DART: Implicit Doppler Tomography for Radar Novel View Synthesis

Tianshu Huang[*1]   John Miller[*12]   Akarsh Prabhakara[1]   Tao Jin[1]   Tarana Laroia[1]

Zico Kolter[12]   Anthony Rowe[12]

[1]Carnegie Mellon University   [2]Bosch Research

{tianshu2, jmiller4, aprabhak, taojin, tlaroia, agr, zkolter}@andrew.cmu.edu

Abstract

Simulation is an invaluable tool for radio-frequency system designers that enables rapid prototyping of various algorithms for imaging, target detection, classification, and tracking. However, simulating realistic radar scans is a challenging task that requires an accurate model of the scene, radio frequency material properties, and a corresponding radar synthesis function. Rather than specifying these models explicitly, we propose DART — Doppler Aided Radar Tomography, a Neural Radiance Field-inspired method which uses radar-specific physics to create a reflectance and transmittance-based rendering pipeline for range-Doppler images. We then evaluate DART by constructing a custom data collection platform and collecting a novel radar dataset together with accurate position and instantaneous velocity measurements from lidar-based localization. In comparison to state-of-the-art baselines, DART synthesizes superior radar range-Doppler images from novel views across all datasets and additionally can be used to generate high quality tomographic images.[1]

1. Introduction

Driven by advances in the automotive industry, miniaturized millimeter wave (mmWave) radar chips are becoming cheaper and more ubiquitous. Boasting a high range resolution and the ability to penetrate light materials, mmWave radars have proven effective in many application domains including collision avoidance and driver assistance in automobiles [14, 36, 57, 64, 65], through-occlusion imaging in airport scanners [30, 68], and vision-denied tracking and mapping [2, 9, 22, 37, 43].

Because designing, testing, and deploying new radar systems in the real world can be costly, many rapid prototyping pipelines heavily rely on simulation. Modern radar simula-
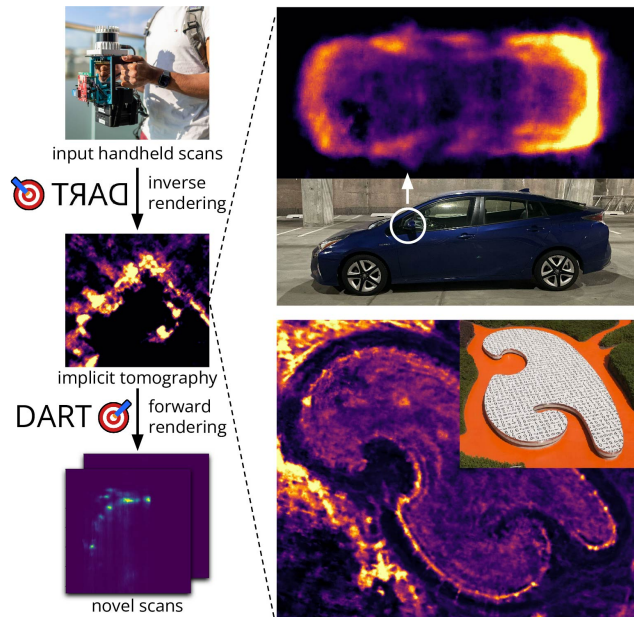

Figure 1. DART uses scans from a handheld radar to learn an implicit tomography of a scene in order to accurately render scans from novel viewpoints (left). DART's implicit tomography can also be sampled to map the radar properties of a scene (right).

tion tools normally require the user to manually specify the geometry and characteristics of the scene, including all material properties [3]. While other sensors (e.g. lidar) can be used to scan an environment and produce a mesh or voxel map, they cannot capture radar-specific material properties that are crucial for generating realistic radar scans. Thus, in practice, this results in greatly simplified environment models due to the difficulty of meticulously surveying a scene and generating (or annotating) a model by hand.

We envision a more intelligent, data-driven approach to scene modeling for radar simulation where a user can carry a handheld radar sensor through a static environment and automatically generate a model suitable for accurate simulation of that environment. To this end, we frame the radar simulation problem as one of *novel view synthesis*: using several radar measurements of a scene to simulate what a

---

*Equal Contribution.

[1]Our implementation, data collection platform, and collected datasets can be found via our project site: https://wiselabcmu.github.io/dart/.

radar would see from a new pose. Such a system would not only accelerate the development and testing of new algorithms across a variety of environmental conditions, but also open the door to a myriad of new inference techniques in radar sensing such as localization, mapping, imaging, and recognition which rely on accurate forward rendering models and could greatly benefit from realistic radar models.

Novel View Synthesis   Neural Radiance Fields (NeRFs) [48] have revolutionized novel view synthesis, leading to an explosion in interest in graphics and beyond. By leveraging a (neural) *implicit* scene representation instead of explicitly modeling scene geometry, textures, and materials, NeRFs are able to capture and reproduce visual intricacies such as specularity, translucency, reflections, and complex occlusions. This results in a 3D scene capture and rendering system that boasts an unprecedented level of photorealism.

Drawing inspiration from the success of NeRFs, we formulate an analogous problem for mmWave radar imaging. Our method, *Doppler-Aided Radar Tomography* (DART), takes a similar approach by implicitly capturing material properties from input scans which are reproduced when the model is sampled from a novel viewpoint. Though our model is implicit, we can also generate an explicit tomographic image by sampling along a voxel grid, which we use to show that DART is not simply memorizing the input data, but is in fact *learning* the geometry and material properties of the scene (Fig. 1).

Key Challenges   Applying NeRF's implicit scene modeling paradigm to the radar domain presents substantial challenges. We derive a rendering model from the ground up that appropriately reflects the unique nature of radar wave propagation. In NeRF, rendering each pixel involves integrating samples along a 1D ray, following a pinhole camera model [48]. However, radar waves propagate *radially* from the antenna. Even after range-azimuth-elevation processing, each radar pixel corresponds to a coarse 2D region of space, as the elevation-azimuth resolution of compact mmWave radars tends to be relatively poor[2]. One key insight is to choose a radar representation space — range-Doppler — which greatly reduces angular ambiguity in one dimension under the assumption that the scene is static and the radar is moving with a known velocity [26]. This presents additional systems challenges, as the sensor platform needs to be moving and its velocity must be measured accurately alongside its position and orientation.

Even with the dimensionality reduction afforded by range-Doppler processing, rendering a single radar pixel involves integrating samples along a *circle*, rather than a ray

---

[2]For context, these radars have angular resolutions on the order of $15°$, orders of magnitude worse than cameras ($\approx 0.01°$) [58, 76]
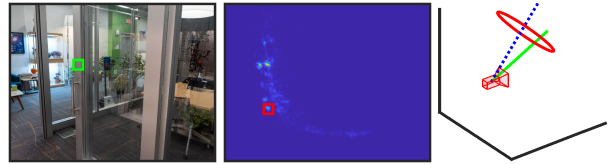

Figure 2. NeRF's pinhole camera model renders a pixel (left) by integrating along a ray (right, green), while DART's range-Doppler model renders a pixel (middle) by integrating along a velocity-dependent (right, blue) circle (right, red).

(Fig. 2). However, appropriately capturing occlusion effects requires that the nearest ranges are processed first due to occlusion caused by objects closer to the radar. Additionally, the size of the integration arc grows as the distance from the radar increases, resulting in an effective decrease in sampling density for points further from the radar that needs to be accounted for. Through careful modeling of these effects and a clever sampling scheme prioritizing sample re-use, we derive a computationally efficient forward rendering function that produces realistic novel radar scans.

Contributions   We propose DART: Doppler-Aided Radar Tomography, which implicitly learns a tomographic representation of the world in order to accurately synthesize radar range-Doppler images. To summarize our contributions:
1. We formulate the problem of radar novel view synthesis from implicit reflectance and transmittance maps using range-Doppler images.
2. Using a NeRF-inspired technique, we explicitly formulate the forward rendering of range-Doppler radar images and implicitly invert it using gradient descent to learn a neural-implicit representation.
3. We construct a data collection rig and collect novel radar imaging datasets with accurate position and instantaneous velocity along with reference lidar point clouds.
4. We evaluate DART across a range of scenarios and show that it out-performs the state-of-the-art, quantitatively and qualitatively, in both its synthetic radar renderings and its implicit imaging of scenes.

Limitations   Since we rely on Doppler, our method is limited to static scenes, and requires accurate velocity estimates and a constantly moving radar. While motion is intrinsic to our method, we believe that it is reasonable to require movement during scanning. Poor velocity estimates or non-static scenes can cause DART to perform poorly; we hope to relax these limitations in the future.

2. Related Work

2.1. Radar Simulation

Model-Based Approaches   *Model-based* methods use a physics and environment models to simulate the propaga-
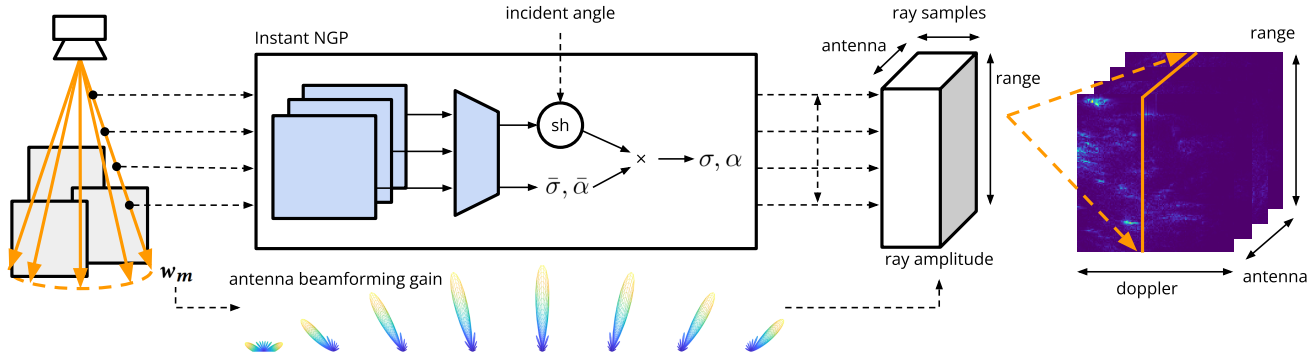
Figure 3. DART tackles radar novel view synthesis by learning a neural implicit map of the world from a trajectory of radar measurements. We make key radar-specific decisions in choosing (1) a high quality radar representation space — **Range-Doppler**, (2) a world model that captures radar interactions — σ **and** α **with spherical harmonics coefficients**, (3) a network architecture to model our desired representation — **Instant NGP**, and (4) an optimized radar rendering and training methodology — **Range-Doppler specific rendering**.

tion of radar signals through the environment using some combination of ray tracing [3, 11, 25, 66, 67], finite element modeling (FEM) [13, 45], or finite-difference time domain (FDTD) simulation [16, 19, 75]. While simulators can replicate complex scene dynamics (e.g. occlusion, path loss, multipath, non-Lambertian effects), they make no attempt to infer the environment structure from sensor data, and their accuracy is limited by the user's ability to create a radar-realistic model of the environment.

Data-Driven Approaches   *Data-driven* methods use real sensor scans to build an environment model. *Sparse* methods use constant false alarm rate detection (CFAR) to detect discrete reflectors in the environment [15, 49, 63]. On the other hand, *dense* methods divide the environment into an explicit voxel grid and infer the radar properties of each cell.

Dense methods can be further divided into coherent and incoherent aggregation. If a fixed (e.g. linear or circular) trajectory or sub-wavelength-accurate pose estimates are available, Synthetic Aperture Radar (SAR) can be used [46, 50, 52, 56, 81, 82]; however, this is impractical for a mobile platform over large areas. Instead, sensor readings (with high angular resolution via many antennas or SAR on smaller pieces of a trajectory) can also be aggregated in an *incoherent* manner, which has been referred to as multi-view 3D reconstruction [33–35] and radargrammetry [12].

2.2. Machine Learning Methods in Radar

Many classical radar problems such as radar super-resolution [10, 17, 20, 21, 23, 53, 54, 72], odometry [2, 43], mapping [42], activity recognition [39, 70, 77, 80], and object classification [32, 69, 85] have been applied to cheaper, lighter, and more compact radar systems using machine learning. We now seek to solve the novel view synthesis problem from compact, low resolution radars while implicitly creating a higher resolution map.

2.3. Neural Radiance Fields

Instead of defining an *explicit* inverse imaging algorithm that recovers a representation of the scene from sensor readings, Neural Radiance Fields [48] *implicitly* invert a forward rendering function through stochastic gradient descent. This requires the following components:

1. **World model**: NeRF defines the world as an RGB color and transparency for each position and viewing angle; subsequent works have generalized this to handle anti-aliasing [5], different cameras, and lighting [47, 73].
2. **World representation**: Beyond neural networks [48] or voxel grids [40], more recent works have explored spatial hash tables [51] as well as function decompositions for view angle dependence [18, 83].
3. **Rendering function and Model Inversion**: NeRFs model each pixel as a ray and ray-trace the radiance field. The invertibility of this rendering function is crucial: by assuming that each pixel is a ray, the NeRF is "supervised" by one RGB image pixel per ray, allowing NeRF to "solve" for the few opaque points along the ray.

We innovate on these key enablers of NeRFs in order to apply this approach to mmWave radars. By applying NeRF techniques to radar, we hope to leverage the extensive body of neural radiance field literature, while also unlocking the potential of neural-implicit representations.

