RadarSim: Simulating Single-Chip Radar via Multimodal Neural Fields

Chuhan Chen[1] Tianshu Huang[12] Akarsh Prabhakara[3] Chaithanya Kumar Mummadi[2]
Zhongxiao Cong[1] Anthony Rowe[12] Matthew O'Toole[1] Deva Ramanan[1]

[1]Carnegie Mellon University [2]Bosch Research [3]University of Wisconsin–Madison
sally-chen.github.io/radar-sim

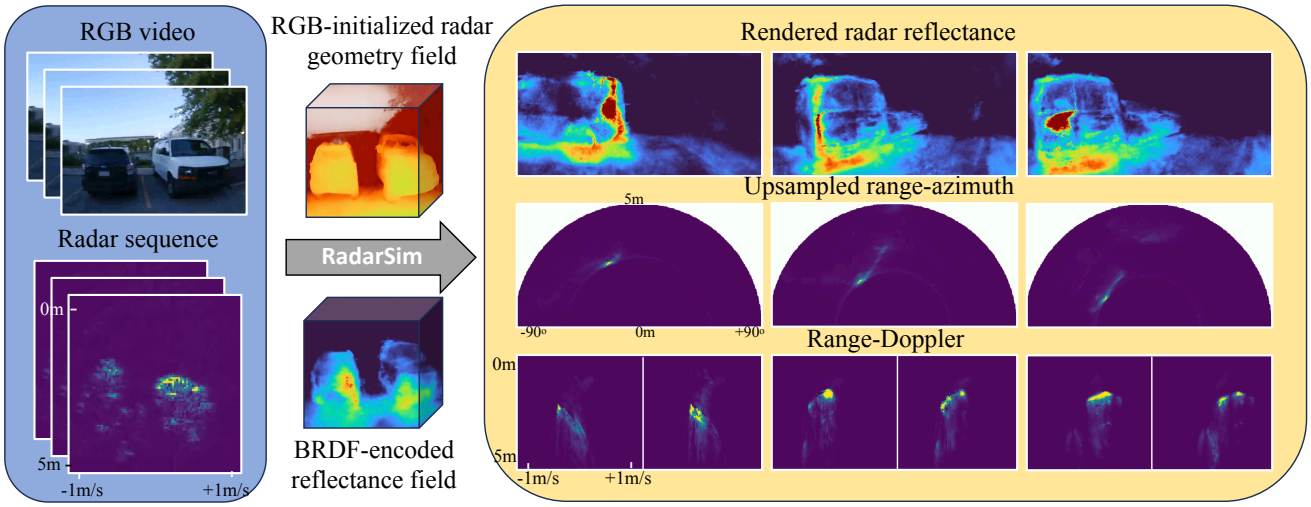

Figure 1. Given synchronized measurements from a mmWave radar and RGB camera, we learn a spectral field model for rendering "superresolution" radar reflectance **(top right)**, which has higher fidelity and interpretability than the input radar sequence **(bottom left)**.

Abstract

Radars are an ideal complement to cameras: both are inexpensive, solid-state sensors, with cameras offering fine angular resolution, while radars provide metric depth and robustness under adverse weather. However, radar data is more difficult to interpret than camera images and varies significantly between sensors, necessitating increased reliance on simulation for prototyping sensors and processing pipelines. Recent work treating radar reconstruction as a novel view synthesis problem has shown great promise in reconstructing radar-relevant geometry and simulating low-level radar data. However, such methods are constrained by the low spatial resolution of the underlying radar. To address this, we propose a unified differentiable renderer, RadarSim, which leverages the high angular resolution of RGB cameras to generate Doppler radar range images from a camera-initialized neural field. Using a novel data set of calibrated radar camera recordings from a custom hand-held rig, we demonstrate that RadarSim produces sharper geometry and Doppler range frames than radar-only reconstructions.

1. Introduction

Low-resolution single-chip mmWave radars are widely used in driver assistance [45, 57], collision avoidance [24], agriculture [51], and smart homes [58] due to their low cost, robustness, and ability to measure absolute range. However, unlike camera images or Lidar depth maps, raw range-Doppler radar data are difficult to interpret in 3D due to angular ambiguity and cross-bin effects like side lobes and bleed

[30]. Additionally, radar data are heavily sensor-dependent, making it impractical to test or train radar processing algorithms on generic datasets, unlike camera-based models using internet-sourced images. To address these challenges, many radar reconstruction techniques extract radar-relevant geometry from multiple radar scans [9, 27–29], while radar simulations [2, 21, 49, 50] help simulate new sensors, test algorithms, and augment datasets [5].

Vision-as-inverse graphics approaches inspired by Neural Radiance Fields [36] have shown great promise as data simulators. These approaches [6, 23] unify 3D reconstruction and simulation as a novel view synthesis problem, and can accurately recover high-resolution radar geometry and simulate radar scans.

However, these methods rely solely on radar data, which has inherently low spatial resolution. This limitation prevents the capture of fine geometric details, leading to blurred reconstructions and a loss of intricate features that cameras or LiDAR can easily capture,

thereby limiting their suitability for high-fidelity novel view synthesis. As a result, while radar-relevant geometry can be recovered, overall quality remains inferior to the state-of-the-art camera and LiDAR-based techniques [22, 36].

To bridge the gap between radar and camera-based reconstruction, we propose *RadarSim*, a unified differentiable renderer that combines radar's depth sensing with the high spatial resolution of cameras. It employs a differentiable multimodal scene representation to generate mmWave range Doppler frames with geometry initialized from a pretrained RGB neural field for enhanced detail.

Key challenges. While mmWave radars and RGB cameras largely share the same underlying spatial geometry, their properties can differ significantly. Radars process electromagnetic spectra at millimeter wavelengths, while visible light consists of spectra at nanometer wavelengths. This can cause dramatic differences in wave propagation across space and wave reflection at surfaces.

For instance, mmWave radars perceive glass as opaque but see plastic bodywork and thin walls as transparent.

Thus, radar-camera reconstruction must align their shared geometry while allowing for modality-specific differences.

Our key insight is that radar field geometry can be initialized and regularized with camera-field geometry (learned from a pre-trained RGB neural field). This allows our approach to preserve fine details provided by camera while accurately simulating radar measurements, ensuring high-resolution reconstruction with radar-consistent depth.

Moreoever, surfaces tend to appear more specular under the large wavelengths of radar, which is often manifested as view-dependant *retro-reflection* (Fig. 2). This provides an additional opportunity for information sharing: by representing the radar's view-dependence using a Bidirectional Reflectance Distribution Function (BRDF) relative to a (learned) surface normal, *RadarSim* can more accurately represent retro-reflective surfaces.

Dataset	Radar Type	Raw Data	Varying View Dir.
RadarSim (Ours)	**Low Res**	**Yes**	**Yes**
RADDet [61]	**Low Res**	**Yes**	No
RADIal [46]	High Res	**Yes**	No
K-radar [41]	High Res	**Yes**	No
Coloradar [26]	High Res	**Yes**	**Yes**
RaDICal [32]	**Low Res**	**Yes**	No

Table 1. **Comparison with other RGB + mmWave radar datasets with raw data.** We capture a dataset using a low-resolution single-chip radar and cover scene content from multiple views directions and positions.

Contributions. We propose *RadarSim*, the first multimodal neural field to combine radar with RGB modality in a unified framework. Our contributions are as follows:

(1) We introduce a camera-radar based framework that leverages camera geometry as a prior to learn radar-specific geometric properties (Sec. 3.2). We also propose a camera-initialized proposal network for radar ray-tracing which allows *RadarSim* to focus on surfaces and correctly model radar's ability to see through some camera-opaque materials (Sec. 3.3).

(2) Unlike prior work, we also model radar's specular retroreflectance using a novel BRDF-based encoding with learned surface normals, which provides further information sharing between camera and radar geometry (Sec. 3.4).

(3) We introduce a LiDAR-free metric scale optimization method that refines scale-less COLMAP-derived camera poses by leveraging structural cues from radar's range-Doppler data, ensuring accurate multimodal alignment and eliminating the need for expensive LiDAR calibration.

(4) Finally, due to a lack of multimodal camera-radar datasets catered toward low-cost (i.e., low-resolution) radars and multiview settings (see Tab. 3), we introduce a new radar-camera dataset and demonstrate that our multimodal architecture improves radar novel view synthesis both qualitatively and quantitatively, while also enhancing density estimation of occluded surfaces (Sec. 4).

2. Related Works

Data-driven radar simulation. While there exist model-based simulators [2, 8, 10, 18, 21, 34, 49, 50, 55] that simulate radar signals based on a known environment, we focus on data-driven radar simulation methods that infer environments from real radar measurements. Sparse methods detect individual reflectors using CFAR (constant false alarm rate) techniques [11, 37, 48]. In contrast, dense methods represent the environment as a voxel grid, estimating radar properties

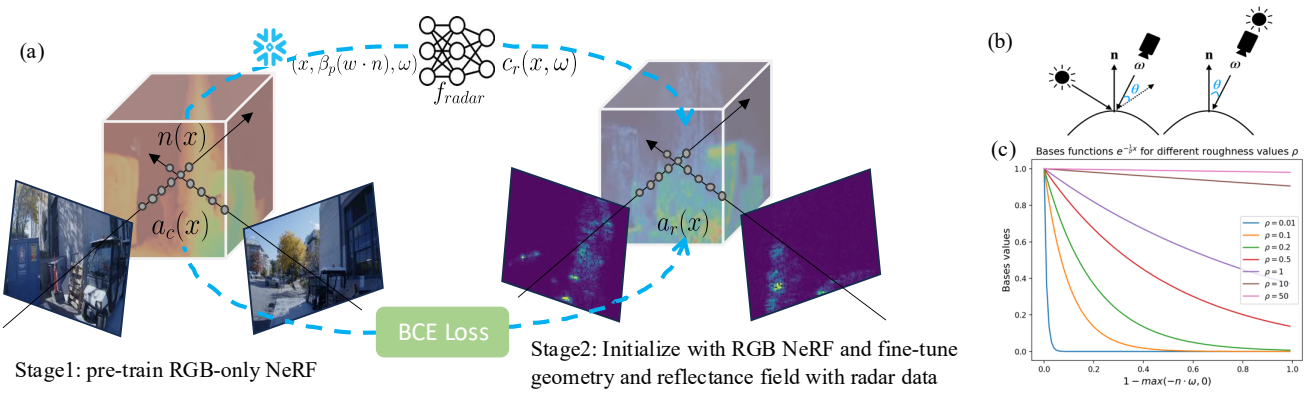

Figure 2. **(a)** *RadarSim* uses volumetric radar reflectance and occupancy models to render a high resolution radar reflectance image. Importantly, it initializes (and regularizes) the radar occupancy model to be similar to a pre-trained RGB-NeRF occupancy model. **(b) left:** To better model radar reflectance, we repurpose classic specular reflectance models (e.g. Phong shading [16]), where the strength of the viewed specularity depends on the angle between the viewing direction and the reflected light source (reflected about the surface normal). **(b) right:** Radars make use of co-located transmitters and receivers, implying the brightness of specular *retro-reflectors* will be determined by the angle between the viewing direction and surface normal. **(c)** We show BRDF basis functions used to capture the degree of view-dependant retroreflectance given by a "roughness" value ρ; for large ρ, there is little view-dependance, implying the surface has the same radar reflectance regardless of viewing angle.

