic spatial contraction function before querying our hash grid. This contraction function projects all possible query points onto the 2x unit cube. This is primarily to reduce geometric artifacts from extremely distant objects, like buildings on the horizon or the sky, which otherwise cannot be represented accurately in a finite scene box and are too far away to experience any meaningful parallax. Scene contraction allows for CaRaFe to deposit its density arbitrarily far away without needing to process prohibitively-large coordinates in half precision.

2.2 Hyperparameter Configurations

We largely follow state-of-the-art neural radiance field recipes [Barron et al. 2023; Müller et al. 2022; Yang et al. 2023] when selecting our hyperparameters. A notable choice is that we define different learning rates for our MLP networks versus our hash grid parameters. We include all our hyperparameter choices in Tab. 1.

3 Evaluation Details

In this section, we provide further detail on our procedure for evaluating CaRaFe and our baselines, both qualitatively and quantitatively.

3.1 Ground-truth Point Cloud Generation

To evaluate 3D scene reconstruction quality, we first need ground truth geometry. Since the Boreas dataset is in-the-wild and contains no true ground truth geometry, we rely on accumulated LiDAR points as a surrogate. We accumulate all LiDAR point clouds into world coordinates, apply ego motion compensation, and exclude any that fall outside our estimated scene box.

3.2 3D Occupancy Extraction

Before computing 3D geometry metrics, we first need to extract a 3D representation from CaRaFe. We do this by defining a grid of 3D scene points within the bounds $[(-70, -50, -h_m), (70, 50, 6.65)]$, in meters, where h_m is the mounting height of the radar. We then query CaRaFe at all of the target locations, and threshold the predicted volume densities to recover an occupancy point cloud.

3.3 3D Geometry Metrics

We report full Relative Chamfer Distance (RCD full) in our quantitative results for 3D geometry. We compute this as a scale-normalized Chamfer Distance:

$$RCD_{\text{full}}(A, B) = \frac{1}{2}\left(\frac{1}{|A|}\sum_{a\in A}\frac{d_{A\to B}(a)}{\|a\|_2^2} + \frac{1}{|B|}\sum_{b\in B}\frac{d_{B\to A}(b)}{\|b\|_2^2}\right),$$

where

$$d_{A\to B}(a) = \min_{b\in B}\|a - b\|_2,$$
$$d_{B\to A}(b) = \min_{a\in A}\|b - a\|_2.$$

Hyper-parameter	default
layers f_σ	2
layers f_η	2
layers $f_\mathbf{c}$	3
MLP hidden dim.	64
$\mathbf{f}_{\text{geo}}$ dim.	64
$\mathbf{f}_{\text{app}}$ dim.	16
max. Hash Grid	32768
MLP lr	1e-2
Hash Grid lr	2e-3
# proposal networks	2
proposal max. Hash Grid	(512, 2048)
λ_C	1.0
λ_R	0.2
λ_D	1e-3
λ_{reg}	1e-6
ϵ	5m
k	2

Table 1. Our choice of hyperparamaters for CaRaFe.

3.4 Novel View Synthesis Metrics

We report both PSNR and SSIM to assess novel view synthesis performance. However, the Boreas dataset has the hood of the ego vehicle in the bottom of camera frames. Therefore, we exclude these pixels from training supervision for CaRaFe and all camera baselines, and also mask them out before computing any image metrics.

4 Additional Validation

In this section, we validate CaRaFe's robustness to adverse conditions in Tables 2, 3. While camera measurements degrade in low-light and scattering media, radar measurements retain robust geometry, which provides our method with a stable signal to preserve geometric fidelity and constrain uncertainty from the camera. To evaluate both of these settings, we use selected sequences from the Radar Fields dataset [Borts et al. 2024], as the Boreas dataset [Burnett et al. 2023] does not have any night, rain, or fog captures. Altogether, these experiments demonstrate that CaRaFe inherits the robustness to adverse conditions from its radar inputs, despite deterioration of the camera input.

4.1 Robustness to Fog

CaRaFe maintains comparable performance on foggy Radar Fields sequences, with only a slight degradation in IoU, F-score, and RCD relative to the broader dataset. This is likely due to the lack of an explicit scattering media model that can fully explain the camera measurements, like in [Ramazzina et al. 2023]. The radar signal and our regularization mitigate smearing of the reconstructed geometry and floaters, allowing for robust performance even in scattering media. CaRaFe still outperforms all baselines in this setting, illustrating that performance does not degrade under compromised camera measurements.