Beyond Visual Fields   The success of NeRFs has inspired numerous other efforts to apply the same general principle to other sensors, including spatial audio [44], imaging sonar [55, 59], LIDAR simulations [27], and RSSI (Received Signal Strength Indicator) mapping [84]. NeRFs have also been applied to radar [29, 71] for camera-like high-resolution Synthetic Aperture Radar instead of the compact and inexpensive radars we explore in this paper.

3. DART: Doppler-Aided Radar Tomography

While our overall approach is inspired by Neural Radiance Fields, the physics of radar presents several new challenges. We make the following key design decisions (Fig. 3):

1. We first choose a radar measurement representation space — range-Doppler — that overcomes the poor spatial resolution of compact radars (Sec. 3.1, 3.2).

2. We then choose a model to account for radar-specific effects of electromagnetic wave interaction which are important for realistic view synthesis such as specularity, ghost reflections and partial occlusions (Sec. 3.3).

3. Finally, to effectively train and learn neural implicit maps for radars, we choose a network architecture for an *adaptive grid* world representation, design a range-Doppler *rendering* method, and propose key rendering optimizations (Sec. 3.3 — 3.4).

3.1. Range-Doppler Representation

Unlike cameras, radars are active sensors which illuminate a scene by transmitting a radio frequency waveform. Upon processing reflections received from objects in the scene, radars can perceive the world in 3 dimensions — range, azimuth, and elevation — as a heatmap indicating the reflectivity of objects at that 3D coordinate [60, 61].

However, while bulky mechanical radars or large solid-state radar arrays can provide azimuth and elevation resolution close to typical cameras, modern inexpensive and compact solid-state radars feature small antenna arrays which make them far inferior in the azimuth and elevation axes [28]. As a result, these compact radars can only generate coarse heatmaps ($>15°$ resolution) in the azimuth and elevation axes, causing each range-azimuth-elevation bin to point to a coarse region of 3D space which is far less sharp than a ray from a camera pixel [38, 41, 76].

To achieve better angular resolution, radars can instead leverage the Doppler effect: objects moving at different relative velocities to the radar have different Doppler velocities, which can be measured by examining the residual phase of the range-azimuth-elevation heatmap [79]. Crucially, in a static scene, these relative velocities depend on not just the relative speed between the radar and the world, but also the relative azimuth and elevation angle between objects and the radar, with each Doppler corresponding to a cone in space [60]. Because of the fine range and Doppler resolutions, Doppler greatly reduces the ambiguity of each bin in 3D space down to a thin ring (Fig. 4), which we further reduce by making a *thinness* argument across the range and Doppler axis in order to simplify integration down to a circle for radar rendering (Sec. 3.4).

3.2. Radar Pre-Processing

mmWave radars use a waveform called Frequency Modulated Continuous Wave (FMCW), and measure a continu-
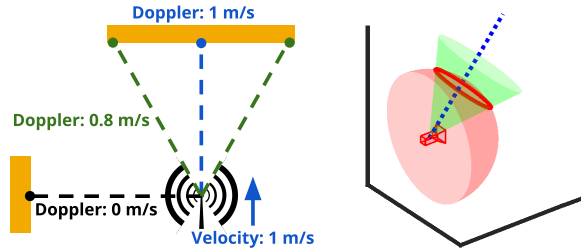

Figure 4. Doppler arises due to differences in relative velocities between points with different relative angles to the radar (left). Each range value (red) corresponds to a sphere, while each Doppler value corresponds to a cone (green). The intersection forms the range-Doppler pixel (see Fig. 2).

ous time signal; we then convert these signals into range-Doppler-antenna heatmaps. To summarize key points of our radar processing pipeline (Appendix A.1):

- **Undesirable Range-Doppler Side Lobes**: A single reflective object can create sidelobes that bleed into several range-Doppler bins and mask off weaker objects [61, 86]. Rather than forcing DART to model this, we use a Hann weighting window along both range and Doppler axis to mitigate this effect (Appendix A.1).

- **Multiple Antennas**: We perform range-Doppler processing on each of the eight transmit-receive (TX/RX) pairs in our radar. During our rendering process (Sec. 3.4), we apply the antenna gain and array factor for each TX/RX pair (Fig. 3), emphasizing 8 sections of the field of view. While our sense of high-quality azimuth-elevation information still stems from leveraging Doppler, this provides some coarse directional information.

3.3. DART's World Model

If we had an accurate model of the world and the electromagnetic wave interaction for all objects in the world, we could just apply this model to the region defined by each range-Doppler pixel to calculate its value. However, due to the complex nature of real-world scenes and interactions, both tasks are highly difficult and typically impractical. Instead, we model these properties in a data-driven way, representing the reflectance and transmittance using a view-dependent neural network-based approach.

Modeling RF Reflectivity Modeling mmWave material interactions is one of the most challenging factors of radar view synthesis. From the perspective of radar, points in space have two key properties: reflectance (the proportion of energy that reflects back), and transmittance (the proportion of energy that continues past) [60]. However, millimeter waves also interact with objects differently depending on incidence angles [4]; for example, metal surfaces can be specular and may be invisible from certain view points. As such, we model each physical point with a reflectance and

transmittance value, each of which depend on the incident angle. We formalize this as

$$\sigma : \mathbb{R}^6 \longrightarrow \mathbb{R}, \qquad \alpha : \mathbb{R}^6 \longrightarrow [0,1], \qquad (1)$$

which model the reflectance σ and transmittance α as a function of the position ($\mathbb{R}^3$) and incident angle ($\mathbb{R}^3$) of an incoming radar wave, and allows DART to model a wide range of radar phenomena such as partial occlusions, specularity, and ghost reflections (Appendix A.2).

World Representation While voxel-based approaches are highly effective for learning visual radiance fields [18, 83], radar images have a much poorer elevation and azimuth resolution compared to cameras even after exploiting the Doppler axis. This magnifies the difference in spatial resolution that σ and α can be resolved for between close and far ranges. Moreover, unlike cameras, our angular resolution is variable at all scales — be it at a trajectory level, frame-to-frame level or even within a frame (Sec. 3.1). Similar to NeRFs [48], we turn to neural implicit representations as a means of creating an "adaptive" grid, and base our model on the Instant Neural Graphics Primitive[3] [51].

Unlike most visual NeRFs, we do not provide the incident angle as an input to the neural network [74]. Instead, our architecture (visualized in the center block of Fig. 3) outputs a "base" reflectance $\bar{\sigma}$ and transmittance $\bar{\alpha}$, as well as shared spherical harmonics coefficients [83] which are applied to the incident angle as an inner product. In addition to computational advantages, this allows us to directly interpret $(\bar{\sigma}, \bar{\alpha})$ as spherical integrals of our learned reflectance and transmittance functions (Appendix A.3).

We also find that the output activation function on σ and α is critical for numerical stability and performance. Since σ is unbounded[4], we apply a linear activation to σ. Then, to constrain $\alpha \in [0,1]$, we apply the activation function

$$f(\alpha) = \exp(\max(0, \alpha)), \qquad (2)$$

which we pair with a custom gradient estimator to handle initialization instability (Appendix A.4).

3.4. Radar Rendering and Model Training

We train σ and α using a differentiable mapping which generates a multi-antenna range-Doppler heatmap from a given (σ, α) network; we refer to this as *radar rendering*. Unlike visual NeRFs, DART must account for a range of physical effects in addition to occlusion including path attenuation, antenna gain patterns, and the radar-specific Doppler axis.

---

[3][51] implicitly creates an adaptive world grid by using many spatial hash tables with geometrically increasing resolutions, and resolves the output with a small neural network; we use the same general architecture.

[4]σ can be negative; however, since the observed range-Doppler-antenna heatmaps cannot be negative, $\sigma < 0$ will always increase both train and validation loss, so allowing this does not cause overfitting.

Ray Tracing Consider a single "ray" emitted from a radar at position $\boldsymbol{x}$ and orientation (rotation matrix) $\boldsymbol{A}$ at an incidence angle $\boldsymbol{w}$. As the ray travels through space up to the maximum range of the processed (range, Doppler, antenna) image, each point $\boldsymbol{x} + r_i\boldsymbol{w}$ at range r_i receives a signal of amplitude u_i, which is attenuated by a factor of r_i due to free space path loss. Each point then reflects a signal of amplitude $u_i\sigma(\boldsymbol{t}_i)$ back towards the radar, and propagates an amplitude of $u_i\alpha(\boldsymbol{t}_i)$ onwards. As reflected signals return to the radar, the signal loses an additional attenuation factor of r_i, while also suffering from occlusion from $\forall j < i : \alpha(\boldsymbol{t}_j)$.

Sampling $r_1 \ldots r_{N_r}$ discretely along the range bins of the processed heatmap across antennas, the radar return amplitude $C(i, k, \boldsymbol{w})$ for ray $\boldsymbol{w}$ at range bin i and antenna k is

$$C(i, k, \boldsymbol{w}) = g_k(\boldsymbol{A}^{-1}\boldsymbol{w})\frac{\sigma(\boldsymbol{x} + r_i\boldsymbol{w})}{r_i^2} \prod_{i'=1}^{i-1} \alpha(\boldsymbol{t}_{i'})^2, \quad (3)$$

where $g_k(\boldsymbol{A}^{-1}\boldsymbol{w})$ is the antenna beamforming gain antenna k at angle $\boldsymbol{w}$ (specified relative to the radar orientation $\boldsymbol{A}$).

Doppler Integration For a given pose with radar position $\boldsymbol{x}$, velocity $\boldsymbol{v}$, and orientation $\boldsymbol{A}$, we evaluate the return $\boldsymbol{Y}(r_i, d_j, k) \in \mathbb{R}$ at each range-Doppler-antenna bin (r_i, d_j, k), synthesizing a view-specific, multi-antenna range-Doppler heatmap. Since the doppler velocity is measured as $d_j = \langle \boldsymbol{w}, \boldsymbol{v} \rangle$, we integrate the return C along the thin ring corresponding to each bin as:

$$\boldsymbol{Y}(r_i, d_j, k) \propto \frac{r_i}{||\boldsymbol{v}||_2} \int_{\langle \boldsymbol{w}, \boldsymbol{v} \rangle = d_j, \, ||\boldsymbol{w}||_2 = 1} C(i, k, \boldsymbol{w}) \, d\boldsymbol{w} \qquad (4)$$

Note that we need to correct for the varying width of the discrete bins as a function of range and radar speed by dividing by the speed $||\boldsymbol{v}||_2$ and multiply by r_i (Appendix A.5).

Approximating this integral as a sum over M random directions $\boldsymbol{w}_1 \ldots \boldsymbol{w}_M$ such that $\langle \boldsymbol{w}, \boldsymbol{v} \rangle = d_j$, we multiply by an additional factor of r_i to correct for the circumference of the range-Doppler intersection as r_i increases. This yields

$$\boldsymbol{Y}(r_i, d_j, k) \propto \frac{r_i^2}{M||\boldsymbol{v}||_2} \sum_{m=1}^{M} C(i, k, \boldsymbol{w_m}). \qquad (5)$$

Optimized Rendering As the (σ, α) field function must be evaluated for every sample, efficient sampling is critical to computational efficiency. Thus, a naive approach of treating each (range, Doppler, antenna) "pixel" as an independent sample as is standard practice in NeRFs would be computationally prohibitive, requiring the field to be sampled (range, Doppler, antenna, range integration, Doppler integration) times to render a single image. As such, we aggressively reuse samples of σ and α by rendering all bins with the same Doppler simultaneously (Appendix A.6).
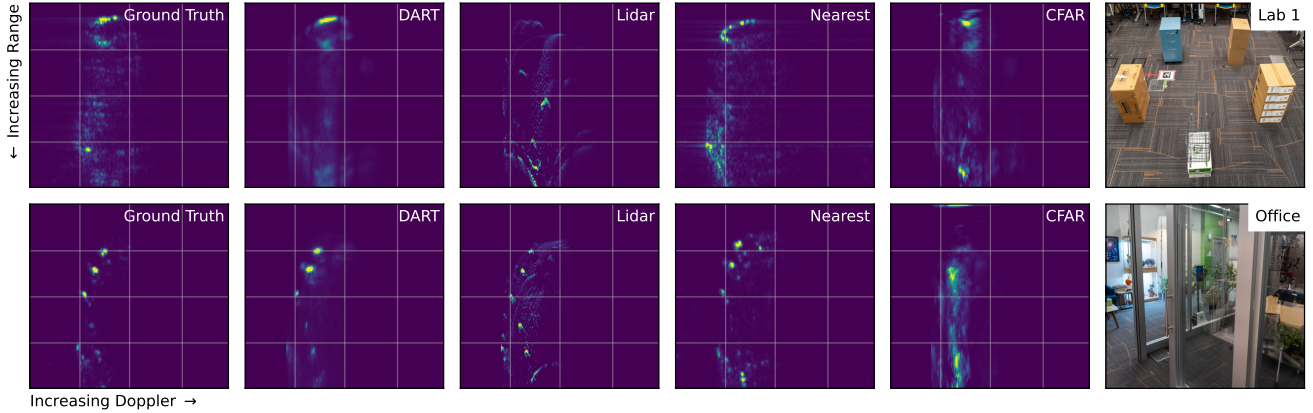
Figure 5. Example (validation) range-Doppler frames and descriptive photos of our method and baselines. DART accurately reproduces the overall radar image, though it lacks the resolution to resolve smaller weak reflectors. *Lidar* can model weak reflectors, but cannot accurately scale them due to a lack of radar-specific information, while *Nearest* produces radar-realistic but inaccurate images since exhaustively measuring all possible poses is impractical. Finally, *CFAR* cannot model transmittance or measure the "volume" of a point.