for each cell. These dense methods can be either coherent (e.g., using Synthetic Aperture Radar [35, 38, 42, 44, 59, 60] with precise motion or fixed paths) or incoherent [27–29] (aggregating data without phase alignment). While SAR provides high resolution, it's typically unsuitable for large-scale mobile applications. Instead, incoherent aggregation—such as multi-view 3D reconstruction or radargrammetry offers a practical alternative by combining measurements from different views or sub-trajectories. Recently, deep learning based approaches such as [7, 15] learn generative models that simulate radar measurements from learnt radar data distributions. While fast, scalable and realistic, they are far less accurate than multi-view reconstruction based approaches which aggregate real measurement in the scene to simulate new views.

Neural fields for radar. While originally developed for photorealistic camera novel-view-synthesis, the neural-implicit inverse rendering approach pioneered by Neural Radiance Fields (NeRFs) [36] has also been extended to the radar domain. For example, DART [23] proposes a NeRF-like approach to simulate low-resolution mmWave radars in the range-Doppler domain using a multi-view sequence. Neural fields have also been proposed for other radar applications such as mechanical radars used in robotics and some autonomous vehicles [6] and synthetic aperture radars in in aerospace and remote sensing [12, 33].

Multimodal neural fields. In addition to radar, NeRF-like approaches have also been applied to a wide variety of domains such as RSSI [62], imaging sonar [43, 47] and Lidar [22]. Beyond single modalities, many have also proposed to incorporate different sensor modalities into a single neural field with conventional RGB cameras, including Lidar [20, 54, 63], thermal or infrared cameras [19, 40], and even language embedding semantics using a camera-like rendering model [3, 25].

Crucially, existing NeRF+X multimodal models all seek to fuse conventional image-based NeRFs with other modalities which also share a similar ray-based rendering model. This is not the case for radar: unlike cameras (or Lidar), whose sensor model has *range* ambiguity, radars trade absolute range resolution for *angular* ambiguity, resulting in an orthogonal sensor model [23].

3. Method

RadarSim builds upon DART [23], which can be viewed as a modification of implicit neural rendering engines (NeRF [36]) for radar. Intuitively, one can view *RadarSim* as a unification of DART (for radar) and NERF (for RGB); given a static scene captured by synchronized radar and camera measurements, we learn a unified neural field that stores volumetric quantities that enable rendering of both RGB and radar (range-Doppler) views. However, combining both modalities is challenging. Modeling radar requires fundamentally different sampling strategies, since range Doppler "pixel" measurements are generated by integrating along a circle in space rather than a ray (since radar waves propagate radially rather than along rays). Because of the differences in transmissive properties for radio waves and visible light, related but different geometric terms are required for camera and radar.

To capture such differences in a unified architecture, we learn a neural field for radar measurements with the help of information learnt from camera data as a regularizer. Specifi-

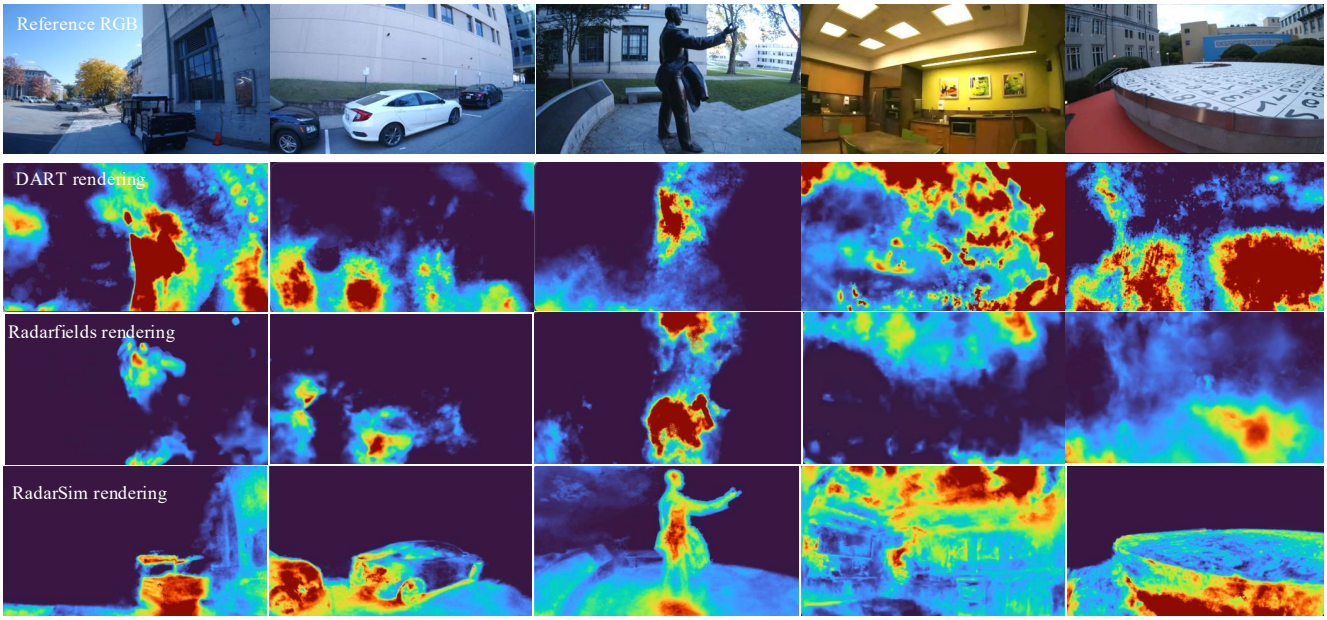

Figure 3. **Novel view synthesis for DART [23] and Radarfields [12] (middle) versus *RadarSim* (bottom)**. Since DART and Radarfields are based solely on radar data, it is limited in spatial, azimuth and elevation resolution and fine-detail. Specifically, they lack the ability to resolve reflectors at different heights because both doppler-range integration (DART) and range integration (Radarfields) still suffer from height ambiguity given limited data. In contrast, *RadarSim* combines radar with RGB camera data, and so captures sharper geometric details while faithfully recovering radar reflectance. In particular, *RadarSim* models radar's characteristic *retro-reflectance* (Fig. 2) as indicated by the strong responses for surfaces who's normal aligns with the camera-view (e.g., the rear of the truck), metallic surfaces, as well as concave structures like bottom of the car and corners.

cally, from a pre-trained camera-only neural field, we initialize a geometry encoder for radar as well a proposal network for generating samples for radar ray-tracing. We model radar reflectance with an MLP conditioned on learnt camera-based geometry embedding for learning high-frequency spatially varying details. Finally, because radar tends to reflect across metallic surfaces with strong view-dependence, we model the specular reflection (that depends upon both the viewing direction and surface normal) via BRDF basis functions, repurposing techniques from implicit BRDF modeling [56] for capturing radar retro-reflectance.

3.1. Background: NeRF and DART

Since *RadarSim* is an integration of these two frameworks, we begin by providing a unified overview of DART and NeRF; for additional details, we refer the reader to the original references [23, 36].

NeRF. NeRFs learn an implicit neural field that can be used to differentiable render an image (or 2D pixel grid) by integrating volumetric *radiance* (or color) $\boldsymbol{c}(\boldsymbol{x}, \boldsymbol{\omega}) \in [0,1]^3$ and density $\sigma(\boldsymbol{x}) \in R$ for each 3D point $\boldsymbol{x}$ and view direction $\boldsymbol{\omega}$ along pixel-aligned ray $\boldsymbol{Y}$ [36]:

$$\boldsymbol{C}(i, \boldsymbol{w}) = \boldsymbol{c}(\mathbf{x}_i, \boldsymbol{w})\alpha(\boldsymbol{x}_i)\prod_{j<i}(1-\alpha(\boldsymbol{x}_j)), \boldsymbol{Y} = \sum_i \boldsymbol{C}(i, \boldsymbol{w}) \tag{1}$$

where $\alpha \in [0,1]$ are alpha-compositing weights equal to $\left(1-\exp -\sigma(\boldsymbol{x}_i)\delta_i\right)$ and δ_i is the distance between adjacent samples on a ray.

DART. Similarly, DART learns an implicit neural field that can be used to render radar measurements, which are naturally represented as a *3D cube* of range, speed (or Doppler), and angle (or antenna) measurements. To do so, DART integrates volumetric *reflectance* $s(\boldsymbol{x}, \boldsymbol{\omega}) \in R$ and *transmittance* $t(\boldsymbol{x}, \boldsymbol{\omega}) \in [0,1]$, capturing the proportion of energy that reflects back and that continues past a point $\boldsymbol{x}$. These quantities can be used to model the radar return amplitude at point sample $\boldsymbol{x}_i = \boldsymbol{x} + r_i\boldsymbol{\omega}$ observed by a radar at position $\boldsymbol{x}$ with antenna k, written as $C(i, k, \omega)$:

$$C(i, k, \boldsymbol{\omega}) = \frac{g_k}{r_i^2}\boldsymbol{s}(\boldsymbol{x}_i, \boldsymbol{\omega})\prod_{j<i} t(\boldsymbol{x}_j, \boldsymbol{w})^2, \tag{2}$$

where i is a discrete range bin. Compared with (1), transmittance can be seen as $1-\alpha$, but is squared since the radar signal is attenuated twice along the ray, during both the outgoing and and incoming directions after reflection. The additional inverse squared fall-off captures the radiometric reduction of energy in the reflected signal, while the antenna-dependent gain factor g_k captures the dependence of the observed signal on the orientation of radar array.