Metric	Ours	EmerNeRF [Yang et al. 2023]	GOF [Yu et al. 2024]	PGSR [Chen et al. 2025]	RadarSplat [Kung et al. 2025]
IoU ↑	**12.98**	4.81	4.40	2.82	3.77
F-score ↑	**22.97**	9.18	8.43	5.49	7.27
RCD (full) ↓	**0.0031**	0.0039	0.0047	0.0050	0.0068

Table 2. Additional Geometric Validation on foggy sequences from the Radar Fields Dataset.

4.2 Robustness to Low-light

CaRaFe also maintains similar performance on low-light scenes. While IoU, F-score, and RCD both worsen slightly, the gap is smaller than with foggy scenes. Note that the night-time scenes in the Radar Fields dataset still have street light coverage, meaning that relevant geometry is not completely absent from camera images. Again, CaRaFe outperforms all baselines by a substantial margin, especially relative to GOF [Yu et al. 2024] and PGSR [Chen et al. 2025], which both suffered from degraded reconstructions in low-light.

Metric	Ours	EmerNeRF [Yang et al. 2023]	GOF [Yu et al. 2024]	PGSR [Chen et al. 2025]	RadarSplat [Kung et al. 2025]
IoU ↑	**13.47**	5.41	3.41	2.25	5.06
F-score ↑	**23.73**	10.27	6.59	4.40	9.63
RCD (full) ↓	**0.0021**	0.0030	0.0234	0.0157	0.0023

Table 3. Additional Geometric Validation on night-time sequences from the Radar Fields Dataset.

References

Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. 2022. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5470–5479.

Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. 2023. Zip-NeRF: Anti-Aliased Grid-Based Neural Radiance Fields. In *ICCV*.

David Borts, Erich Liang, Tim Broedermann, Andrea Ramazzina, Stefanie Walz, Edoardo Palladin, Jipeng Sun, David Brueggemann, Christos Sakaridis, Luc Van Gool, Mario Bijelic, and Felix Heide. 2024. Radar Fields: Frequency-Space Neural Scene Representations for FMCW Radar. In *ACM SIGGRAPH 2024 Conference Papers* (Denver, CO, USA) *(SIGGRAPH '24)*. Association for Computing Machinery, New York, NY, USA, Article 130, 10 pages. doi:10.1145/3641519.3657510

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. 2023. Boreas: A multi-season autonomous driving dataset. *The International Journal of Robotics Research* 42, 1-2 (2023), 33–42.

Danpeng Chen, Hai Li, Weicai Ye, Yifan Wang, Weijian Xie, Shangjin Zhai, Nan Wang, Haomin Liu, Hujun Bao, and Guofeng Zhang. 2025. PGSR: Planar-Based Gaussian Splatting for Efficient and High-Fidelity Surface Reconstruction. *IEEE Transactions on Visualization and Computer Graphics* 31, 9 (Sept. 2025), 6100–6111. doi:10.1109/tvcg.2024.3494046

Pou-Chun Kung, Skanda Harisha, Ram Vasudevan, Aline Eid, and Katherine A. Skinner. 2025. RadarSplat: Radar Gaussian Splatting for High-Fidelity Data Synthesis and 3D Reconstruction of Autonomous Driving Scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. 2022. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)* 41, 4 (2022), 1–15.

Andrea Ramazzina, Mario Bijelic, Stefanie Walz, Alessandro Sanvito, Dominik Scheuble, and Felix Heide. 2023. ScatterNeRF: Seeing Through Fog with Physically-Based Inverse Neural Rendering. *arXiv preprint arXiv:2305.02103* (2023).

Patrick Rim, Hyoungseob Park, Vadim Ezhov, Jeffrey Moon, and Alex Wong. 2025. Radar-Guided Polynomial Fitting for Metric Depth Estimation. arXiv:2503.17182 [cs.CV]

Jiawei Yang, Boris Ivanovic, Or Litany, Xinshuo Weng, Seung Wook Kim, Boyi Li, Tong Che, Danfei Xu, Sanja Fidler, Marco Pavone, and Yue Wang. 2023. EmerNeRF: Emergent Spatial-Temporal Scene Decomposition via Self-Supervision. arXiv:2311.02077 [cs.CV] https://arxiv.org/abs/2311.02077

Zehao Yu, Torsten Sattler, and Andreas Geiger. 2024. Gaussian Opacity Fields: Efficient Adaptive Surface Reconstruction in Unbounded Scenes. *ACM Trans. Graph.* 43, 6 (2024). doi:10.1145/3687937