Training We train our (σ, α) field function using stochastic gradient descent with the Adam [31] optimizer and a $l1$ (i.e. mean-absolute-error) loss. For details about our training process and other hyperparameters, see Appendix A.7.

4. Experiments

We constructed a handheld data collection rig with a mmWave radar and a lidar used for localization[5] (Fig. 6; Appendix B.1). We used this to collect 12 traces ranging from 5 to 15 minutes long in a diverse set of environments including a lab space, townhouse, high-rise apartment, and an early 20th century house (Appendix B.2).

4.1. Baselines

We implement three baselines for radar novel view synthesis and mapping, a *model-based* approach and two *data-driven* approaches (see Sec. 2.1).

- **Lidar Scan-Based Simulator:** We use lidar scans to create an occupancy grid, which we then use in a raytracing radar simulator (assuming occupied grids have a fixed constant reflectance and no transmittance, similar to [3] without material annotations). This baseline represents the standard practice in radar simulation [3, 13, 66, 75].
- **Nearest Neighbor:** We implement a naive nearest-neighbor baseline which finds the training point with the closest (position, velocity) to the novel viewpoint. While simple, this has the advantage of using radar data to "simulate" images compared to our lidar-based simulator [7].
- **CFAR Point Cloud Aggregation:** CFAR is a commonly used adaptive algorithm in radar systems to detect target returns against a background of noise, clutter and interference [49]. We use the Matlab Phased Array System Tool-
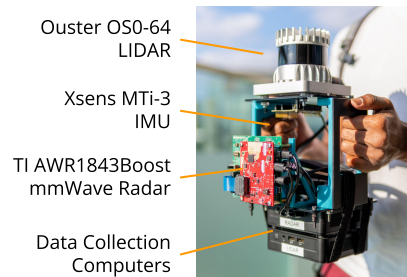

Figure 6. Handheld data collection system; see Appendix B.1.

box [1] to detect radar-reflective targets, de-project those targets into 3D points using Bartlett direction-of-arrival estimation [6], then reproject the points according to the novel pose. This approach is similar to our lidar baseline in that it uses point cloud aggregation, but is better able to capture radar-specific scene properties.

For additional details on our baselines, see Appendix B.3.

4.2. Metrics

We apply our model to a holdout test set consisting of the last 20% of each trace. We then compute the SSIM [78] of the test images and the effective sample size-corrected standard error (SE) for the mean SSIM; see Appendix B.4. We also compute the SSIM values of 25/30/35dB-equivalent Gaussian noise to help quantify our SSIM values.

5. Results

DART synthesizes significantly more accurate radar images than each baseline on all traces collected, while using minimal training. We also demonstrate DART's ability to sample tomographic images from its implicit map which are more dense than CFAR point clouds, and more faithfully reproduce radar characteristics than lidar scans.

---

[5]Note that while we use lidar to obtain pose estimates using Cartographer [24], any accurate 3D SLAM system can be used.
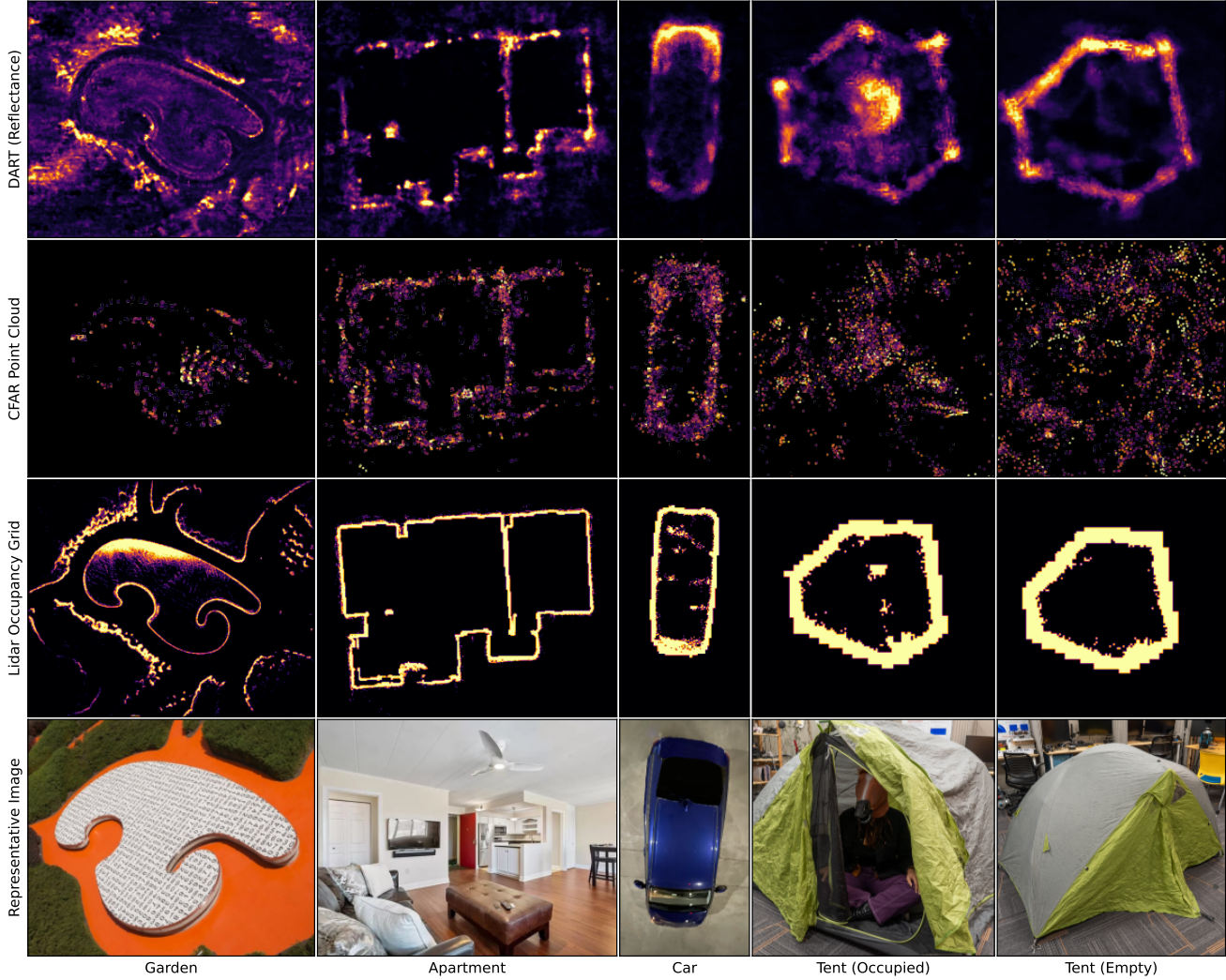
Figure 7. Comparison of DART (top) with CFAR (middle) and a Lidar occupancy grid for reference. While CFAR struggles with cluttered scenes and creates point clouds which are both noisy and sparse, DART creates relatively clear maps which capture radar-specific properties on both ourdoors (Garden) and indoors (Apartment) environments. DART can also image objects with relative clarity (Car), including resolving objects partially occluded by radar-transparent surfaces (Tent — Occupied/Empty).

Training Time We train DART for 3 epochs on each dataset using a RTX 4090 GPU, taking between 1-2× the data collection time (≈ 10 minutes) of each dataset[6]; this indicates the potential of real-time training with future algorithmic and computing hardware improvements.

Ablations Each part of DART's design significantly improves its accuracy, including view dependence using spherical harmonics and our dynamic grid representation (Tab. 1). For additional ablations, see Appendix C.1.

---
[6]Training time is not directly proportional to the dataset length: since Doppler bins are not observed when the radar speed is less than the Doppler velocity of that bin, we omit these bins, decreasing the training time. See Appendix B.2 for the length and training time of each dataset.

5.1. Comparison with Baselines

DART synthesizes far more accurate radar images than each baseline on all traces in our dataset (Appendix C.2), with the Lidar-based simulator and Nearest Neighbor baselines performing the worst, and CFAR-based simulation in between. DART is also significantly better than each baseline when evaluated as a whole (Tab. 1).

To understand the performance differences between DART and each baseline, we selected two example range-Doppler images from our dataset (Fig. 5):

- *Lidar-based simulation* (Lidar) can accurately identify reflector positions, but cannot correctly scale their radar return due to the lack of radar-specific material properties.
- *Nearest-Neighbor* (Nearest) approaches can, by defini-

Method	Mean SSIM	SSIM Improvement
DART	0.636 ± 0.012	—
Lidar	0.463 ± 0.005	0.174 ± 0.013
Nearest	0.468 ± 0.006	0.168 ± 0.012
CFAR	0.545 ± 0.007	0.091 ± 0.006
No View Dep.	0.614 ± 0.015	0.022 ± 0.005
20cm Grid	0.591 ± 0.015	0.046 ± 0.004

Table 1. Mean SSIM and SSIM improvement of DART over each baseline (and select ablations) across our dataset along with 95% confidence intervals; see Appendix C.2 for a breakdown by dataset.

tion, generate radar-realistic images. However, measuring all possible (position, orientation, velocity) poses is impractical, leading to "misplaced" images which do not vary continuously over different poses.

- *Constant False Alarm Rate* (CFAR) is commonly used to generate point clouds from radar images. Compared to lidar point clouds, CFAR point clouds are sparse and low-resolution, but capture radar specific properties not measured in lidar. However, CFAR cannot provide any notion of the *size* of each point or its transmittance, which requires the point or grid size to be manually tuned, leading to either excessively sparse or blurry images.

DART therefore achieves its efficacy by using a domain-appropriate sensor and carefully selecting a representation which allows it to use all available sensor information.

5.2. Tomography and Mapping

While DART is not designed primarily as an *explicit* tomography or mapping tool, we can sample the implicit representation[7] to create a (σ, α) reflectance and transmittance grid. This also allows us to verify that DART truly learns the mmWave properties of a scene (and does not simply memorize and interpolate the training data).

Material Properties Example We created an evaluation scene with 5 different boxes. DART is able to learn the unique reflectance and transmittance properties of each materials, which we visualize through tomographic reflectance and transmittance maps (Fig. 8). For additional examples from our datasets, see Appendix C.3.

Comparison with Baselines In addition to creating more accurate radar simulations, DART can also produce more accurate and dense maps than CFAR. Fig. 7 shows several examples comparing tomographic maps of reflectance learned by DART with corresponding slices of the point cloud generated by CFAR. While not as sharp as lidar scans,

---

[7]To address view dependence, we analytically take the spherical integral of σ and α at each point; see Appendix A.3 for details.
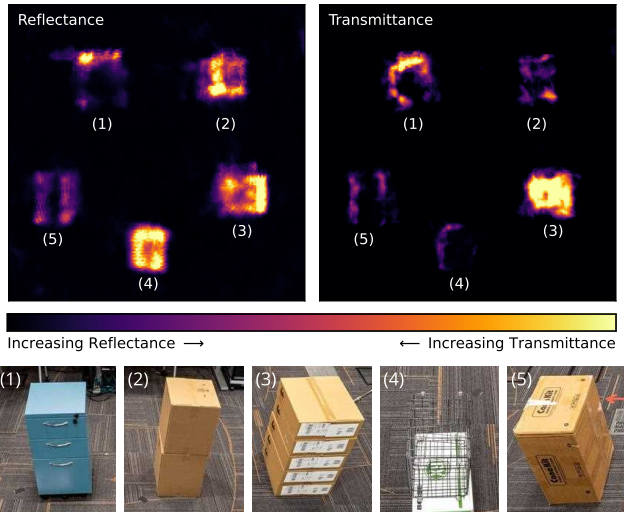

Figure 8. Tomographic images of 5 boxes made from different materials: (1) a metal filing cabinet which appears less reflective (due to specularity), but blocks radar waves; (2) an empty box which reflects radar waves but does not block them; (3) a stack of boxes containing electronics equipment which both reflect and block radar waves; (4) a highly reflective metal mesh with large holes that allow radar to penetrate it; and (5) a different empty box which neither reflects nor blocks radar waves.

DART produces reasonably clear maps which capture the radar-specific properties of each scene.

6. Conclusion

We present DART: Doppler Aided Radar Tomography, a NeRF-inspired radar novel view synthesis algorithm which learns an implicit tomographic map from range-Doppler images, and demonstrate its effectiveness against state-of-the-art baselines. We derive a physics-based rendering model for radar from first principles, and construct an end-to-end system for learning an implicit scene representation and generate realistic novel radar views. While DART provides a strong baseline for future work, many opportunities remain to apply lessons learned from visual NeRFs; given the rapid pace of innovation in NeRF, these opportunities will likely multiply in the coming years. We also currently make a number of assumptions – such as a static scene and the availability of accurate ground-truth pose – which could be relaxed as has been done with visual NeRFs, enabling a single-chip radar solution for localization, mapping, and imaging. Finally, as we add mmWave radar to the repertoire of NeRF-enabled sensing technologies, this furthers the potential for multimodal implicit mapping in the future.