Importantly, instead of accumulating values along a ray,

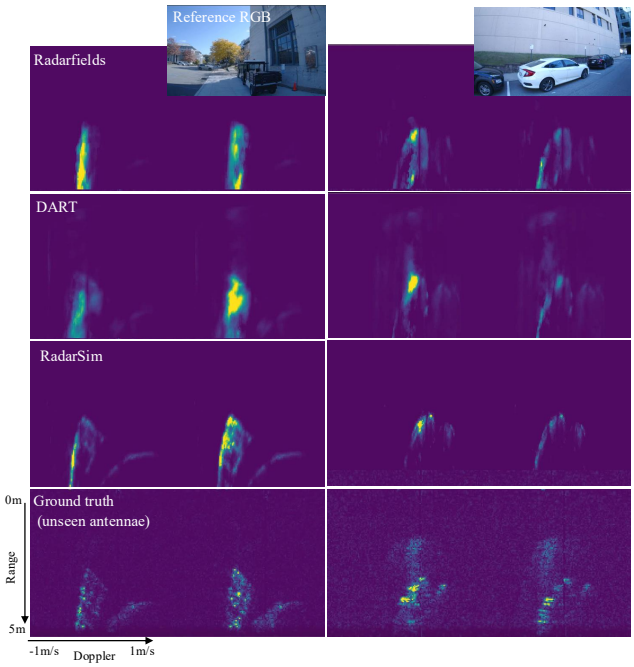

Figure 4. **Simulating unseen antennae.** *RadarSim* can generate novel-views with modified "intrinsics" that capture novel configurations of antennae. Here, we train *RadarSim* on the first 4 of 8 available antennae and generate renderings of the last 4, comparing them to held-out ground-truth antennae observations. We visualize 2 of the 4 unseen antennae in this figure. *RadarSim*'s renderings are sharper and closer to the ground truth. Fig. 5 uses the same approach to generate simulations of 128 antennae, to increase the angular azimuthal resolution of the radar.

we must generate (or "render") range-Doppler measurements, where the Doppler velocity of an object is its relative radial velocity. In particular, the apparent Doppler of a static point with viewing angle $\boldsymbol{w}$ captured by a moving radar with velocity $\boldsymbol{v}$ is $\langle \boldsymbol{w}, \boldsymbol{v} \rangle$: points directly in front have an apparent speed of $-||\boldsymbol{v}||$, but those off-center will have a cosine fall-off. Thus, to render a particular "pixel" for range r_i and Doppler d_j, we integrate samples that lie at the intersection of a *cone* of directions $\boldsymbol{w}$ (given a particular cosine fall-off of $d_j = \langle \boldsymbol{w}, \boldsymbol{v} \rangle$) with a *sphere* of radius r_i. Geometrically, this intersection is a circle in 3D [23]:

$$\boldsymbol{Y}(r_i, d_j, k) \propto \frac{r_i}{||\boldsymbol{v}||_2} \int_{\langle \boldsymbol{w}, \boldsymbol{v} \rangle = d_j,\ ||\boldsymbol{w}||_2 = 1} C(i, k, \boldsymbol{w})\, d\boldsymbol{w} \quad (3)$$

where the additional factors correct for the varying width of the spherical (range-Doppler) bins.

3.2. Sharing Geometry

Given our background models, we can now define our *RadarSim* architecture. Succinctly, we build a unified implicit neural field that generates volumetric geometry and reflectance quantities needed to render range-Doppler sensor measurements from a camera-only neural field. Two baselines (to which we compare in our ablations) are learning two separate neural fields with no sharing, as well as learning a single geometric neural field that is "fully-shared" across camera and radar. The former does not allow radar to benefit from cameras, while the latter does not model the fact that geometric transmission is different across the two modalities. Instead, we first train a camera-only neural field and learn a radar geometry encoder that is regularized to be similar (but not identical) to camera geometry. But to do so, we reconcile an inconsistency between the two formulations: unlike NeRF (Eq. 1), which fully separates geometry and radiance, DART implicitly captures scene geometry in its reflectance $s(\boldsymbol{x}_i, \boldsymbol{\omega})$ as well (Eq. 2).

Similar to other neural rendering approaches for active sensors [1], we separate reflectance into a geometry-independent reflectance term $c_r(\boldsymbol{x}, \boldsymbol{\omega})$ that captures how much energy is reflected (akin to radiance in NeRF) and a geometric-only term capturing radar-specific density $\alpha_r(\boldsymbol{x})$ that is equivalent to $1 - t(\boldsymbol{x}, \boldsymbol{\omega})$. This allows us to rewrite Eq. 2 in a form analogous to Eq. 1:

$$C(i, k, \boldsymbol{w}) = \frac{g_k}{r_i^2} c_r(\boldsymbol{x}_i, \boldsymbol{w}) \alpha_r(\boldsymbol{x}_i) \prod_{j<i} (1 - \alpha_r(\boldsymbol{x}_j))^2, \quad (4)$$

Geometry encoder. Given the modified formulation above, we now define our shared geometry encoder. We learn two neural fields for camera and radar:

$$(\alpha_c(\boldsymbol{x}_i), \boldsymbol{l}_{geo_c}) = f_{geo_c}(\boldsymbol{x_i}; \theta_{geo_c}) \quad (5)$$

$$(\alpha_r(\boldsymbol{x}_i), \boldsymbol{l}_{geo_r}) = f_{geo_r}(\boldsymbol{x_i}; \theta_{geo_r}) \quad (6)$$

in the form of multi-resolution hash tables [39] that store geometry codes $\boldsymbol{l}_{geo_c}$ and $\boldsymbol{l}_{geo_r}$ and capture geometric properties for camera and radar respectively. These codes are MLP-decoded into radar density $\alpha_r(\boldsymbol{x})$ and camera density $\alpha_c(\boldsymbol{x})$, respectively. The density heads are implemented as linear layers atop a shared MLP decoder. We first train f_{geo_c} with camera data, and distill f_{geo_c} into f_{geo_r} by initializing θ_{geo_r} with θ_{geo_c}. Then θ_{geo_c} is frozen and θ_{geo_r} is fine-tuned with radar data while constrained through a binary cross entropy loss between $\alpha_r(\boldsymbol{x}_i)$ and $\alpha_r(\boldsymbol{x}_i)$.

3.3. Radar Ray Sampling

Performance of state-of-the-art NeRF architectures such as [4] can be attributed to efficient importance sampling on ray-surface intersections. Extending on the proposal network used in [52][4] that generates samples from density stored in a light weight network self-supervised by the rendering weight of NeRF, we propose to share the proposal network between radar and camera and fine-tune a pre-trained proposal network for camera with rendering weight distribution

of radar. While in DART [23], samples on radar rays are generated linearly according to range bins, we generate samples based on the sampling distribution from the proposal network, and query f_{geo_r} and f_{radar} to obtain α_r and c_r for each sample on a ray. In case there are multiple samples assigned to a particular range bin, we aggregate the samples by taking the mean of the sample values; and if there are no samples, we assign 0 for α_r and c_r. We show the effect of such shared sampling scheme in geometry improvement for radar in Fig. 9 and reconstruct geometry behind occluded surface in Fig. 8.

3.4. BRDF Encoding

We now describe improvements to our radar reflectance model $c_r(\boldsymbol{x}_i, \boldsymbol{w})$ that leverage improved estimates of geometry. Our motivation is that many metallic surfaces appear highly specular under radar due to its large wavelength, a phenomena sometimes known as retroreflectance. Our key insight here is to repurpose innovations from the NeRF literature on capturing surface reflectance models (BRDFs), to better model retroreflectance common in radar sensing. To do so, we augment our model to explicitly reason about surface normals and surface roughness.

Surface normals. While normal maps could be derived by computing the spatial gradient of our geometric density model, such estimates are noisy in practice. We instead learn a MLP that predicts normals which is supervised by a monocular normal predictor on our input images [17].

Surface roughness. Classic models of specularity compute the dot product between the viewing angle and angle of reflectance from an incident light source, where the angle of reflectance is computed by mirror-flipping the incident angle across a surface normal (Fig. 2). However, for radars, where transmitters and receivers are collocated, the viewing and source angle are identical, implying that the quantity of interest is the dot product between the viewing angle $\boldsymbol{\omega}$ and surface normal $\boldsymbol{n}$. Retroreflective surfaces generate strong returns when viewed fronto-parallely, with a response that falls when viewed off-angle. To capture different rates of fall off, we make use of spectral basis functions

$$\beta_\rho(\boldsymbol{\omega} \cdot \boldsymbol{n}) \equiv e^{-\frac{1}{\rho}(1-\max(-\boldsymbol{\omega}\cdot\boldsymbol{n},0))}, \quad \rho \in P. \tag{7}$$

We now can define our final model of radar reflectance:

$$\boldsymbol{c}_r(\boldsymbol{x}_i, \boldsymbol{\omega}) = f_{radar}(\boldsymbol{l}_{\boldsymbol{geo_r}}, \{\beta_\rho(\boldsymbol{\omega} \cdot \boldsymbol{n})\}, \boldsymbol{\omega}; \theta_{radar}) \tag{8}$$

3.5. Scene Scale Optimization

In order to optimize f_{geo_r} and f_{radar} using doppler integration proposed in DART [23], we require knowledge of the true metric sensor velocity. Because COLMAP-derived sensor poses are scaleless, it is important to obtain metric scale for each scene. We observe that for scenes of different scales, range-doppler images contain structures that appear more expanded or compressed on the range axis as shown in Fig. 6, hence we propose to leverage our geometry sharing scheme to perform scale optimization by exploiting such structure information. From a pre-trained f_{geo_c}, we render range-doppler frames using α_c, and use the difference in shape of the structures alone between the ground truth and synthesized radar frames to optimize for scale using only a Structural Similarity Index Measure (SSIM) loss. Refer to Fig. 6 to the optimization process and Appendix for evaluation of our learnt metric scale.

4. Experiments

We validate the efficacy of *RadarSim* through experiments on a diverse set of indoor and outdoor scenes.