References

[1] Constant false alarm rate (cfar) detection. Url: https://www.mathworks.com/help/phased/ug/constant-

false-alarm-rate-cfar-detection.html; Accessed: 2023-11-15. 6, 8

[2] Yasin Almalioglu, Mehmet Turan, Chris Xiaoxuan Lu, Niki Trigoni, and Andrew Markham. Milli-rio: Ego-motion estimation with low-cost millimetre-wave radar. *IEEE Sensors Journal*, 21(3):3314–3323, 2020. 1, 3

[3] Stefan Auer, Richard Bamler, and Peter Reinartz. Raysar - 3d sar simulator: Now open source. 2016. 1, 3, 6

[4] Kshitiz Bansal, Keshav Rungta, Siyuan Zhu, and Dinesh Bharadia. Pointillism: Accurate 3d bounding box estimation with multi-radars. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, pages 340–353, 2020. 4

[5] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021. 3

[6] M. S. Bartlett. Periodogram analysis and continuous spectra. *Biometrika*, 37(1/2):1–16, 1950. 6

[7] Nitin Bhatia et al. Survey of nearest neighbor techniques. *arXiv preprint arXiv:1007.0085*, 2010. 6

[8] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. 5

[9] Sarah H Cen and Paul Newman. Precise ego-motion estimation with millimeter-wave radar under diverse and challenging conditions. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6045–6052. IEEE, 2018. 1

[10] Yuwei Cheng, Jingran Su, Mengxin Jiang, and Yimin Liu. A novel radar point cloud generation method for robot environment perception. *IEEE Transactions on Robotics*, 2022. 3

[11] 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. 3

[12] 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. 3

[13] Qingyun di and Miaoyue Wang. Migration of ground-penetrating radar data method with a finite-element and dispersion. *Geophysics*, 69, 2004. 3, 6

[14] Fangqiang Ding, Andras Palffy, Dariu M. Gavrila, and Chris Xiaoxuan Lu. Hidden gems: 4d radar scene flow learning using cross-modal supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9340–9349, 2023. 1

[15] 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. 3

[16] Jacques R. Ernst, Hansruedi Maurer, Alan G. Green, and Klaus Holliger. Full-waveform inversion of crosshole radar data based on 2-d finite-difference time-domain solutions of maxwell's equations. *IEEE Transactions on Geoscience and Remote Sensing*, 45(9):2807–2828, 2007. 3

[17] Shiwei Fang and Shahriar Nirjon. Superrf: Enhanced 3d rf representation using stationary low-cost mmwave radar. In *International Conference on Embedded Wireless Systems and Networks (EWSN)...*, page 120. NIH Public Access, 2020. 3

[18] Sara Fridovich-Keil, Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5501–5510, 2022. 3, 5, 2

[19] 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. 3

[20] Xiangyu Gao, Sumit Roy, and Guanbin Xing. Mimo-sar: A hierarchical high-resolution imaging algorithm for mmwave fmcw radar in autonomous driving. *IEEE Transactions on Vehicular Technology*, 70(8):7322–7334, 2021. 3

[21] Andrew Geiss and Joseph C Hardin. Radar super resolution using a deep convolutional neural network. *Journal of Atmospheric and Oceanic Technology*, 37(12):2197–2207, 2020. 3

[22] Junfeng Guan, Sohrab Madani, Suraj Jog, Saurabh Gupta, and Haitham Hassanieh. Through fog high-resolution imaging using millimeter wave radar. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[23] Junfeng Guan, Sohrab Madani, Suraj Jog, Saurabh Gupta, and Haitham Hassanieh. Through fog high-resolution imaging using millimeter wave radar. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11464–11473, 2020. 3

[24] Wolfgang Hess, Damon Kohler, Holger Rapp, and Daniel Andor. Real-time loop closure in 2d lidar slam. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 1271–1278. IEEE, 2016. 6, 8

[25] 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. 3

[26] Franz Hlawatsch. *Time-frequency analysis and synthesis of linear signal spaces: time-frequency filters, signal detection and estimation, and Range-Doppler estimation*. Springer Science & Business Media, 1998. 2

[27] 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. 3

[28] Cesar Iovescu and Sandeep Rao. The fundamentals of millimeter wave sensors. *Texas Instruments*, pages 1–8, 2017. 4, 1

9

[29] JR Jamora, Dylan Green, Ander Talley, and Thomas Curry. Utilizing sar imagery in three-dimensional neural radiance fields-based applications. In *Algorithms for Synthetic Aperture Radar Imagery XXX*, page 1252002. SPIE, 2023. 3

[30] Mahfuza Khatun, Hani Mehrpouyan, David Matolak, and Ismail Guvenc. Millimeter wave systems for airports and short-range aviation communications: A survey of the current channel models at mmwave frequencies. In *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*, pages 1–8, 2017. 1

[31] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[32] Atsutake Kosuge, Satoshi Suehiro, Mototsugu Hamada, and Tadahiro Kuroda. mmwave-yolo: A mmwave imaging radar-based real-time multiclass object recognition system for adas applications. *IEEE Transactions on Instrumentation and Measurement*, 71:1–10, 2022. 3

[33] 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. 3

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

[35] 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. 3

[36] Peizhao Li, Pu Wang, Karl Berntorp, and Hongfu Liu. Exploiting temporal relations on radar perception for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17071–17080, 2022. 1

[37] Yu-Jhe Li, Jinhyung Park, Matthew O'Toole, and Kris Kitani. Modality-agnostic learning for radar-lidar fusion in vehicle detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 918–927, 2022. 1

[38] Yu-Jhe Li, Shawn Hunt, Jinhyung Park, Matthew O'Toole, and Kris Kitani. Azimuth super-resolution for fmcw radar in autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17504–17513, 2023. 4

[39] Jaime Lien, Nicholas Gillian, M Emre Karagozler, Patrick Amihood, Carsten Schwesig, Erik Olson, Hakim Raja, and Ivan Poupyrev. Soli: Ubiquitous gesture sensing with millimeter wave radar. *ACM Transactions on Graphics (TOG)*, 35(4):1–19, 2016. 3

[40] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *Advances in Neural Information Processing Systems*, 33:15651–15663, 2020. 3

[41] Yunfei Long, Daniel Morris, Xiaoming Liu, Marcos Castro, Punarjay Chakravarty, and Praveen Narayanan. Radar-camera pixel depth association for depth completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12507–12516, 2021. 4

[42] Chris Xiaoxuan Lu, Stefano Rosa, Peijun Zhao, Bing Wang, Changhao Chen, John A Stankovic, Niki Trigoni, and Andrew Markham. See through smoke: robust indoor mapping with low-cost mmwave radar. In *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, pages 14–27, 2020. 3

[43] Chris Xiaoxuan Lu, Muhamad Risqi U Saputra, Peijun Zhao, Yasin Almalioglu, Pedro PB De Gusmao, Changhao Chen, Ke Sun, Niki Trigoni, and Andrew Markham. milliego: single-chip mmwave radar aided egomotion estimation via deep sensor fusion. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, pages 109–122, 2020. 1, 3

[44] Andrew Luo, Yilun Du, Michael Tarr, Josh Tenenbaum, Antonio Torralba, and Chuang Gan. Learning neural acoustic fields. In *Advances in Neural Information Processing Systems*, pages 3165–3177. Curran Associates, Inc., 2022. 3

[45] M. Malinen and P. Råback. *Elmer finite element solver for multiphysics and multiscale problems*. Forschungszentrum Juelich, 2013. 3

[46] 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

[47] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *CVPR*, 2021. 3

[48] 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, 5

[49] G. Minkler and J. Minkler. CFAR: The principles of automatic radar detection in clutter. *NASA STI/Recon Technical Report A*, 90:23371, 1990. 3, 6

[50] 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

[51] 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. 3, 5, 2

[52] 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

[53] Akarsh Prabhakara, Diana Zhang, Chao Li, Sirajum Munir, Aswin C Sankaranarayanan, Anthony Rowe, and Swarun Kumar. Exploring mmwave radar and camera fusion for high-resolution and long-range depth imaging. In *2022*

IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 3995–4002. IEEE, 2022. 3

[54] Akarsh Prabhakara, Tao Jin, Arnav Das, Gantavya Bhatt, Lilly Kumari, Elahe Soltanaghaei, Jeff Bilmes, Swarun Kumar, and Anthony Rowe. High resolution point clouds from mmwave radar. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023. 3

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

[56] 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

[57] Kun Qian, Shilin Zhu, Xinyu Zhang, and Li Erran Li. Robust multimodal vehicle detection in foggy weather using complementary lidar and radar signals. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 444–453, 2021. 1

[58] 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 (CVPR)*, pages 17021–17030, 2022. 2

[59] 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

[60] Mark A Richards, Jim Scheer, William A Holm, and William L Melvin. *Principles of Modern Radar*. Citeseer, 2010. 4, 2, 5

[61] Mark A Richards et al. *Fundamentals of radar signal processing*. Mcgraw-hill New York, 2005. 4, 2

[62] Christian P Robert, George Casella, and George Casella. *Monte Carlo statistical methods*. Springer, 1999. 8

[63] Hermann Rohling. Radar cfar thresholding in clutter and multiple target situations. *IEEE Transactions on Aerospace and Electronic Systems*, AES-19(4):608–621, 1983. 3

[64] Mark E Russell, Arthur Crain, Anthony Curran, Richard A Campbell, Clifford A Drubin, and Willian F Miccioli. Millimeter-wave radar sensor for automotive intelligent cruise control (icc). *IEEE Transactions on microwave theory and techniques*, 45(12):2444–2453, 1997. 1

[65] Nicolas Scheiner, Florian Kraus, Fangyin Wei, Buu Phan, Fahim Mannan, Nils Appenrodt, Werner Ritter, Jurgen Dickmann, Klaus Dietmayer, Bernhard Sick, and Felix Heide. Seeing around street corners: Non-line-of-sight detection and tracking in-the-wild using doppler radar. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[66] 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. 3, 6

[67] 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. 3

[68] David M Sheen, Douglas L McMakin, and Thomas E Hall. Cylindrical millimeter-wave imaging technique for concealed weapon detection. In *26th AIPR Workshop: Exploiting New Image Sources and Sensors*, pages 242–250. SPIE, 1998. 1

[69] Xian Shuai, Yulin Shen, Yi Tang, Shuyao Shi, Luping Ji, and Guoliang Xing. millieye: A lightweight mmwave radar and camera fusion system for robust object detection. In *Proceedings of the International Conference on Internet-of-Things Design and Implementation*, pages 145–157, 2021. 3

[70] Akash Deep Singh, Sandeep Singh Sandha, Luis Garcia, and Mani Srivastava. Radhar: Human activity recognition from point clouds generated through a millimeter-wave radar. In *Proceedings of the 3rd ACM Workshop on Millimeter-wave Networks and Sensing Systems*, pages 51–56, 2019. 3

[71] William Snyder, Stephen DelMarco, Dylan Snover, Amit Bhatia, and Scott Kuzdeba. Extending neural radiance fields (nerf) for synthetic aperture radar (sar) novel image generation. In *Synthetic Data for Artificial Intelligence and Machine Learning: Tools, Techniques, and Applications*, pages 268–276. SPIE, 2023. 3

[72] Yue Sun, Honggang Zhang, Zhuoming Huang, and Benyuan Liu. Deeppoint: A deep learning model for 3d reconstruction in point clouds via mmwave radar. *arXiv preprint arXiv:2109.09188*, 2021. 3

[73] Matthew Tancik, Vincent Casser, Xinchen Yan, Sabeek Pradhan, Ben Mildenhall, Pratul P Srinivasan, Jonathan T Barron, and Henrik Kretzschmar. Block-nerf: Scalable large scene neural view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8248–8258, 2022. 3

[74] 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. 5, 2

[75] 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. 3, 6

[76] *AWR1843AOP Single-chip 77- and 79-GHz FMCW mmWave Sensor Antennas-OnPackage (AOP)*. Texas Instruments, 2021. 2, 4, 5

[77] Yuheng Wang, Haipeng Liu, Kening Cui, Anfu Zhou, Wensheng Li, and Huadong Ma. m-activity: Accurate and real-time human activity recognition via millimeter wave radar. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8298–8302. IEEE, 2021. 3

[78] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6

11

[79] Christoph Wasserzier, Josef G. Worms, and Daniel W. O'Hagan. How noise radar technology brings together active sensing and modern electronic warfare techniques in a combined sensor concept. In *2019 Sensor Signal Processing for Defence Conference (SSPD)*, pages 1–5, 2019. 4

[80] Hongfei Xue, Qiming Cao, Yan Ju, Haochen Hu, Haoyu Wang, Aidong Zhang, and Lu Su. M4esh: mmwave-based 3d human mesh construction for multiple subjects. In *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems*, pages 391–406, 2022. 3

[81] 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

[82] 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

[83] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. Plenoctrees for real-time rendering of neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5752–5761, 2021. 3, 5, 2

[84] Xiaopeng Zhao, Zhenlin An, Qingrui Pan, and Lei Yang. Nerf2: Neural radio-frequency radiance fields. *arXiv preprint arXiv:2305.06118*, 2023. 3

[85] Zihao Zhao, Yuying Song, Fucheng Cui, Jiang Zhu, Chunyi Song, Zhiwei Xu, and Kai Ding. Point cloud features-based kernel svm for human-vehicle classification in millimeter wave radar. *IEEE Access*, 8:26012–26021, 2020. 3

[86] Yi Zhou, Lulu Liu, Haocheng Zhao, Miguel López-Benítez, Limin Yu, and Yutao Yue. Towards deep radar perception for autonomous driving: Datasets, methods, and challenges. *Sensors*, 22(11), 2022. 4

DART: Implicit Doppler Tomography for Radar Novel View Synthesis

Supplementary Material

A. Method Details

DART represents the sum of careful consideration of each step in the novel view synthesis pipeline from radar pre-processing to rendering. We provide additional details not included in our main paper, including our radar processing pipeline (Sec. A.1), spherical harmonics representation (Sec. A.3), transmittance representation and custom gradient estimator (Sec. A.4), optimized radar rendering pipeline (Sec. A.6), and hyperparameters (Sec. A.7).

A.1. Radar Pre-Processing

Our radar processing pipeline (Fig. 9) consists of a range, Doppler, and azimuth fast Fourier transform (FFT). The range and Doppler FFT use a Hann window in order to suppress undesirable side lobes. Finally, we perform range decimation to take the first 128 valid range bins of each processed range-Doppler-azimuth heatmap, and apply calibration to account for antenna delay.

Note that since we do not have sub-mm pose accuracy or restrict data collection to fixed trajectories, DART does not use coherent radar images. As such, while each FFT produces complex values, we only use the magnitude of the final output.

Range FFT State-of-the-art solid-state compact radars operate in the radio frequency regime of millimeter wave (mmWave) frequencies (60 GHz / 77 GHz). The wide bandwidths (about 4 GHz) available at these frequencies provide the impressive range resolution of mmWave radars of around 4cm ($\frac{c}{2B}$, where c is speed of light and B is the bandwidth).

This means that the radar can distinguish objects in the world separated by 4 cm apart purely based on range processing. Since the range of an object manifests in the time delay of its reflection, mmWave radars send "chirps" in a waveform called Frequency Modulated Continuous Wave (FMCW) which converts time delays to frequency shifts [28]. We perform range processing by simply computing a 1D Fourier transform of the received reflections which converts the frequency shifts back to time delays that is proportional to range (Range $R = \frac{c\Delta T}{2}$, where c is speed of light and ΔT is the two-way time delay).

The time window that this Fourier transformed is performed over is referred to as *fast time*, and is equal to the maximum theoretical number of range bins.

Doppler FFT We next compute the Doppler velocities by processing a time series of FMCW waveforms. Unlike
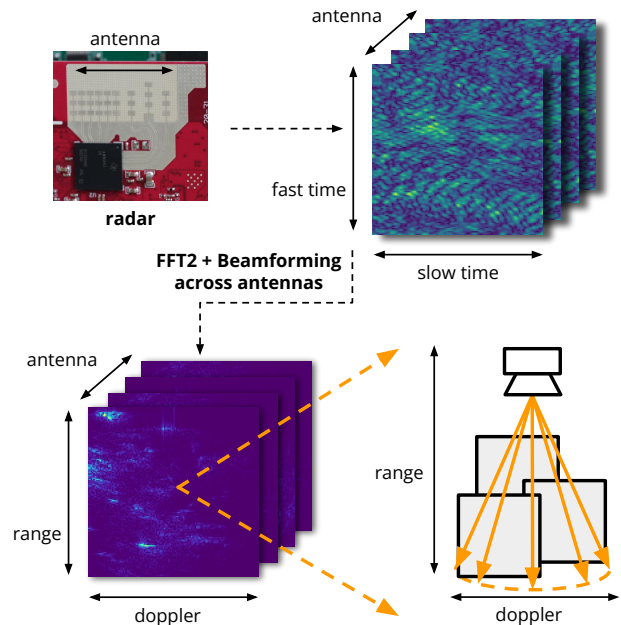

Figure 9. DART performs range-Doppler pre-processing on each of the antennas in the mmWave radar using 2D FFTs and antenna beamforming gains. A key point to note is the reduction in range-Doppler sidelobes using signal processing windows prior to FFTs for effective DART training.

range information which is observed as time delay of reflections, Doppler velocities manifest as phase change of the reflected signal when there is relative motion between radar and the world. As the radar moves, an object changes phase as the range R changes slightly over a time window

$$\exp\left(\frac{j2\pi 2R(t)}{\lambda}\right), \tag{6}$$

where λ is the wavelength corresponding to the FMCW waveform's center frequency. Because of the tiny wavelengths (4 mm) of mmWave signals, even tiny shifts in range show up as significant phase changes. The rate of change of these phase changes over the time series of FMCW waveforms provides the Doppler velocity.

Similar to range processing, we perform another Fourier Transform, but across a time sequence of range processed FMCW reflections. The time window (across different chirps) which this Fourier transform is performed over is referred to as *slow time*.

Impact of Doppler Resolution Unlike azimuth or elevation resolution which is governed by the number of antennas in a radar system, the Doppler resolution depends only

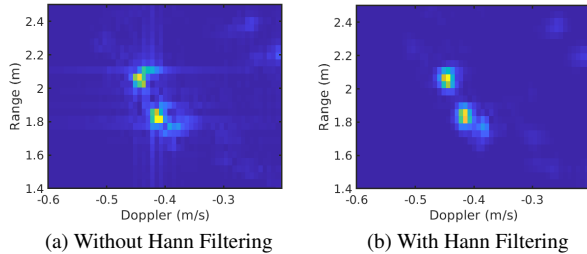(a) Without Hann Filtering   (b) With Hann Filtering

Figure 10. Spectral leakage causes inter-bin interference, visible as streaking in the horizontal and vertical directions (range-Doppler images are cropped to a small window). Hann low-pass filtering softens window edges and reduces spectral leakage significantly at the cost of reduced sharpness.

on the radar frame integration time. In theory, we can have arbitrarily fine Doppler resolution with long frame times; however, this assumes constant radar velocity over the entire integration time window, which is impractical.

Moreover, Doppler resolution can be simplified and mapped to azimuth resolution (in 2D) as

$$\Delta\theta = \frac{\Delta D \lambda}{2 v \sin\theta} \quad (7)$$

Therefore, as ΔD becomes lower with longer frame integration time, the azimuth resolution becomes better. In our case, the (theoretical) azimuth resolution derived from Doppler is $0.85°$ for a head-on point at $\theta=90°$, a radar speed of 0.5 m/s and a radar frame integration time of 256 ms — much better than the best case of $15°$ that compact radars today are capable of [58, 76].

Hann Filtering   Since both the range FFT and Doppler FFT are taken over a finite-time window, the hard edges of the time window induce high-frequency effects that result in a phenomenon known as *spectral leakage*. This leakage is a mathematical artifact that can be modeled as a $sinc()$ function and is clearly visible in radar scans [60, 61]. While it would be possible to accurately reproduce such artifacts in DART's forward rendering model, this would require modeling inter-column effects (breaking DART's per-column rendering model) and greatly increase the computational cost of our algorithm. Instead, we apply a Hann window weighting to the raw range-Doppler frames before computing the FFT, which slightly blurs the resulting images but greatly reduces the inter-bin artifacts (Fig. 10).

Azimuth FFT   After applying the range and Doppler FFT, we also apply an Azimuth FFT across 8 azimuth bins (2 TX × 4 RX antennas). While this also results in a *sinc* pattern, we model it explicitly in the form of an array factor (Sec. A.5) applied in addition to the antenna gain on each of the resulting azimuth bins.
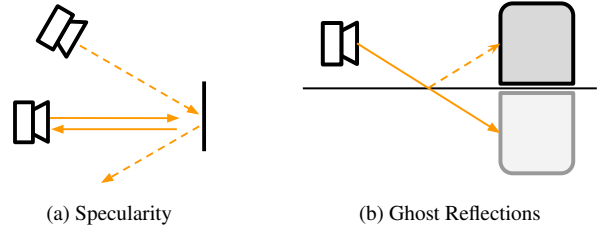

(a) Specularity   (b) Ghost Reflections

Figure 11. View-dependent reflectance and transmittance captures both specularity and ghost reflections.

A.2. Reflectance and Transmittance Modeling

DART's view-dependent transmittance-reflectance model is analogous to the density-color representation of NeRFs, and enables modeling a range of physical phenomena, such as:

1. Diffuse reflections: This makes up for the majority of first order reflections from non-metallic surfaces.
2. Partial occlusions: Modeling transmittance allows objects behind surfaces such as drywall to be visible while occluding objects behind opaque surfaces, e.g. concrete.
3. Specularity (Fig. 11a): This manifests as high reflectance in specific directions, e.g. metallic surfaces.
4. Ghost reflections (Fig. 11b): This occurs as a result of seeing an object after multiple orders of reflections from other objects, e.g. seeing the ghost reflection of a chair off of the ground when radar is looking at the ground.

A.3. Spherical Harmonics

Our usage of spherical harmonics provides view-dependence, while also making interpretation of the resulting implicit map easier. In this section, we describe how our usage of spherical harmonics contrasts with common practice in visual Neural Radiance Fields and provides key advantages for our use case.

Spherical Harmonics Representation   Spherical harmonics are a spherical (orthonormal) basis function which is commonly used to decompose functions defined on the unit sphere. Neural Radiance Field models typically handle view dependence by projecting the incidence angle of the incoming ray to between 9 and 25 (real) spherical harmonic coefficients, which is used as an input to the neural network [51, 74], acting as essentially a positional embedding on the viewing angle. We instead use spherical harmonic coefficients as an *output* similar to non-neural-network based radiance field methods [18, 83].

We use spherical harmonic coefficients of degree 25. Let $Y : \mathbb{R}^3 \to \mathbb{R}^{25}$ be the function that maps incoming incident angles to the degree-25 spherical harmonics. Then, our field

function output is given by

$$\sigma(\boldsymbol{w}) = \bar{\sigma}\langle \boldsymbol{Y}(\boldsymbol{w}), \boldsymbol{c}/||\boldsymbol{c}||_2 \rangle,$$
$$\alpha(\boldsymbol{w}) = f\left(\bar{\alpha}\langle \boldsymbol{Y}(\boldsymbol{w}), \boldsymbol{c}/||\boldsymbol{c}||_2 \rangle\right), \tag{8}$$

where $\bar{\sigma} \in \mathbb{R}, \bar{\alpha} \in \mathbb{R}, \boldsymbol{c} \in \mathbb{R}^{25}$ are outputs of the neural network, f is the activation function on α (Sec. A.4), and $\boldsymbol{w}$ is the incident angle.

Crucially, unlike visual NeRFs, we also must model transmittance α (which is analogous to the density in NeRF) as having view dependence. While reflective surfaces in visual NeRFs are modeled as "screens" whose content changes depending on the viewing angle, radars measure distance, which requires reflective surfaces to instead be measured as "holes" which become transparent depending on the viewing angle.

Interpretability Since spherical harmonics are an orthonormal basis, we can analytically evaluate the spherical $\mathcal{L}_2$ norm of our DART field at each point without requiring numerical integration in order to sample a "mean" reflectance and transmittance. For example:

$$||\sigma||_2^2 = \int_w \left(\bar{\sigma}\langle \boldsymbol{Y}(\boldsymbol{w}), \boldsymbol{c}/(||\boldsymbol{c}||_2)\rangle\right) d\boldsymbol{w}\right)^2$$
$$= \frac{\bar{\sigma}^2}{||\boldsymbol{c}||_2^2} \int_w \boldsymbol{c}^T \boldsymbol{Y}(\boldsymbol{w})\boldsymbol{Y}(\boldsymbol{w})^T \boldsymbol{c} \, d\boldsymbol{w} \tag{9}$$
$$= \bar{\sigma}^2$$

since, by the orthonormality of spherical harmonics $\boldsymbol{Y}$,

$$\int_{\boldsymbol{w}} \boldsymbol{Y}(\boldsymbol{w})\boldsymbol{Y}(\boldsymbol{w})^T \, d\boldsymbol{w} = \boldsymbol{I}, \tag{10}$$

and neural network output coefficients $\boldsymbol{c}$ are unrelated to $\boldsymbol{w}$ given a fixed point in space. If we ignore the activation function f, the same applies to α.

A.4. Transmittance Representation

Each ray in our rendering pipeline corresponds to one sampled reflectance σ from DART. As such, negative σ are immediately penalized by gradient descent, so an activation function on σ is not necessary, and can harm the numerical stability of DART (Sec. C.1).

On the other hand, since each transmittance value σ affects all bins behind it, allowing $\alpha > 1$ (i.e. for points in space to transmit *more* energy than they receive) can allow for a significant degree of overfitting. At the same time, we must account for the fact that most points in space are empty (so it should be "easy" for the model to represent $\alpha = 1$), while handling initialization instability and the possibility of "dead" gradients.

Finally, to further reduce the potential for overfitting, we only allow $\alpha < 1$ when σ exceeds a set threshold (0.05

in our experiments). To reduce the "swiss cheese" effect caused when this threshold is initialized to a (relatively) large value, we also anneal this threshold in over 600 steps (starting from -1, which in practice is always less than the initialized σ, to 0.0 linearly in 100 steps, and from 0.0 to 0.05 linearly in 500 steps).