Dataset. We used a handheld data collection rig with a radar and camera for data collection. The rig's compact and portable design enabled us to collect 8 sequences each between 3-5 minutes duration in a diverse set of indoor and outdoor environments; we provide detailed descriptions of the data collection rig and collected traces in the Appendix.

Baselines. We evaluate our multimodal framework against DART [23] and Radarfields [6], the state of the art methods for radar-based 3D reconstruction, on the quality of reflectance and transmittance field learned by *RadarSim*: rendering out reflectance by tracing rays from camera pixels into the 3D neural field and accumulating reflectance using rendering weight computed with learnt radar density. We also evaluate on radar novel view synthesis in range-doppler frame quantitatively and qualitatively which shows the effectiveness of modelling radar reflectivity by reproducing the input signal while being capable of generalizing to unseen radar frames. We demonstrate our model can create high resolution visual rendering of radar reflectance and density, essentially a high resolution radar, from the same input low resolution radar frames. We also compare RadarSim with DART [23] baselines (CFAR, Lidar, Nearest Neighbor) in Tab. 2, demonstrating improvements over them.

4.1. Qualitative Results: High Resolution Radar Simulation

We visualize the quality of *RadarSim* against DART in Fig. 3. Our multimodal framework creates a high resolution rendering of radar reflectance field by leveraging shared geometry with RGB reconstruction, achieving a much better radar simulator than baselines. We are able to show structures that radar strongly reflects off such as retro-reflector like structures such as inset corners, bottom of cars, light on the ceiling. We also demonstrate surface normal-dependent specular reflection, e.g., when we point toward a surface,

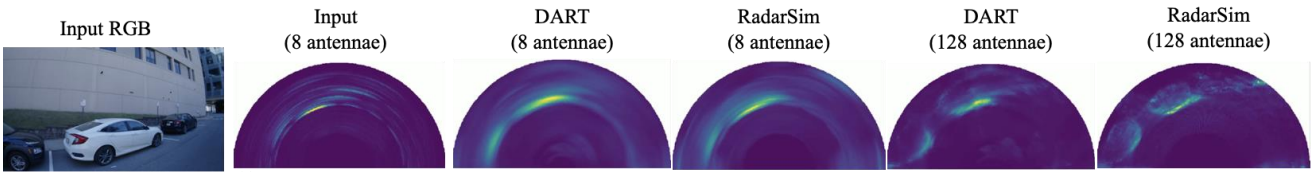

Figure 5. We show range-azimuth reconstructions of *RadarSim* and DART [23]. Recall our input radar data is recorded with 8 antennae, which limits the angular azimuth resolution. We can use our neural reconstructions to construct plots rendered with virtual radars with any number of antennae, allowing us to "super-resolve" additional detail across azimuth angles (e.g., we can distinct parked cars with even more detail than DART).

reflectance is strong. This is better viewed in our included videos in Supplemental materials. We are also able to distinguish materials such as metal, which is usually a strong reflector of radar, from walls, which are weak reflectors, as well as materials that radar signals transmit through such as glass from other non-transmissive materials. We further visualize synthesized range-Doppler frames for antennae held-out during training compared to ground truth and those synthesized by baselines in Fig. 4. *RadarSim* 's simulation appears most similar to the ground truth, showing its superiority in generalizing to unseen view directions and radar intrinsics. We also visualize synthesized 8-antenna and simulated 128 antenna range-azimuth radar frames in Fig. 5, showing *RadarSim* 's ability to reconstruct sharper geometry and radar reflectance field and effectiveness in improving azimuth resolution of single-chip radar.

4.2. Quantitative Results

To evaluate radar novel view synthesis, we applied our model to a holdout test set consisting of the last 20% of each sequence, and computed the Structural Similarity Index (SSIM) and Peak Signal-to-Noise Ratio (PSNR)[1] of the synthesized range-Doppler frames against their respective ground-truth frames. *RadarSim* achieves higher PSNR and SSIM compared to baselines (Table 2). We further evaluate view extrapolation by splitting each sequence spatially, where our model out-performs baselines by a big margin, showing our model's effectiveness in leveraging camera information for generalizable radar simulation.

4.3. Density Estimation for Occluded Surfaces

Radar has the capability to penetrate certain materials that are opaque to RGB cameras, such as cloth, cardboard, and foam. Because our multimodal model estimate different geometry for radar and RGB camera, we show in Fig. 7 and 8 that utilizing our multimodal model, we could estimate "emptiness" of occluded surfaces in high fidelity.

[1]Similar to [23], we also account for the sparsity of range-Doppler frames by ignoring regions of the range-Doppler image which are under a per-sequence noise threshold; refer to the Appendix for the procedure we used to calibrate this threshold.

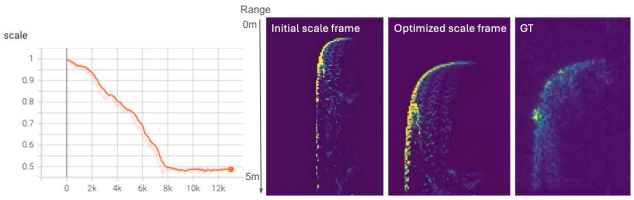

Figure 6. **Scale optimization process.** Recall that our pipeline uses COLMAP to infer up-to-scale camera poses. We use radar to metrically-upgrade our scene reconstruction by optimizing for the scale that produces the best (metric) range-doppler reconstruction. The (**left**) plot shows the optimization curve, where the scale factor adjusts from a randomly initialized value of 1 to a final optimized value 0.483. The (**right**) images display range-Doppler renderings with the initialized scale of 1, the optimized scale of 0.483, and the ground truth range-Doppler frame with a scale of 0.503. The optimized scale deviates from the ground truth by only 3.98%.

4.4. Ablations

Performance when not using BRDF bases and sampling We analyze the effect of BRDF bases and proposal network for radar. As shown in Fig. 9, our BRDF bases models sharp reflectance change based on angle between normal and view direction, while SH view direction encoding is much lower frequency. We also ablate on the effect of our shared proposal network with radar to generate radar ray samples; it produces sharper geometry for *RadarSim*.

Ablations with "naive" RGB-radar baselines Tab. 2 shows two additional variants of a "naive" baseline: radar/RGB images rendered with the same occupancy predicted from a shared geometry encoder, using the same Nerfacto [53] and DART [23] rendering process (excludes our geometry sharing (Sec. 3.2), BRDF encoding (Sec. 3.4), and ray sampling (Sec. 3.3) solutions.

- **RGB-only geometry:** Geometry encoder is pre-trained only by RGB loss, frozen and queried by the radar MLP that learns to predict reflectance. Performance drops significantly showing it is insufficient to "use" the geometry from RGB camera because radar has its unique transmissive properties.
- **Fully-shared geometry:** Same as above, but geometry encoder is jointly trained using both RGB and radar losses. This remains inferior to RadarSim showing our proposed

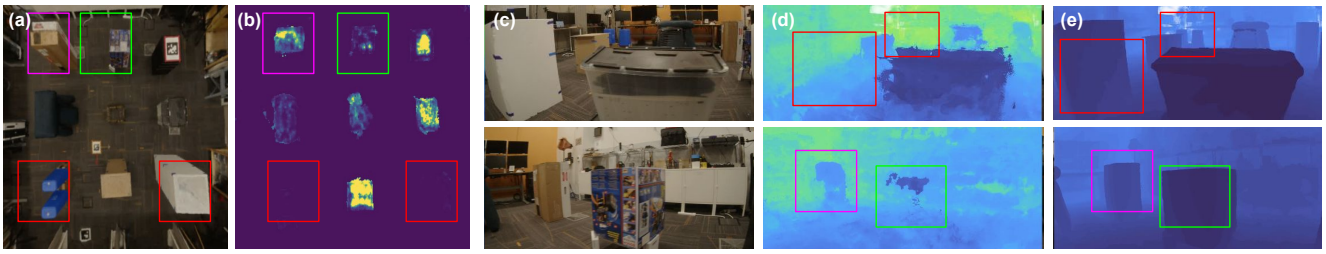

Figure 7. RGB birds-eye view of the scene (**a**), radar occupancy α_r slice (at 0.5m in height) (**b**) reconstructed by *RadarSim*. Reference RGB images (**c**) and corresponding depth map rendering using radar occupancy (**d**) and camera occupancy (**e**). Because radar transmits through materials such as plastic cardboard, foam, etc., such geometries (annotated in **red**) do not appear in the radar occupancy slices or depth renderings. We also compare boxes in purple (cardboard box with electronics) and green box (empty cardboard box), which the empty box does not show up in radar occupancy map or depth map rendered with radar occupancy.

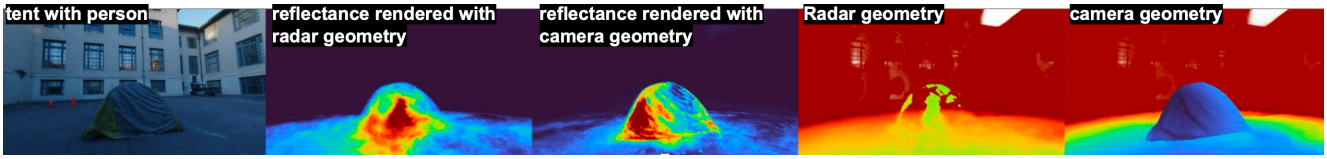

Figure 8. A tent with (**top**) a person sitting inside, shown as an RGB image (**left**), radar reflectance rendered from radar occupancy (**left-center**) and camera occupancy (**center**), depth map rendered from radar occupancy (**right-center**), and camera occupancy (**right**). As radar can transmit through cloth, radar density reveals the presence of a person, while camera density is unable to do so.