Existing Activation Functions No existing activation functions satisfy our desired properties:
- Sigmoid: this requires a $-\infty$ input to represent 0, and suffers from vanishing gradients at either end.
- Value clipping $\alpha = \text{clip}(\alpha', 0, 1)$: clipping output values to $[0, 1]$ suffers greatly from "dead" regions in space, especially where α is initialized at $\alpha' < 0$, and cannot recover.
- Partial clipping $\alpha = \exp(\min(0, \alpha'))$: clipping at only one end suffers from the same "dead" region problem, though only at one side.

Indeed, any activation function on [0, 1] must have vanishing gradients of some form. As such, we instead turn to a custom gradient estimator to create an activation function which satisfies our desired properties.

Gradient Estimator Using partial clipping $\alpha = \exp(\min(0, \alpha'))$ as a starting point, we specify our gradient estimator for $\min(0, \alpha')$ as follows:
- $\alpha' \leq 0$: gradients pass through, i.e. $d/d\alpha' = d/d\alpha$.
- $\alpha' > 0, d/d\alpha' \geq 0$: In this case, $d/d\alpha' > 0$, indicating that the gradients are trying to push α' negative, even though $\min(0, \alpha)$ is "saturated". As such, we pass $d/d\alpha' = d/d\alpha$ as well.
- $\alpha' > 0, d/d\alpha' < 0$: In this case, the gradients are trying to push α' positive, even though it is already positive. To prevent our parameter values from exploding, we set $d/d\alpha' = 0$.

A.5. Rendering Equation Derivation Details

DART's rendering process is summarized by Eq. 5 at a high level; in this section, we expand on the derivation of this equation, as well as specific elements of our implementation of the rendering equation.

Field of view calculation While radar rendering theoretically requires integrating spherically around the radar, the gain to the rear of the radar is close to zero in practice[8]. As such, we only integrate across the front of the radar.

Our rendering equation begins by generating an orthonormal basis

$$\left\{p, q, \frac{\boldsymbol{v}}{||\boldsymbol{v}||_2}\right\} \tag{11}$$

---

[8]Our radar uses a PCB-based antenna, which acts as a metal plane preventing the radar from seeing backwards.

(a) Section of the Doppler cone   (b) Integration arc/circle along the p-q plane   (c) Projection of $\boldsymbol{v}$ to the $+x$ axis
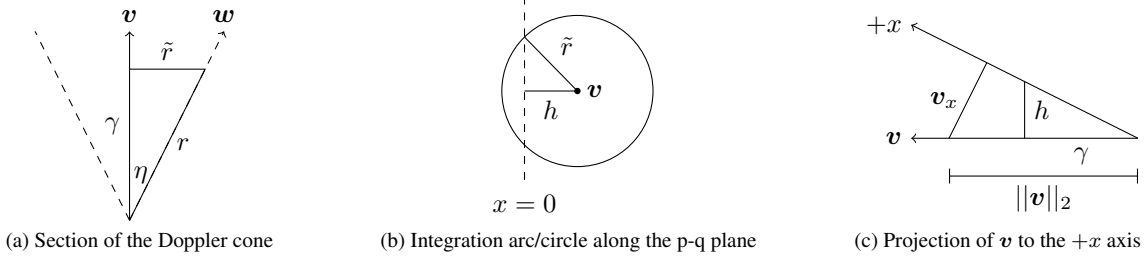
Figure 12. Illustration of our integration arc equation derivation.

for velocity vector $\boldsymbol{v}$. We select p such that its direction is aligned with the positive x axis (which define to be the front of the sensor), i.e.

$$p \propto \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T - \frac{\boldsymbol{v}_x \boldsymbol{v}}{||\boldsymbol{v}||_2^2}, \qquad (12)$$

and define q accordingly.

Each range-Doppler bin is modeled as the intersection of a sphere with radius r and a cone $\langle \boldsymbol{v}, \boldsymbol{w} \rangle = d$. When intersected with the half-space corresponding with $x \geq 0$, this results in an arc or circle on the p-q plane (or the empty set if degenerate). Since we define the sensor to be facing in the $+x$ axis, and define p to be aligned with $+x$, the valid integration arc will always be $(-\psi, \psi)$ for angle ψ counterclockwise from the $+p$ axis, again relative to the p-q plane.

We now calculate this angle ψ. We begin by calculating the distance γ to the center of the integration arc and its radius $\tilde{r}$ (Fig. 12a). Choosing an arbitrary direction $\boldsymbol{w}$ in our p-q-v basis, the cosine of the angle η between $\boldsymbol{v}$ and $\boldsymbol{w}$ yields

$$\cos(\eta) = \frac{\gamma}{r} = \frac{\langle \boldsymbol{v}, \boldsymbol{w} \rangle}{||\boldsymbol{v}||_2 ||\boldsymbol{w}||_2} \implies \gamma = \frac{rd}{||\boldsymbol{v}||_2} \quad (13)$$

using the definition of Doppler d and orthonormality of $\boldsymbol{w}$. It follows that

$$\tilde{r} = \sqrt{r^2 - \frac{r^2 d^2}{||\boldsymbol{v}||_2}} = r\sqrt{1 - \frac{d^2}{||\boldsymbol{v}||_2^2}}. \quad (14)$$

Next, we calculate the length of the chord corresponding to the integration arc $\pm \psi$. Let h be the distance from the chord to $\boldsymbol{v}$ (Fig. 12b); h can be specified relative to γ and the angle θ between $\boldsymbol{v}$ and the $+x$ axis (Fig. 12c). It follows that

$$h = \gamma \tan(\theta) = \frac{rd}{||\boldsymbol{v}||_2} \left(\frac{\boldsymbol{v}_x}{\sqrt{||\boldsymbol{v}||_2^2 - \boldsymbol{v}_x^2}} \right), \quad (15)$$

which finally yields (Fig. 12b)

$$\psi = \arccos\left(\frac{h}{\tilde{r}} \right). \quad (16)$$
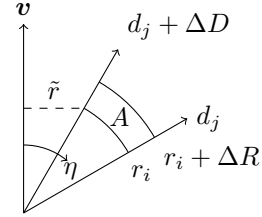
We also must account for three degenerate cases:

Figure 13. Annular section A for the revolved solid approximation of Eq. 5. η is the angle to the velocity vector $\boldsymbol{v}$.

- When $d > ||\boldsymbol{v}||_2$, the Doppler bin exceeds the velocity (and Eq. 14 becomes imaginary).
- When $h > \tilde{r}$, the entire doppler cone is pointing behind the radar (i.e. the $x > 0$ half-space no longer intersects with the cone; Eq. 16 is undefined).
- When $h < -\tilde{r}$, the entire doppler cone is in front of the radar. In this case, we instead have $\psi = \pi$.

Volume of integration region   Expanding on Eq. 4 more explicitly using our integration bound ψ, the radar return $\boldsymbol{Y}(r_i, d_j, k)$ can be expressed as the circular integral

$$\boldsymbol{Y}(r_i, d_j, k) \propto \tilde{r} \int_{-\psi \leq \phi \leq \psi} C(r, k, \boldsymbol{w}(\phi)) \, d\phi \quad (17)$$

for projection $\boldsymbol{w}$ to the integration arc.

While our rendering equation (Eq. 4) makes a "thin-shell" assumption, we must also examine the effect of *bin size* in order to correctly determine which constants to apply to our approximation. Let ΔR and ΔD be the width of each range and Doppler bin. Instead of using a "point" approximation of C, we instead derive the integral based on the revolved solid of an annular section A (Fig. 13). Using again the fact that $\cos(\eta) = d/||\boldsymbol{v}||_2$, we arrive at the approximation

$$dA = r_i \, dr d\theta = \frac{r_i}{||\boldsymbol{v}||_2 \sqrt{1 - \frac{d_j^2}{||\boldsymbol{v}||_2^2}}} \, dr dd$$

$$= \frac{r_i^2}{||\boldsymbol{v}||_2 \tilde{r}} \, dr dd, \quad (18)$$
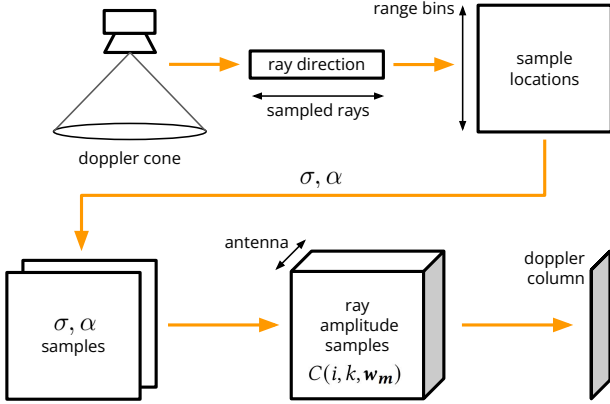
4

Figure 14. Radar rendering dataflow.

which yields the net correction of $r_i^2/||\boldsymbol{v}||_2$ found in Eq. 5 when combined with Eq. 17.

Array factor and antenna gain In our rendering process, we begin by applying the antenna gain for the radar as a whole, which is specified by the AWR1843Boost datasheet [76]. After performing the azimuth FFT, each azimuth bin (i.e. transmit-receive pair) then acts as an independent beamformed radar with a narrower azimuth gain; as such, we also apply the array factor to each azimuth bin [60].

A.6. Optimized Radar Rendering

Figure 14 shows our optimized rendering dataflow:
1. We use each Doppler "column" across all antennas as a sample which has (range, antenna) dimensions.
2. For this Doppler, we sample a fixed (128) number of rays (i.e. from $(-\psi, \psi)$, projecting to 3D space via the v-p-q basis); we trace each ray (Eq. 3), sampling σ and α once at each range bin for a total of (Doppler, ray samples) times.
3. For each antenna, we apply the corresponding gain and sum across the Doppler samples (Eq. 5), yielding (range, antenna) outputs for this Doppler column.

In total, we sample the field (range, Doppler, ray samples) times per image per epoch, which is comparable to the (width, height, range samples) times in NeRF. With the dimensions of images in our datasets (128 range bins, 256 Doppler bins, 128 ray samples, and 8 antenna), this equates to only 16 field samples per output value — less than most visual NeRF methods today such as Nerfacto [74], which samples 48 points per pixel.

A.7. Training and other Hyperparameters

Implementation and Training DART is implemented using JAX [8], which we tune for training on the RTX 4090 (24GB of VRAM). In our implementation, we perform the ray-tracing cumulative product in the log-space as

Figure 15. Our data collection system consists of a handheld rig (Fig. 6) and a backpack which carries power equipment.

an exp-sum-log, learning the log-transmittance instead. We also absorb any constants (including the square for transmittance) into σ and α.

We trained our (σ, α) field function using stochastic gradient descent with the Adam optimizer and an l_1 objective with a learning rate of 0.01 for 3 epochs with a batch size of 1024 "Doppler columns" (i.e. all range-Doppler-antenna bins with the same Doppler value in a given image; see Sec. A.6). For each Doppler column, we sampled 128 rays along the arc defined by that Doppler bin intersected with the half-sphere corresponding to the front of the radar (since our radar is forward-facing).

Architecture Hyperparameters For our base model, we use the Instant Neural Graphics Primitive [51] with the following parameters:
- Hash table size: 2^{20}.
- Number of features per hash table level: 2.
- Hash table resolution scale factor: $2^{0.43} \approx 1.347$ (selected to avoid "aliasing" from different hash table levels lining up to multiples of each other).
- Number of hash table levels: 12.
- Grid resolution: 25cm (coarsest) to 0.94cm (finest).
- Output MLP: 2 hidden layers of 64 and 32 units.

B. Datasets and Evaluation

In this section, we provide additional details about our data collection system (B.1), collected radar traces (B.2), baselines (B.3), and evaluation methodology (B.4).

B.1. Data Collection System

Our data collection rig (Fig. 15) is handheld and fully portable, and consists of a TI AWR1843Boost mmWave radar with a DCA1000EVM capture card for recording

Dataset	Length	Training Time	Description
Lab 1	4:55	8:43	A lab space with 5 boxes of varying material placed inside.
Lab 2	4:24	7:22	
Office 1	10:37	15:14	An office space with cubicles and an enclosed meeting room.
Office 2	15:39	22:18	
Yard	7:38	12:13	A backyard and adjacent detached garage.
House 1	7:55	9:53	A early 20th century house with two above-ground floors. Traces include only the
House 2	9:48	12:54	ground floor, and both the ground and 2nd floor.
Apartment 1	6:04	9:26	A late 20th century high rise one-bedroom apartment; traces include just the
Apartment 2	6:38	10:30	open-plan living area, and both the living area and bedroom.
Rowhouse 1	7:41	10:48	Traces of varying lengths and room inclusions in a 2-bedroom apartment with an
Rowhouse 2	7:37	10:41	open-plan living space in a low-rise townhouse.
Rowhouse 3	10:51	15:17	

Table 2. Total trace length and training times (minutes) for our method on each trace; the training time of our method is between 1-2× the data collection time. While we captured multiple traces from each location, each trace was collected independently.
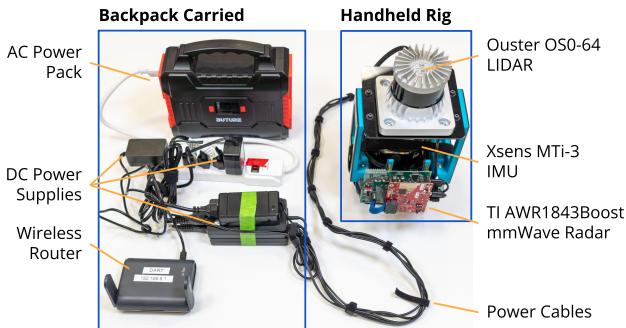

Figure 16. Components of our data collection system.

raw I/Q frames. The data collection rig also includes an Ouster OS0-64 64-beam LIDAR and a Xsens MTi-3 IMU, which are used for our LIDAR-based radar simulator baseline (Appendix B.3) as well as pose estimates using Cartographer SLAM [24]. These components, along with the backpack-carried power and control components, are shown in Fig. 16.

Radar Configuration We configured the radar for interchirp period of 1ms with 512 samples per chirp, and processed the data into range-Doppler images with a frame length of 256 chirps computed on a rolling window with a stride of 64 chirps. This gives a maximum Doppler of ±0.95 m/s across 256 Doppler bins and a maximum range of 21.6m across 512 range bins, which we truncated to the first 128 range bins (5.4m) after removing bins with negative range after radar calibration.

Radar Processing To avoid degenerate Doppler bins at low speeds (violating our thinness assumption) and aliased speeds exceeding the maximum Doppler velocity, we removed frames with speeds below 0.2m/s and above 0.95m/s

from our training and validation datasets. We also use all 8 azimuth antennas by configuring transmitters to operate in a time-division multiplex mode.

Pose Processing DART relies on accurate velocity estimates; however, SLAM systems such as Cartographer are typically not designed to produce accurate or continuous velocity estimates, especially at the frequency that we generate range-Doppler images (64ms). This can lead to erratic velocity estimates, especially if loop closure results in discontinuous pose estimates.

To calibrate the estimated speed, we implemented a speed estimation algorithm which uses simple thresholding to detect the maximum velocity detected in a range-Doppler image. Assuming a static scene, this then indicates the speed of the radar. Using this estimate, we tuned a gaussian smoothing parameter applied to the SLAM pose estimates to produce our final velocity estimates.

In our datasets, we observed that false or sudden relocalizations can also cause erratic speed estimates. Thus, as a final check, we calculated the acceleration of the final speed estimates, and excluded frames (and ±15 neighbors) if the acceleration exceeded 2.0 m/s^2. Datasets that we collected that included a high frequency of likely SLAM system failures were then either recollected or excluded.

B.2. Collected Traces

Our 12 collected radar novel view synthesis traces include 6 unique environments across residential and commercial spaces of various construction. Each trace consists of several passes through each scene with different paths, velocities, and orientations, and is collected with a full pass through the scene at the end to enable a convenient test set. We summarize these traces in Table 2, and provide a visual map of the lidar scan and DART's radar tomography in Fig. 17.
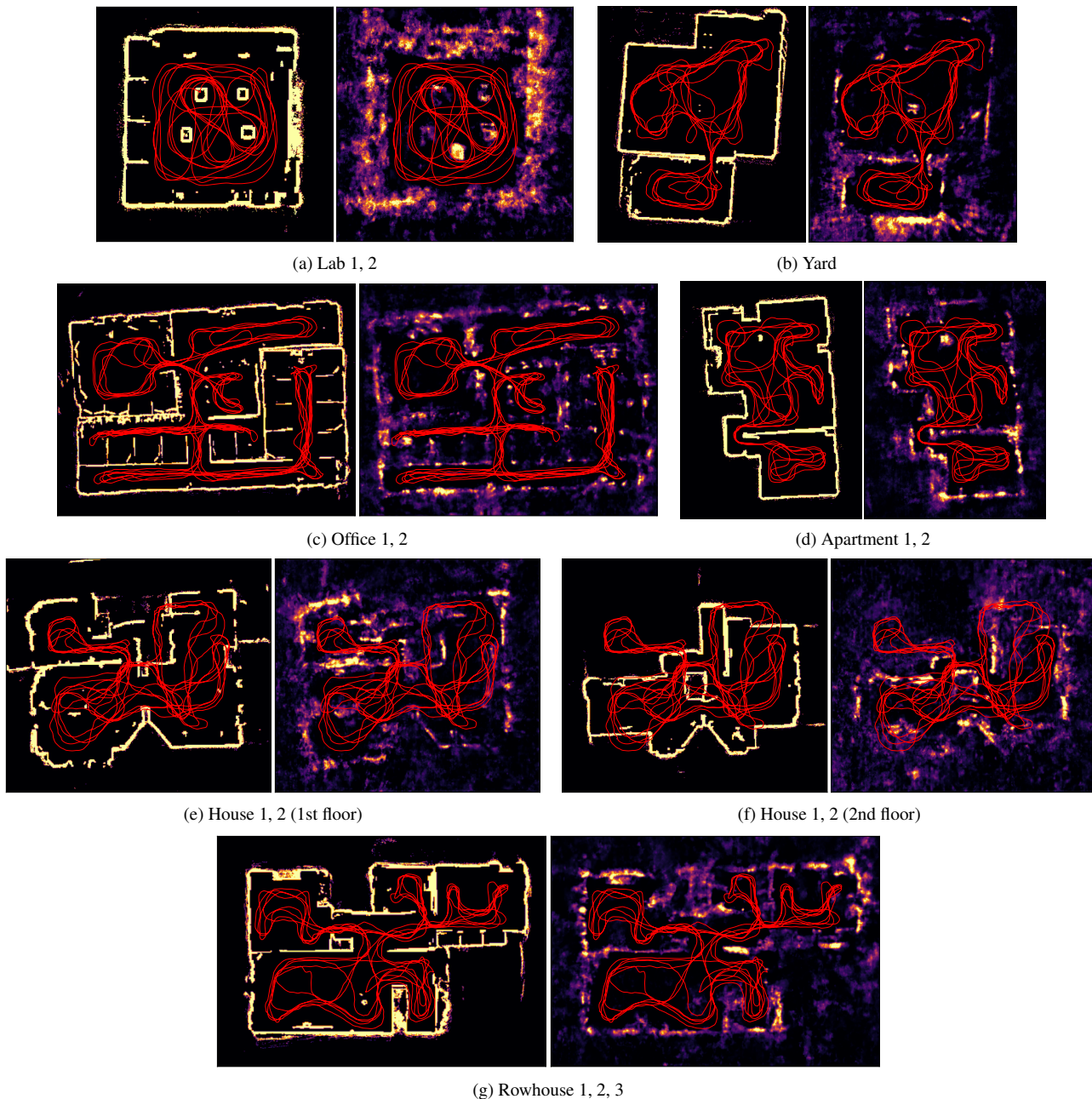
Figure 17. Lidar maps (left), radar tomography maps (right), and sample trajectories (red) on the 6 locations used to collect our novel radar view synthesis dataset. Multiple traces were collected from each location, with each trace being completely independent (and in some cases visiting different rooms). Note that *House* contains two floors; the trajectory shown here is not separated by floor.

Trajectory Like many off-the-shelf radars, the TI AWR1843 radar is designed for automotive applications, and has a relatively narrow vertical field of view ($\pm 20°$, compared to $\pm 50°$ horizontally). As such, we took multiple passes with varying attitude (e.g. horizontal, up, and down) in order to ensure full 3D coverage.

Velocity We found Cartographer to be most stable for our handheld data collection platform when moving smoothly at ≈ 0.5m/s, and targeted this speed during each trace. This also corresponds to the velocity range (0.2m/s – 0.95m/s) which we tuned our radar system for.

B.3. Baseline Details

In this section, we provide additional details about the implementation of our baselines.

Lidar Scan-based Simulator Using Cartographer [24], we created an occupancy grid of each scene with a 2cm resolution. Each occupied cell was assigned a reflectance of 1 unit, and a transmittance of 0 (i.e. fully opaque). We then used a ray-tracing simulator (based on our rendering equation) to simulate radar images, with trilinear interpolation applied to the grid during sampling. This simulator used the same settings as DART (128 rays per Doppler bin, 128 range bins, simultaneous range-antenna rendering).

CFAR Point Cloud Aggregation We used the CFAR implementation provided by the Matlab Phased Array System Toolbox [1] with a false alarm rate of 0.01 to select discrete reflectors. Each reflector was assigned an estimated azimuth angle using direction of arrival estimation, projected to 3D space, and aggregated into a common point cloud along with their estimated reflectance amplitude.

Since finite element modeling is impractical with tens of thousands of CFAR points, we projected each reflector to a 5cm grid (which must be coarser than the 2cm lidar-based simulator grid due to lower point cloud density), taking the maximum gain for each point when multiple fall in the same grid cell. Finally, we used the same simulation settings as our lidar baseline to generate range-Doppler images. Note that since CFAR has no way of measuring the transmittance of points in space, we set all grid cells to be fully transparent (i.e. transmittance $\alpha = 1$).

For our comparison of the CFAR point cloud with DART's sampled tomography images (Fig. 7), we qualitatively tuned the visualization to show sharp outlines and the general geometry of the scene to the best of our ability:

- The scatter plot shows points within ± 10cm of the reference plane to ensure we capture a sufficient density of strong reflectors.
- To reduce clutter due to spurious points in the CFAR point cloud, we then removed the 80% weakest reflectors, and scaled the color range in our visualization so that points close to the 80% cutoff point have a color similar to the plot background.

B.4. Evaluation Metrics

In order to evaluate DART and our baselines, we calculate the SSIM of each valid[9] range-Doppler-antenna frame

(treating each antenna as a different channel) after normalizing each image to a set [0, 1] range. Then, to calculate error bars and evaluate the statistical significance of our results, we estimate the effective sample size of each trace in order to calculate standard errors and perform a paired z-test.

Image Scaling Not all of our baselines are capable of producing accurately scaled output images. Since SSIM requires the input dynamic range to be properly scaled, we scale all images relative to the ground truth prior to SSIM calculation as follows:

1. We clip extreme values from the ground truth images (0.1% and 99.9% percentiles), and normalize this range to $[0, 1]$, which we use as the dynamic range for SSIM calculation.
2. We compute the l_2 "optimal scaling" multiplier between the predicted $\hat{y}$ and actual y^* images using scale factor

$$\arg \min_{\xi} ||y^* - \xi \hat{y}||_2^2 = \frac{\hat{y}^T y^*}{\hat{y}^T \hat{y}} \qquad (19)$$

and apply this to $\hat{y}$. We then clip $\xi \hat{y}$ to the same thresholds as y^*, and normalize to [0, 1].
3. We then compute the SSIM, using a dynamic range of 1.

SSIM Calculation Range-Doppler-antenna radar images are mostly sparse, and even contain large regions where nonzero values cannot theoretically be observed (i.e. when Doppler velocities exceed the current actual speed for a sensor in a stationary scene). As such, a naive application of SSIM leads to misleadingly high SSIM, since the SSIM between empty regions is exactly 1.0.

To correct for this, we exclude empty regions of our image, which are defined as areas where the pixel sample mean is less than a set threshold ($\varepsilon = 0.005$ in our dataset).

Sample Size Correction The standard error of the mean (SE) of a measurement X is defined as $\text{std}(X)/\sqrt{N}$. However, this assumes each of the n samples are independent and identically distributed.

Intuitively, sampling the same signal with greater frequency does not give any additional insight into the general performance of a method. Thus, in time series signals such as ours (i.e. SSIM over time), the number of samples must be *corrected* to estimate the effective sample size, which we substitute for N. This is given by [62]

$$N_{\text{eff}} = \frac{N}{1 + 2 \sum_{t=1}^{\infty} \rho_t} \qquad (20)$$

for autocorrelation ρ_t (where t is the delay). In our calculations, we estimate the sum up to $t = N/2$, and clip ρ_t estimates to positive values[10].

---

[9]Since our Doppler FFT "wraps" once we exceed our maximum doppler velocity, we must exclude these frames. Our method also does not work on stationary radars (as we note in our limitations), so frames with a velocity near zero are also excluded.