	Comparison to Prior Art	SSIM	PSNR
	RadarSim	**0.821**	**29.08**
	Radarfields [6]	0.771	27.80
	DART [23] + pose opt	0.799	28.47
	DART [23]	0.784	28.00
DART baselines	CFAR points	0.671	24.45
	Lidar occupancy	0.733	28.30
	Nearest Neighbor	0.725	25.36
	Diagnostics	SSIM	PSNR
Extreme novel-views	DART[23] + pose opt	0.747	27.14
	RadarSim	0.771	28.05
Geometry ablations	RGB-only geometry	0.771	28.36
	Fully-shared geometry	0.798	28.53
Architecture ablations	w/o Bases	0.802	28.78
	w/o Sampling	0.805	28.82

Table 2. **Top:** *RadarSim* outperforms radar-only prior art such as DART and RadarFields. We also compare to the baselines such as lidar occupancy and nearest neighbors. **Bottom:** Diagnostic Analysis. We find even stronger performance deltas for extreme novel-views, by spatially splitting up scenes into a train-vs-test split (instead of splitting up scene logs by timestamp, as above). We also compare to a "naive" variant where radar occupancy is fixed to be identical to the pre-trained RGB model. Optimizing such a fully-shared model for both RGB and radar reconstruction helps somewhat, but still unperforms *RadarSim*. Finally, removing the reflectance model or the radar ray sampler modestly hurts.

solutions effectively help reproduce radar reflectance.

5. Conclusion

We propose *RadarSim*, which leverages the complementary properties of radar and cameras with a unified differentiable

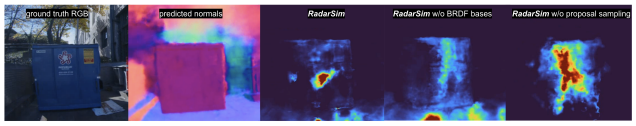

Figure 9. Ablation on our proposed BRDF bases encoding to model view dependence (**second to the right**) and sampling (**right**). We show that we model high-frequency normal-dependent reflectance changes: when view direction points toward the surface normal, strong reflectance is shown, and quickly decreases when view-direction deviates from the normal. While conditioning with spherical harmonic encoded view direction can model such effects, the view-dependence is much lower frequency, producing a blob in place of a dot. Without proposal sampling, render radar geometry is much cloudier compared to *RadarSim*.

renderer to learn high-resolution radar-specific 3D geometry and simulate more accurate radar range-Doppler images. We show the applicability of *RadarSim* across several diverse indoor and outdoor scenes and demonstrate that implicit geometry sharing can be an incredibly powerful tool for sensor fusion in 3D reconstruction for sensors with greatly varying characteristics. While *RadarSim* can leverage cameras to improve radar simulation, one limitation is that the reliance on cameras for high resolution angular information may degrade performance under conditions where camera data is compromised, such as in low light or environments with shiny surfaces. Additionally, despite enhancing radar reconstructions, range-Doppler accuracy remains limited by radar's inherent spatial resolution. Motivated by the strong performance of *RadarSim*, we believe other sensors could be integrated into a neural-implicit multi-sensor field. We en-

vision a future system that learns shared geometry from any sensor subset and dynamically adapts to available modalities, fully leveraging sensor synergies for scalable multimodal scene understanding across diverse environments and tasks.

Acknowledgements

Chuhan Chen was supported by a NSERC Postgraduate Scholarship (PGS-D).

References

[1] Benjamin Attal, Eliot Laidlaw, Aaron Gokaslan, Changil Kim, Christian Richardt, James Tompkin, and Matthew O'Toole. Törf: Time-of-flight radiance fields for dynamic scene view synthesis. *Advances in Neural Information Processing Systems*, 34, 2021. 5

[2] Stefan Auer, Richard Bamler, and Peter Reinartz. Raysar - 3d sar simulator: Now open source. 2016. 2

[3] Francesco Ballerini, Pierluigi Zama Ramirez, Roberto Mirabella, Samuele Salti, and Luigi Di Stefano. Connecting nerfs images and text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 866–876, 2024. 3

[4] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. *CVPR*, 2022. 5, 14

[5] Oded Bialer and Yuval Haitman. Radsimreal: Bridging the gap between synthetic and real data in radar object detection with simulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15407–15416, 2024. 2

[6] 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. 2, 3, 6, 8, 12, 15, 16

[7] Xingyu Chen and Xinyu Zhang. Rf genesis: Zero-shot generalization of mmwave sensing through simulation-based data synthesis and generative diffusion models. In *ACM Conference on Embedded Networked Sensor Systems (SenSys '23)*, pages 1–14, Istanbul, Turkiye, 2023. ACM, New York, NY, USA. 3

[8] C. J. Coleman. A ray tracing formulation and its application to some problems in over-the-horizon radar. *Radio Science*, 33(4):1187–1197, 1998. 2

[9] Michele Crosetto and F Pérez Aragues. Radargrammetry and sar interferometry for dem generation: validation and data fusion. In *SAR workshop: CEOS committee on earth observation satellites*, page 367, 2000. 2

[10] Qingyun di and Miaoyue Wang. Migration of ground-penetrating radar data method with a finite-element and dispersion. *Geophysics*, 69, 2004. 2

[11] Christopher Doer and Gert F. Trommer. Yaw aided radar inertial odometry uisng manhattan world assumptions. In *2021 28th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS)*, pages 1–10, 2021. 2

[12] Thibaud Ehret, Roger Marí, Dawa Derksen, Nicolas Gasnier, and Gabriele Facciolo. Radar fields: An extension of radiance fields to sar. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 564–574, 2024. 3, 4

[13] Fong et al. Panoptic nuscenes: A large-scale benchmark for lidar panoptic segmentation and tracking. *arXiv*, 2021. 12

[14] Rebut et al. Raw high-definition radar for multi-task learning. In *CVPR*, pages 17021–17030, 2022. 12

[15] Eduardo C. Fidelis, Fabio Reway, Herick Y. S. Ribeiro, Pietro L. Campos, Werner Huber, Christian Icking, Lester A. Faria, and Torsten Schön. Generation of realistic synthetic raw radar data for automated driving applications using generative adversarial networks, 2023. 3

[16] David A Forsyth and Jean Ponce. A modern approach. *Computer vision: a modern approach*, 17:21–48, 2003. 3

[17] Xiao Fu, Wei Yin, Mu Hu, Kaixuan Wang, Yuexin Ma, Ping Tan, Shaojie Shen, Dahua Lin, and Xiaoxiao Long. Geowizard: Unleashing the diffusion priors for 3d geometry estimation from a single image. In *ECCV*, 2024. 6, 15

[18] C.M. Furse, S.P. Mathur, and O.P. Gandhi. Improvements to the finite-difference time-domain method for calculating the radar cross section of a perfectly conducting target. *IEEE Transactions on Microwave Theory and Techniques*, 38(7): 919–927, 1990. 2

[19] Mariam Hassan, Florent Forest, Olga Fink, and Malcolm Mielle. Thermonerf: Multimodal neural radiance fields for thermal novel view synthesis. *arXiv preprint arXiv:2403.12154*, 2024. 3

[20] Quentin Herau, Nathan Piasco, Moussab Bennehar, Luis Roldao, Dzmitry Tsishkou, Cyrille Migniot, Pascal Vasseur, and Cédric Demonceaux. Soac: Spatio-temporal overlap-aware multi-sensor calibration using neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15131–15140, 2024. 3

[21] Nils Hirsenkorn, Paul Subkowski, Timo Hanke, Alexander Schaermann, Andreas Rauch, Ralph Rasshofer, and Erwin Biebl. A ray launching approach for modeling an fmcw radar system. In *2017 18th International Radar Symposium (IRS)*, pages 1–10, 2017. 2

[22] Shengyu Huang, Zan Gojcic, Zian Wang, Francis Williams, Yoni Kasten, Sanja Fidler, Konrad Schindler, and Or Litany. Neural lidar fields for novel view synthesis. *arXiv preprint arXiv:2305.01643*, 2023. 2, 3

[23] Tianshu Huang, John Miller, Akarsh Prabhakara, Tao Jin, Tarana Laroia, Zico Kolter, and Anthony Rowe. Dart: Implicit doppler tomography for radar novel view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24118–24129, 2024. 2, 3, 4, 5, 6, 7, 8, 12, 14, 15, 16

[24] Jonas Jansson. *Collision Avoidance Theory: With application to automotive collision mitigation*. PhD thesis, Linköping University Electronic Press, 2005. 1

[25] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. Lerf: Language embedded radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19729–19739, 2023. 3

[26] Andrew Kramer, Kyle Harlow, Christopher Williams, and Christoffer Heckman. Coloradar: The direct 3d millimeter wave radar dataset. *The International Journal of Robotics Research*, 41(4):351–360, 2022. 2, 12, 13

[27] Jaime Laviada, Ana Arboleya-Arboleya, Yuri Álvarez, Borja González-Valdés, and Fernando Las-Heras. Multiview three-dimensional reconstruction by millimetre-wave portable camera. *Scientific reports*, 7(1):6479, 2017. 2, 3

[28] Jaime Laviada, Ana Arboleya-Arboleya, and Fernando Las-Heras. Multistatic millimeter-wave imaging by multiview portable camera. *IEEE Access*, 5:19259–19268, 2017.

[29] Jaime Laviada, Miguel Lopez-Portugues, Ana Arboleya-Arboleya, and Fernando Las-Heras. Multiview mm-wave imaging with augmented depth camera information. *IEEE Access*, 6:16869–16877, 2018. 2, 3

[30] Xinrong Li, Xiaodong Wang, Qing Yang, and Song Fu. Signal processing for tdm mimo fmcw millimeter-wave radar sensors. *IEEE Access*, 9:167959–167971, 2021. 2

[31] Teck-Yian Lim, Spencer A. Markowitz, and Minh N. Do. Radical: A synchronized fmcw radar, depth, imu and rgb camera data dataset with low-level fmcw radar signals. *IEEE Journal of Selected Topics in Signal Processing*, 15(4):941–953, 2021. 12

[32] Teck-Yian Lim, Spencer A Markowitz, and Minh N Do. Radical: A synchronized fmcw radar, depth, imu and rgb camera data dataset with low-level fmcw radar signals. *IEEE Journal of Selected Topics in Signal Processing*, 15(4):941–953, 2021. 2, 13

[33] Afei Liu, Shuanghui Zhang, Chi Zhang, Shuaifeng Zhi, and Xiang Li. Ranerf: Neural 3d reconstruction of space targets from isar image sequences. *IEEE Transactions on Geoscience and Remote Sensing*, 2023. 3

[34] M. Malinen and P. Råback. *Elmer finite element solver for multiphysics and multiscale problems*. Forschungszentrum Juelich, 2013. 2

[35] Babak Mamandipoor, Greg Malysa, Amin Arbabian, Upamanyu Madhow, and Karam Noujeim. 60 ghz synthetic aperture radar for short-range imaging: Theory and experiments. In *2014 48th Asilomar Conference on Signals, Systems and Computers*, pages 553–558. IEEE, 2014. 3