---

[10]Negative ρ_t can occur for large delays t due to broader trends in our dataset, e.g. scanning different rooms or directions

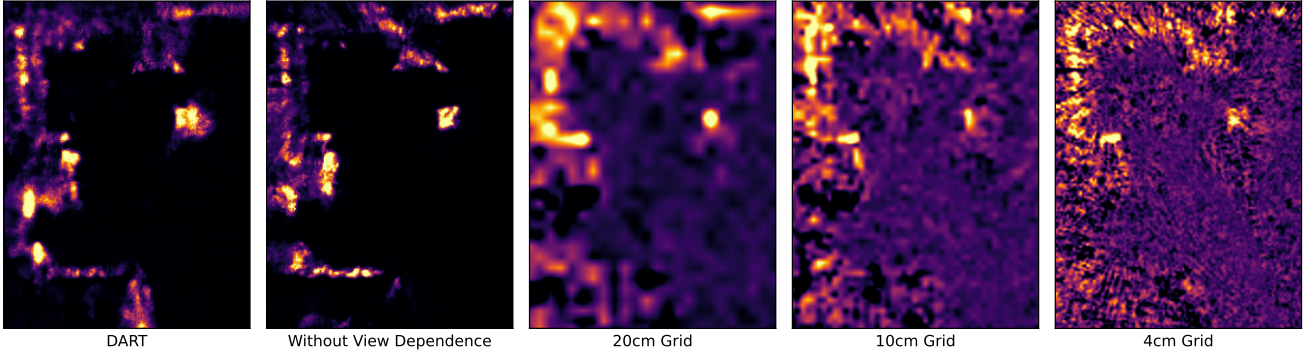|				
DART	Without View Dependence	20cm Grid	10cm Grid	4cm Grid

Figure 18. Sample tomographic maps of the reflectance (σ) compared to using a simple grid with varying resolutions. Coarse grids result in blurry tomographic images, while finer grids result in "holes" in walls and other surfaces. By using an adaptive grid, DART is able to learn tomographic maps which are reasonably sharp but also largely continuous.
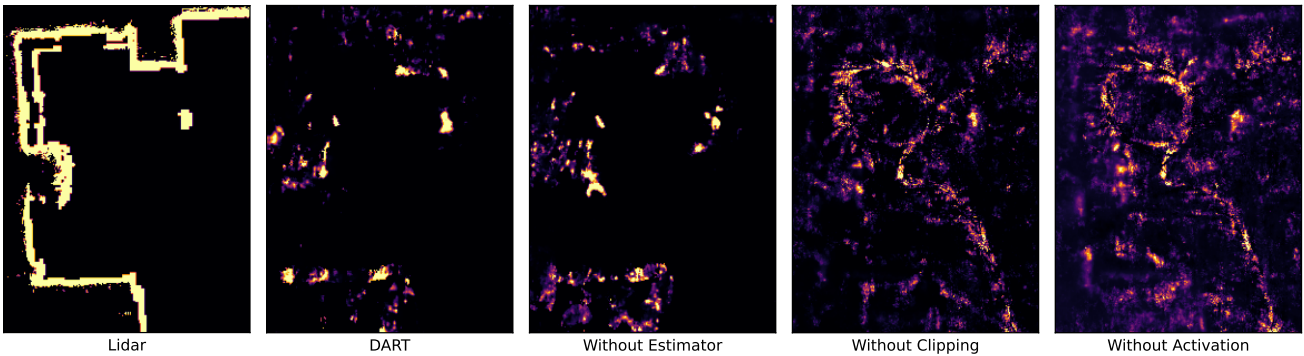

Lidar	DART	Without Estimator	Without Clipping	Without Activation

Figure 19. Sample tomographic maps of the transmittance (α) learned by DART (compared to a Lidar occupancy grid), along with ablations on our representation for α without the gradient estimator (*Without Estimator*), without clipping α when $\sigma < 0.05$ (*Without Clipping*), and without any activation treatment (*Without Activation*).

Paired z-Test   Since each method is evaluated on exactly the same test frames, we use a paired z-test to evaluate the statistical significance of our results. Specifically, we provide error bars for a one-sided p-value of 0.05 (i.e. indicating 95% confidence that one method is better than another).

C. Additional Results

C.1. Ablations

In addition to view dependence, we ran additional ablations showing the impact of DART's implicit representation and transmittance activation function.

Grid Size and Adaptive Grid   To demonstrate the benefit of an (implicit) adaptive grid, we trained a version of DART using a fixed grid of 20cm, 10cm, and 4cm resolutions. DART's adaptive grid leads to more accurate radar simulations compared to fixed grids (Table 3). Examining tomographic slices of the learned reflectance reveals the reason: a coarse grid leads to a blurry image (and therefore blurry synthesized radar range-Doppler images), while

Method	Mean SSIM	Relative SSIM
DART	0.636 ± 0.012	—
No View Dep.	0.614 ± 0.015	0.022 ± 0.005
20cm Grid	0.591 ± 0.015	0.046 ± 0.004
10cm Grid	0.580 ± 0.017	0.056 ± 0.006
4cm Grid	0.564 ± 0.019	0.073 ± 0.008

Table 3. Mean and relative SSIM (to DART) of different grid sizes; we include a baseline which does not use view dependence for comparison since a simple reflectance and transmittance grid does not represent view dependence.

a fine grid leads to holes in the learned reflectance (Fig. 18).

Transmittance Activation Function   While we did not run full ablations for each element of the design of DART's activation function (Sec. A.4), we trained ablations on an example scene (Fig. 19) for the activation function as a whole in addition to ablations on our transmittance clipping procedure and gradient estimator.

• Without any activation function on α, DART can greatly

Dataset	DART	Lidar	Nearest	CFAR
Lab 1	0.676	0.438 (0.238 ± 0.017)	0.490 (0.186 ± 0.033)	0.571 (0.105 ± 0.013)
Lab 2	0.680	0.471 (0.209 ± 0.012)	0.468 (0.212 ± 0.045)	0.566 (0.114 ± 0.014)
Office 1	0.642	0.461 (0.181 ± 0.014)	0.482 (0.160 ± 0.026)	0.555 (0.087 ± 0.013)
Office 2	0.669	0.478 (0.190 ± 0.025)	0.449 (0.220 ± 0.028)	0.562 (0.107 ± 0.019)
Rowhouse 1	0.622	0.458 (0.164 ± 0.023)	0.471 (0.151 ± 0.031)	0.535 (0.087 ± 0.018)
Rowhouse 2	0.619	0.458 (0.161 ± 0.018)	0.450 (0.169 ± 0.036)	0.529 (0.090 ± 0.025)
Rowhouse 3	0.620	0.462 (0.158 ± 0.017)	0.480 (0.140 ± 0.026)	0.540 (0.080 ± 0.022)
House 1	0.626	0.471 (0.155 ± 0.017)	0.462 (0.164 ± 0.025)	0.549 (0.077 ± 0.013)
House 2	0.629	0.474 (0.155 ± 0.016)	0.459 (0.170 ± 0.026)	0.524 (0.105 ± 0.016)
Yard	0.615	0.457 (0.158 ± 0.020)	0.466 (0.149 ± 0.033)	0.538 (0.077 ± 0.022)
Apartment 1	0.620	0.464 (0.155 ± 0.015)	0.459 (0.160 ± 0.032)	0.533 (0.087 ± 0.015)
Apartment 2	0.618	0.460 (0.158 ± 0.022)	0.479 (0.139 ± 0.030)	0.537 (0.081 ± 0.018)
Overall	**0.636**	**0.463 (0.174 ± 0.013)**	**0.468 (0.168 ± 0.012)**	**0.545 (0.091 ± 0.006)**

Table 4. The mean SSIM of each dataset for each method along with the improvement of DART over each baseline and 95% (one-sided) confidence intervals calculated using a z-Test with a effective sample size-adjusted standard error. CFAR is the next best baseline in all cases, followed by *Lidar* and *Nearest* which perform similarly depending on the similarlity between the train and test sets.
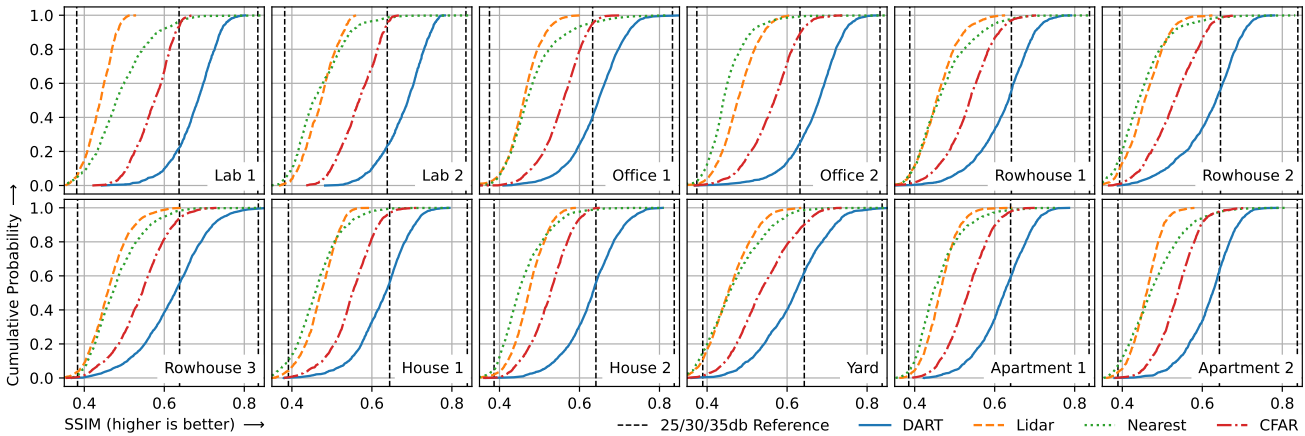

Figure 20. SSIM CDFs for each of the 12 traces (down and to the right is better), along with the mean SSIM of Gaussian noise with 25/30/35db PSNR for reference. DART achieves much higher SSIMs than each baseline on all traces in our dataset, which is roughly equivalent to gaussian noise with a PSNR of 30db.

overfit the training data (e.g. with $\alpha > 1$). This can be seen in a very "cloudy" transmittance map, along with a "trail" of low transmittance that appears to follow the training trajectory. These artifacts are similar to "foggy" artifacts that can be seen in many NeRF techniques.

- Adding the activation function $\alpha = \exp(\min(0, \alpha'))$ with our gradient estimator results in a cleaner map; however, the trail along the training trajectory still remains, since this only addresses some forms of overfitting.

- To prevent DART from overfitting by creating this low-transmittance trail, we set $\alpha = 1$ at approximately empty regions of space (i.e. $\sigma < 0.05$). However, without the gradient estimator, points with an unfavorable initialization may "saturate" $\min(0, \alpha')$ and become stuck at $\alpha = 1$. This can be seen in the pillar in Fig. 19, which is split into multiple parts when the gradient estimator is not used.

C.2. SSIM by Dataset

Table 4 provides a full breakdown of the SSIM of DART and each baseline on each view synthesis dataset; DART dominates each baseline on all datasets. 95% (one-sided) confidence intervals for the difference to DART are calculated using the procedure described in Sec. B.4. We also provide the CDF of the SSIM for each dataset as a distributional view (Fig. 20).

C.3. Additional Tomography Examples

Garden Fig. 22 shows a zoomed view of the *Garden* scene shown in Fig. 7. Metal features along the edge of the raised section are clearly visible in our tomographic images.

Tent Fig. 21 shows a 3D volume rendering of the *Tent* scene shown in Fig. 7 when occupied and unoccupied. In

(a) Empty - Volume   (b) Occupied - Volume   (c) Empty - Section   (d) Occupied - Section
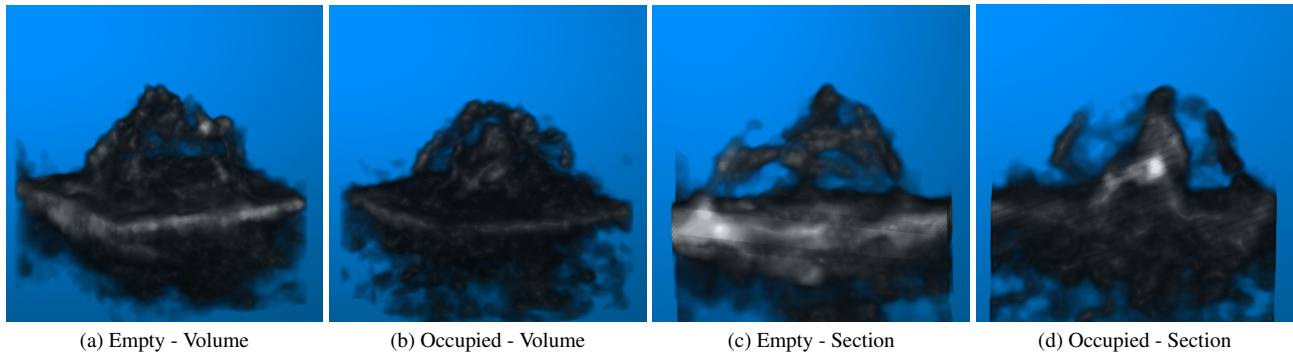
Figure 21. Volume renderings of the tent with and without a person inside, created only using radar sensor readings from outside a tent.
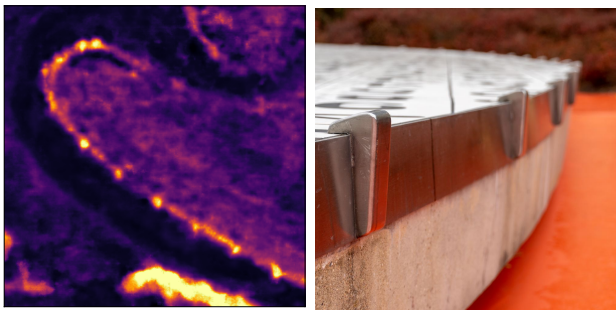

Figure 22. Metal "anti-skate strips" (the bright dots) are clearly visible in our tomography map along with a metal band on the outside of a raised section (the bright outline of the shape).

addition to the tent poles, the tent's occupant is clearly visible; a rough approximation of the occupant's seated pose can also be seen after cutting along a section plane.

Mapping DART can also be used to generate maps of spaces. Fig. 17 shows example learned reflectance maps compared to Lidar occupancy grids; DART is able to learn a coarse map of the room via its radar reflectance, indicating the potential for future localization and mapping algorithms based on DART.

11