[36] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2, 3, 4

[37] G. Minkler and J. Minkler. CFAR: The principles of automatic radar detection in clutter. *NASA STI/Recon Technical Report A*, 90:23371, 1990. 2

[38] Mohammadreza Mostajabi, Ching Ming Wang, Darsh Ranjan, and Gilbert Hsyu. High-resolution radar dataset for semi-supervised learning of dynamic objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2020. 3

[39] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 5

[40] Mert Ozer, Maximilian Weiherer, Martin Hundhausen, and Bernhard Egger. Exploring multi-modal neural scene representations with applications on thermal imaging. *arXiv preprint arXiv:2403.11865*, 2024. 3

[41] Dong-Hee Paek, Seung-Hyun Kong, and Kevin Tirta Wijaya. K-radar: 4d radar object detection for autonomous driving in various weather conditions. *Advances in Neural Information Processing Systems*, 35:3819–3829, 2022. 2, 13

[42] Akarsh Prabhakara, Vaibhav Singh, Swarun Kumar, and Anthony Rowe. Osprey: a mmwave approach to tire wear sensing. In *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, pages 28–41, 2020. 3

[43] Mohamad Qadri, Michael Kaess, and Ioannis Gkioulekas. Neural implicit surface reconstruction using imaging sonar. *arXiv preprint arXiv:2209.08221*, 2022. 3

[44] Kun Qian, Zhaoyuan He, and Xinyu Zhang. 3d point cloud generation with millimeter-wave radar. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 4(4), 2020. 3

[45] Ralph H Rasshofer and Klaus Gresser. Automotive radar and lidar systems for next generation driver assistance functions. *Advances in Radio Science*, 3:205–209, 2005. 1

[46] Julien Rebut, Arthur Ouaknine, Waqas Malik, and Patrick Pérez. Raw high-definition radar for multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17021–17030, 2022. 2, 13

[47] Albert W Reed, Juhyeon Kim, Thomas Blanford, Adithya Pediredla, Daniel C Brown, and Suren Jayasuriya. Neural volumetric reconstruction for coherent synthetic aperture sonar. *arXiv preprint arXiv:2306.09909*, 2023. 3

[48] Hermann Rohling. Radar cfar thresholding in clutter and multiple target situations. *IEEE Transactions on Aerospace and Electronic Systems*, AES-19(4):608–621, 1983. 2

[49] Christian Schöffmann, Barnaba Ubezio, Christoph Böhm, Stephan Mühlbacher-Karrer, and Hubert Zangl. Virtual radar: Real-time millimeter-wave radar sensor simulation for perception-driven robotics. *IEEE Robotics and Automation Letters*, 6(3):4704–4711, 2021. 2

[50] Christian Schüßler, Marcel Hoffmann, Johanna Bräunig, Ingrid Ullmann, Randolf Ebelt, and Martin Vossiek. A realistic radar ray tracing simulator for large mimo-arrays in automotive environments. *IEEE Journal of Microwaves*, 1(4): 962–974, 2021. 2

[51] Susan C Steele-Dunne, Heather McNairn, Alejandro Monsivais-Huertero, Jasmeet Judge, Pang-Wei Liu, and Kostas Papathanassiou. Radar remote sensing of agricultural canopies: A review. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(5): 2249–2273, 2017. 1

[52] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Justin Kerr, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, David McAllister, and Angjoo Kanazawa. Nerfstudio: A modular framework for neural radiance field development. In *ACM SIGGRAPH 2023 Conference Proceedings*, 2023. 5, 15

[53] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Justin Kerr, Terrance Wang, Alexander Kristoffersen, Jake

Austin, Kamyar Salahi, et al. Nerfstudio: A modular framework for neural radiance field development. *arXiv preprint arXiv:2302.04264*, 2023. 7

[54] Tang Tao, Guangrun Wang, Yixing Lao, Peng Chen, Jie Liu, Liang Lin, Kaicheng Yu, and Xiaodan Liang. Alignmif: Geometry-aligned multimodal implicit field for lidar-camera joint synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21230–21240, 2024. 3

[55] F.L. Teixeira, Weng Cho Chew, M. Straka, M.L. Oristaglio, and T. Wang. Finite-difference time-domain simulation of ground penetrating radar on dispersive, inhomogeneous, and conductive soils. *IEEE Transactions on Geoscience and Remote Sensing*, 36(6):1928–1937, 1998. 2

[56] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T Barron, and Pratul P Srinivasan. Ref-nerf: Structured view-dependent appearance for neural radiance fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5481–5490. IEEE, 2022. 4, 15

[57] Christian Waldschmidt, Juergen Hasch, and Wolfgang Menzel. Automotive radar—from first efforts to future systems. *IEEE Journal of Microwaves*, 1(1):135–148, 2021. 1

[58] Qian Wan, Yiran Li, Changzhi Li, and Ranadip Pal. Gesture recognition for smart home applications using portable radar sensors. In *2014 36th annual international conference of the IEEE engineering in medicine and biology society*, pages 6414–6417. IEEE, 2014. 1

[59] Hiroyoshi Yamada, Takumi Kobayashi, Yoshio Yamaguchi, and Yuuichi Sugiyama. High-resolution 2d sar imaging by the millimeter-wave automobile radar. In *2017 IEEE Conference on Antenna Measurements & Applications (CAMA)*, pages 149–150. IEEE, 2017. 3

[60] Muhammet Emin Yanik and Murat Torlak. Near-field mimo-sar millimeter-wave imaging with sparsely sampled aperture data. *Ieee Access*, 7:31801–31819, 2019. 3

[61] Ao Zhang, Farzan Erlik Nowruzi, and Robert Laganiere. Raddet: Range-azimuth-doppler based radar object detection for dynamic road users. In *2021 18th Conference on Robots and Vision (CRV)*, pages 95–102. IEEE, 2021. 2, 13

[62] Xiaopeng Zhao, Zhenlin An, Qingrui Pan, and Lei Yang. Nerf2: Neural radio-frequency radiance fields. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, pages 1–15, 2023. 3

[63] Haidong Zhu, Yuyin Sun, Chi Liu, Lu Xia, Jiajia Luo, Nan Qiao, Ram Nevatia, and Cheng-Hao Kuo. Multimodal neural radiance field. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9393–9399. IEEE, 2023. 3

Appendix

1. Supplementary Videos

Please refer to the attached supplementary videos for results. We show comparison between *RadarSim*- Radar reflectance rendered with radar occupancy (**left column**) and radar-only baselines DART [23](**middle column**) and Radarfields[6](**right column**). Observe that compared to the baseline, our reconstruction quality shown in depth map at the bottom is significantly better, while being able to accurately model radar reflectivity. Specifically, strong reflectance is visible for radar signals:

- at inset corners that act as retro-reflectors where all transmitted rays are reflected due to bouncing at corners, such as bottom of car, inside windows, wall/ceiling intersections, light fixture on ceilings etc.
- where view direction aligns with surface normal
- metallic material in general

We also show synthesized range-azimuth view in 128 azimuth directions, achieving super-resolution of the input radar data which only contains 8 directions from 8 antenna measurement. Notice our synthesized range-azimuth rendering is much sharper than DART[23], while preserving radar reflectivity.

2. Dataset

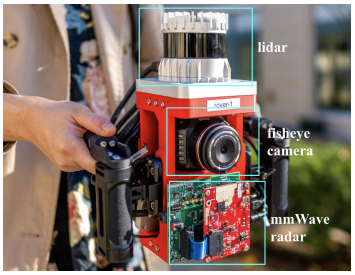

Figure 10. Image of our hand-held data collection rig with three key components labeled.

2.1. Data capture rig and pose processing

We build a hand-held rig for data collection with time-synced mmWave radar, fisheye camera, and lidar collecting data at 30 fps, 30 fps and 10 fps respectively.

For each sequence, we run COLMAP to get camera poses $\boldsymbol{A_c}$ (can be broken down into rotation $\boldsymbol{R_c}$ and position $\boldsymbol{x_c}$). Since coordinate systems of camera and radar are calibrated, we can easily convert camera pose to radar pose $\boldsymbol{A_r}$ (or $\boldsymbol{R_r}$ and $\boldsymbol{x_r}$) by interpolating into the sequence of camera poses with synced-timestamps followed with a transformation. We use the proposed scale estimation approach mentioned in Section 3.5 of the main paper to obtain scale of the scene for estimating ego-velocity required for our system.

2.2. Multimodal pose refinement

While COLMAP provides fairly accurate poses , radar ray tracing requires accurate **velocity**, so we design a pose and velocity refinement module by learning a per-frame pose and velocity offset $\Delta\boldsymbol{x_c}, \Delta\boldsymbol{R_c}, \Delta\boldsymbol{v_c}$ and regularized through:

1. L2 regularization on the offsets: $L_{regp} = ||\Delta\boldsymbol{x_c}||_2^2 + ||\Delta\boldsymbol{R_c}||_2^2 + ||\Delta\boldsymbol{v_c}||_2^2$
2. L2 loss that enforces velocity to be close to derivative of position $L_{regv} = ||d(\boldsymbol{x_c})/dt - (\boldsymbol{v_c} + \Delta\boldsymbol{v_c})||_2^2$
3. L2 regularization on acceleration $L_{rega} = ||d(\Delta\boldsymbol{v_c} + \boldsymbol{v_c})/dt||_2^2$
4. kinematic loss: $L_{regk} = ||d_{window}(\boldsymbol{x_c} + \Delta\boldsymbol{x_c}) - d_{window}(\int \Delta\boldsymbol{v_c} + \boldsymbol{v_c}dt)||_2^2$

Note that it is possible to derive optimized velocity from optimized positions ($\boldsymbol{x} + \Delta\boldsymbol{x}$), but we empirically found it to be more stable to keep a different set of velocity offset and position offset and have them loosely connected through regularization. Such pose optimization scheme is shared across camera and radar as radar pose and velocity can be interpolated from camera poses and velocities. We show in Table 5 that our proposed pose refinement method improves on original DART[23] without pose refinement.

2.3. Evaluation traces break down

We show lidar map of the scene, trajectory and a RGB view of our 8 evaluation traces in Figure 11. Number of images and radar frames range from 2000 to 3000. Image size used for training is 960x540 px.

2.4. Existing multimodal camera-radar dataset

We include a summary of existing multimodal datasets that contain raw radar measurement in Table 3 in the main paper and also attached here that we further elaborate on. There is a lack of multimodal radar-camera dataset that contains raw single-chip radar measurement for scene reconstruction, which requires overlapping, and view-direction varying scene content across multiple measurements. Automotive datasets, such as RADIal [14] and RaDICaL [31], offer limited variation in viewpoints and sensor height, making it difficult to use for evaluating novel view synthesis. RADIal [14] also uses Cascaded Radar with a complex modulation function, which makes modelling radar rendering equation difficult for inverse rendering. Coloradar [26] does capture indoor senes that can be used for reconstruction, but uses high-resolution radar which is also beyond our sin-radar focus. Most public RGB-Radar datasets, such as NuScenes [13], provide only *post-processed* CFAR data—not the *raw* measurements that we use. Hence we collect our own multimodal dataset and evaluate performance of our approach and baselines.

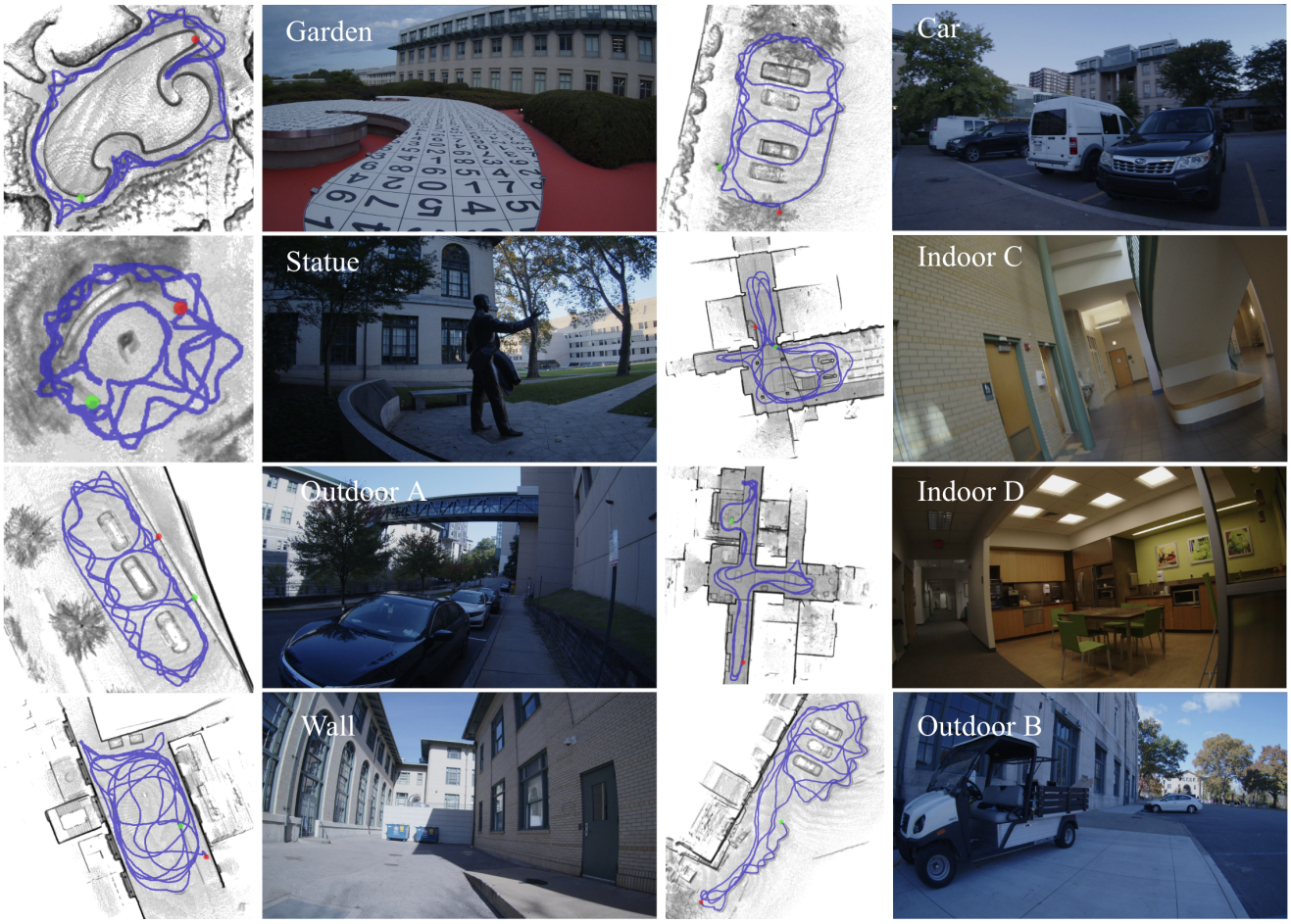

Figure 11. Visualization of the 8 indoor and outdoor scenes we collect and evaluate on in our experiments, in lidar map **(left)** and a sample RGB image **(right)**.

Dataset	Radar Type	Raw Data	View-direction Varying	Other Sensors
RadarSim (Ours)	**Low Res**	**Yes**	**Yes**	Lidar, Camera, IMU
RADDet [61]	**Low Res**	**Yes**	No	Camera
RADIal [46]	High Res	**Yes**	No	Lidar, Camera, GPS
K-radar [41]	High Res	**Yes**	No	Lidar, Camera, IMU, GPS
Coloradar [26]	High Res	**Yes**	**Yes**	Lidar, IMU
RaDICal [32]	**Low Res**	**Yes**	No	Depth Camera, IMU

Table 3. **Comparison with other mmWave radar datasets with raw data.** We capture a dataset using a low-resolution single-chip radar and cover scene content from multiple views directions and positions.

3. Method details

We elaborate here on technical details omitted in the main paper. See figure 12 for detailed architecture diagram.

3.1. Modelling radar view dependence with BRDF bases

View dependence of Radar reflectance can be broken into 2 scenarios, surface normal dependent reflectance and surface normal independent reflectance. The latter includes reflectance from retro-reflectance structures such as corners, bottom of car etc. (where rays always bounce back in a particular direction) and inter-reflections. To model these 2 scenarios, we input to Radar MLP with BRDF bases to model normal dependent view dependence, and spherical harmonics encoded view directions to model normal independent view dependence. Geometry code is also input to the Radar MLP as both types of reflections are spatially varying.

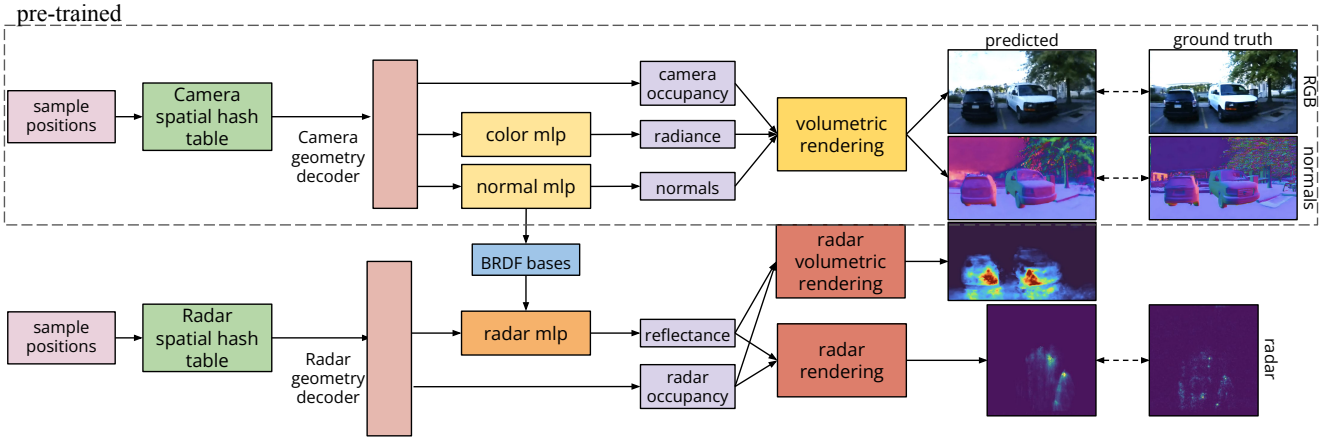

Figure 12. Architecture diagram of our framework.

3.2. Training details

Following DART [23], we process radar raw data into Range-Doppler-Azmith frames with dimension size 128, 128, 8. At each training iteration, a batch of radar Doppler columns Y_r are sampled to form radar rays. Radar rays are sampled on a cone with directions determined by velocity of the sensor at the current frame, and apex angle determined by the dot product between speed and Doppler value of the sampled column.

Radar rendering. Radar ray batch is sampled using a fine-tuned radar proposal sampler which outputs different sampling weights from camera. The hash table and density MLP corresponding to f_{geo_r} are initialized with pre-trained f_{geo_c} and optimized using radar measurements. They are queried to obtain radar occupancy and a geometry code, which is decoded and fed to the Radar MLP along with SH encoded view directions and our proposed BRDF bases to decode into radar reflectance. Reflectance and radar occupancy are assigned to each range bin as discussed in Sec 3.3 in the main paper, and rendered with DART rendering equation to synthesize the input Doppler column $\hat{Y_r}$ [23] .

3.3. Loss Functions

RGB model is supervised with losses used in Nerfacto. Our model is supervised with L_1 reconstruction loss and SSIM loss for radar measurement, binary cross entropy loss between radar and camera occupancy to constrain radar geometry to be close to camera geometry, as well as auxiliary losses for pose regularization (mentioned in Section 2.2), proposal sampling network and normals. We list the loss functions here:

Reconstruction loss

$$L_r = ||Y_r - \hat{Y_r}||_1 \tag{9}$$

SSIM loss

$$L_{ssim} = 1 - SSIM(Y_r, \hat{Y_r}) \tag{10}$$

Geometry consistent loss

$$L_{bce} = -\left[\alpha_c \log(\alpha_r) + (1 - \alpha_c) \log(1 - \alpha_r)\right] \tag{11}$$

Interlevel Losses We fine-tune the proposal sampler for radar with a proposal loss $L_{prop}(\boldsymbol{t}, \boldsymbol{w}, \hat{\boldsymbol{t}}, \hat{\boldsymbol{w}})$ discussed in [4] to encourage histgram of rendering weights $\hat{\boldsymbol{w}}$ queried from the proposal network at samples $\hat{\boldsymbol{t}}$ to match the rendering weight $\boldsymbol{w}$ of the geometry field at a set of different sample positions $\boldsymbol{t}$. Specifically,

$$L_{prop}(\boldsymbol{t}, \boldsymbol{w}, \hat{\boldsymbol{t}}, \hat{\boldsymbol{w}}) = \Sigma_i \frac{1}{\boldsymbol{w_i}} max(0, \boldsymbol{w_i} - bound(\hat{\boldsymbol{t}}, \hat{\boldsymbol{w}}, T_i)) \tag{12}$$

where $bound(\hat{\boldsymbol{t}}, \hat{\boldsymbol{w}}, T_i)$ is the sum of proposal weights $\hat{\boldsymbol{w}}$ in interval i. This loss function penalizes the proposal weights that under-estimates the rendering weight distribution from geometry field. Please refer to [4] for details about this loss function. We apply this loss to radar as L_{prop_r}, to enforce the proposal network to generate sampling weights that focus on radar geometry. Radar rendering weights are computed by

$$\boldsymbol{w_r} = \alpha_r(\boldsymbol{t}_i) \prod_{j<i} (1 - \alpha_r(\boldsymbol{t}_j)) \tag{13}$$

Normal Supervision Losses We supervise our normal prediction MLP with psudo ground truth normal $\boldsymbol{n_{gt}}$ from

a monocular normal estimator [17] by first converting it to world coordinate using camera extrinsics $\boldsymbol{A_c}$:

$$L_{norm} = ||\boldsymbol{n} - \boldsymbol{A_c n_{gt}}||_2^2 \quad (14)$$

Our combined loss is

$$L = \lambda_r L_r + \lambda_{bce} L_{bce} + \lambda_{ssim} L_{ssim} + \lambda_{prop_r} L_{prop_r} + \lambda_{norm} L_{norm} + \lambda_{norm_g} L_{norm_g} + \lambda_{norm_o} L_{norm_o} + \lambda_{regp} L_{regp} + \lambda_{regv} L_{regv} + \lambda_{rega} L_{rega} + \lambda_{regk} L_{regk} \quad (15)$$

In this equation, $L_{norm_g} + L_{norm_o}$ are adapted from [56] to guide the predicted normal with gradient direction of the camera density field, and encourage normals to point outward from a surface. We use $\lambda_r = 1e^{-3}$ for outdoor scenes and $\lambda_r = 1e^{-4}$ for indoor scenes where abundance of multi-path reflections result in overall high reflectance in the scenes. We choose $\lambda_{bce} = 0.01$ and $\lambda_{ssim} = 0.01$. We choose $\lambda_{norm} = 0.1$ to obtain strong supervision from ground truth normal. We follow default values in Nerfstudio and use $\lambda_{prop_r} = 1$, $\lambda_{norm_g} = 1e^{-3}$, $\lambda_{norm_o} = 1e^{-4}$. For pose refinement, we use $\lambda_{regp} = 1e^{-3}$, $\lambda_{regv} = 1$, $\lambda_{rega} = 5e^{-3}$, $\lambda_{regk} = 1$.

3.4. Implementation Details

We implement our pipeline using Pytorch in Nerfstudio [52] based on Nerfacto-big [52] . For training we used a single 24 GB Nvidia Rtx3090 GPU. Each sequence is trained on RGB images first for 50k iterations and fine-tuned on radar data for 30k iterations. Training time is around 2 hours. Learning rate for the radar model is $1e^{-2}$ annealed to $1e^{-4}$ after 30k steps, and for pose refinement is $1e^{-3}$ annealed to $1e^{-4}$ after 5k steps. We list model hyperparameters in Table 4.

4. Experiment details and additional result

4.1. Evaluation metric and denoising procedure

We calculate PSNR and SSIM values between *RadarSim* and ground truth for evaluation and comparison against baselines. Because most of the radar frame consists of noise, we design a denoising procedure by finding the noise threshold for each dataset. The noise threshold is calculated by fitting a chi-square distribution to the empty Doppler columns of each dataset (where speed is smaller than the Doppler values) and take a p-value of 0.01 of the noise distribution as the noise threshold. During evaluation, ground truth and synthesized range-Doppler frames are clipped at 0.01 and 99.99 percentile of the ground truth over the entire dataset and normalized to 0 and 1 to calculate SSIM and PSNR. Areas where the ground truth frames are below this threshold are considered to be noise and ignored during both PSNR and SSIM calculation.

4.2. Description of baselines

We run a pytorch version of DART [23] using code provided by the authors. The parameter for the hashtable and geometry decoder is set to match the size of *RadarSim*. We use the same set of poses for Radar for the baseline and our approach, where they are time-interpolated from COLMAP-derived camera poses. All comparisons included in the main paper with DART use additional pose refinement described in Section 2 for fair comparison with *RadarSim*. We include comparison with DART without pose refinement in Table 5. It can be observed that our proposed pose refinement scheme improves DART's novel view synthesis quality as the model is less prone to overfitting to the inaccuracies in poses. We also implement RadarFields [6] as an additional radar-only baseline. Since this approach is designed for high resolution radar, it only uses range-azimuth. To compare with our method, we train it on range-azimuth slices of the input data only and evaluate on range-doppler-azimuth synthesis quality. We also include result in Table 5. Although this method includes pose refinement, its performance is inferior to that of DART with pose refinement, suggesting the importance of leveraging Doppler for low resolution radar.

4.3. Additional application: geometry improvement for textureless region

We further illustrate the effectiveness of our BRDF encoding for modelling radar reflectance in a multimodal training framework where a single geometry field is optimized to represent radar and camera and trained simultaneous using radar and camera reconstruction losses. Without using monocular normal or depth priors, RGB-only reconstruction fails at identifying the true geometry for textureless regions. In a multimodal training setting, since our BRDF encoding effectively models radar view-dependence using predicted normals from the shared geometry, information from radar measurement can be propagated to the geometry and reconstruct textureless regions correctly as shown in figure 13. Comparison with other RGB-based or multimodal-based methods in improving geometry quality is left to future work.

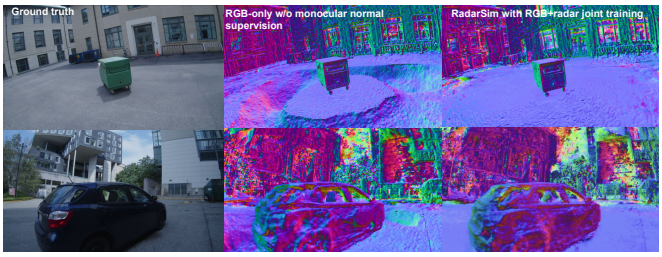

Figure 13. Normal-dependent BRDF encoding allows *RadarSim* to improve reconstruction of textureless regions in a joint training setting, without using monocular-predicted normals.

Model	Configuration	Value
SH	degree	25
BRDF bases	number	11
Proposal Hash Encoding level 0,1	# of levels	5, 5
	Hash table size	2^17, 2^17
	# of feature dim. per entry	2,2
	Coarse resolution	16,16
	Fine resolution	128, 256
	Decoder feature dim	16,16
	Number of layer	2,2
	# of ray samples	512, 256
Hash Encoding	# of levels	16
	Hash table size	2^21
	# of feature dim. per entry	2
	Coarse resolution	16
	Fine resolution	2048
	# of ray samples	64
Density MLP	# of hidden layer	2
	# of neuron per layer	128
	Output activation	Exp
	Density feature dim	15
Radar MLP	# of hidden layer	2
	# of neuron per layer	128
	Output activation	None
Color MLP	# of hidden layer	2
	# of neuron per layer	128
	Output activation	Sigmoid
Normal MLP	# of hidden layer	2
	# of neuron per layer	64
	Output activation	None

Table 4. List of hyperparameters used in our architecture.

	Car		**Garden**		**Statue**		**Wall**		**Outdoor A**		**Outdoor B**		**Indoor C**		**Indoor D**		**Aggregated**	
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
CFAR	0.694	24.9	0.415	24.1	0.711	17.9	0.772	27.1	0.502	20.2	0.734	25.3	0.774	28.0	0.770	28.2	0.671	24.5
Lidar	0.782	29.4	0.470	26.7	0.830	28.9	0.796	27.9	0.595	26.8	0.782	28.6	0.805	29.1	0.809	29.2	0.734	28.3
Nearest Neighbor	0.733	26.5	0.592	24.6	0.798	26.2	0.810	24.6	0.580	22.9	0.726	24.3	0.782	26.6	0.785	27.3	0.726	25.4
RadarFields [6]	0.831	29.3	0.603	27.0	0.869	29.0	0.781	27.1	0.667	24.4	0.773	27.5	0.814	28.6	0.831	29.3	0.771	27.8
DART [23] without pose refinement	0.832	29.8	0.616	27.1	0.866	29.1	0.809	27.0	0.691	26.3	0.797	27.1	0.817	28.5	0.846	29.3	0.784	28.0
DART [23] with pose refinement	0.850	30.5	0.658	**27.7**	0.887	30.1	0.818	27.1	0.702	26.6	0.801	27.5	0.827	28.9	0.853	29.5	0.799	28.5
RadarSim (ours)	**0.859**	**30.8**	**0.669**	27.5	**0.889**	**30.2**	**0.851**	**28.3**	**0.741**	**27.6**	**0.853**	**29.1**	**0.845**	**29.5**	**0.863**	**29.7**	**0.821**	**29.1**

Table 5. **Per-scene break down comparison with baselines RadarFields [6] and DART[23] with and without pose refinement.** *RadarSim* achieves significantly higher PSNR and SSIM on outdoor scenes (*Car, Statue, Wall, Outdoor A Outdoor B*) compared to baselines, as well as higher performance on indoor scenes (*Indoor C, Indoor D*), though the gains are less pronounced. This is because the effectiveness of leveraging surface normals to reproduce input radar recordings is reduced in indoor environments due to the increased presence of multipath reflections from corners and clutter. For Outdoor scene *Garden*, DART slightly outperforms *RadarSim* as this scene is dominated by inter-reflections between the metallic edge of the garden and ground as well as from bushes which diffusely reflects radar signal, making our normal-dependent model less effective.