High Resolution Point Clouds from mmWave Radar

Akarsh Prabhakara⋆, Tao Jin⋆, Arnav Das†, Gantavya Bhatt†, Lilly Kumari†,
Elahe Soltanaghai#, Jeff Bilmes†, Swarun Kumar⋆ and Anthony Rowe⋆$

⋆Carnegie Mellon University   †University of Washington
#University of Illinois, Urbana-Champaign $Bosch Research

Abstract— This paper explores a machine learning approach on data from a single-chip mmWave radar for generating high resolution point clouds – a key sensing primitive for robotic applications such as mapping, odometry and localization. Unlike lidar and vision-based systems, mmWave radar can operate in harsh environments and see through occlusions like smoke, fog, and dust. Unfortunately, current mmWave processing techniques offer poor spatial resolution compared to lidar point clouds. This paper presents RadarHD, an end-to-end neural network that constructs lidar-like point clouds from low resolution radar input. Enhancing radar images is challenging due to the presence of specular and spurious reflections. Radar data also doesn't map well to traditional image processing techniques due to the signal's sinc-like spreading pattern. We overcome these challenges by training RadarHD on a large volume of raw I/Q radar data paired with lidar point clouds across diverse indoor settings. Our experiments show the ability to generate rich point clouds even in scenes unobserved during training and in the presence of heavy smoke occlusion. Further, RadarHD's point clouds are high-quality enough to work with existing lidar odometry and mapping workflows.

Single chip radar
(stretched out along azimuth for
comparison)

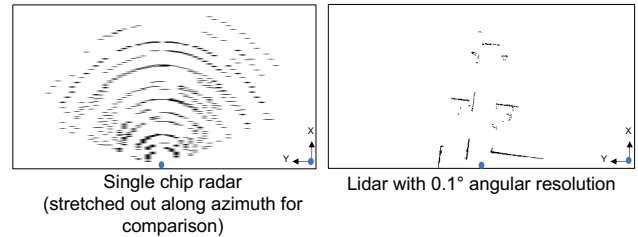Lidar with 0.1° angular resolution

Fig. 1. Single chip radar clearly has a much lower angular resolution than state-of-the-art lidar. Noticeably, radar also shows other spurious artifacts. ● marks the origin and both images show the same scene as perceived by each sensor in a 10x20m area.

I. INTRODUCTION

Lidar is often considered the gold standard in terms of sensors used for mapping, localization, and collision avoidance in robotics. A key enabler is its ability to generate low-noise and high-density point clouds, which can be easily tracked from one position to the next. Despite its ubiquitous use, lidar, just like visible light cameras, fail when used in environments with occlusions – e.g. when robots navigate thick fog, smoke or dust, say for search-and-rescue, disaster recovery and firefighting. Such applications demand sensors that see past occlusions and sense the world in high fidelity.

Radars (the Radio Frequency - RF - equivalent of lidars) show promise given the robustness of RF waves to occlusions [1]. However, due to the longer wavelengths of RF (even at mmWave frequencies), single-chip radars achieve an angular resolution that is two orders of magnitude (hundred times) lower than a lidar. Therefore, a radar resolves point clouds at a much lower resolution than lidar, limiting them to coarse-grained collision avoidance type applications. Higher resolution applications often resort to large mechanical radars, adding bulk and cost. Our goal is to push the resolution limits of a lightweight and compact, single-chip radar, suitable for much more portable platforms (e.g. future small robots, drones, AR/VR headsets, and mobile phones). We specifically seek to exploit the enormous amounts of low-level RF data normally discarded by traditional mmWave processing to dramatically enhance resolution.

Current techniques to improve radar angular resolution include (1) synthetic aperture, which moves radars along specific trajectories precisely [2], [3], [4], [5], [6], and (2) multi-modality (camera or lidar) sensor fusion for better information on angular resolution [7], [8], [9]. However, neither of these are applicable for radars that can move arbitrarily or remain static, and as previously stated, occlusions cause auxiliary sensors like cameras and lidars to fail.

We propose RadarHD, which is a customized end-to-end neural network that generates *lidar-like* point clouds from low resolution radar point clouds. We opt for an end-to-end learning-based pipeline to generate point clouds from radar, allowing for learning features ordinarily missed or thrown away by traditional signal processing pipelines. We show that our generated point clouds are excellent for scene capture, odometry, and mapping, even in smoke-filled environments.

RadarHD is inspired by important recent work on using neural networks on mmWave radar for individual applications: odometry and mapping [10], [11]. However, unlike these systems that target specific higher-level applications, RadarHD targets a broader and more general problem: generating high resolution point cloud data directly from radar I/Q streams that is as good as what a lidar would output (albeit working in lidar-denied scenes). Our approach has two key benefits over per-application end-to-end learning: (1) Point clouds provide an interpretable, easy to understand output. For instance, it's more intuitive to debug and reason about point cloud errors rather than odometry errors, especially when both are output of a machine learning pipeline. (2) High quality point clouds enable a more general representation that can replace lidar in existing point cloud processing pipelines for several tasks beyond just odometry and mapping such as object detection and classification.

In designing RadarHD, we encountered two challenges. *First*, raw radar measurements are impacted by various spurious artifacts – sidelobes from strong reflectors that create sinc-like patterns across azimuth due to its poor azimuth

resolution (see Fig. 1), specular reflections from certain objects, and other processing artifacts [12]. Eliminating these artifacts to recover the true objects is crucial for constructing a dense, accurate point cloud. *Second*, radar images are coarse – meaning that they struggle with resolving sharp environmental features with high angular resolution. In other words, the data from a low resolution radar is quite different from low resolution camera images, where naïve super resolution such as interpolation would give a sensible result.

RadarHD overcomes these challenges by posing a supervised learning problem where large datasets of radar-lidar pairs collected on identical scenes, are used to inform radar to distinguish true objects from noise/artifacts. RadarHD's core contribution is the customization of the entire neural network pipeline – input/output representation, architecture and loss functions – to tackle each of the challenges mentioned above.

To implement RadarHD, we collect a large repository (200k pairs) of raw I/Q radar data from TI AWR1843 mmWave radars paired with lidar point clouds across different indoor and outdoor environments for generalization. RadarHD's evaluation reveals low point-cloud error (24 cm) versus lidar ground-truth and 3.5× superior to traditional radar point-clouds. We also demonstrate the quality of our point clouds with two applications: odometry and mapping, using Google Cartographer [13].

Contributions: RadarHD makes three key contributions:
- Application of a super resolution model for generating lidar-like point clouds from low resolution radar [1].
- A detailed evaluation of the system in new, unseen environments and severe occlusion such as smoke.
- A large repository of raw radar I/Q and lidar pairs along with source code.

II. RELATED WORK

Radar Super Resolution: The mm-level wavelength and the wide bandwidth available at mmWave frequency range provide high ranging accuracy and sensitivity. Combined with the robustness of mmWave radars to different lighting and weather conditions, mmWave radar is a popular option for sensing purposes [14], [15], [16], [17], [18], [19], [20], [21], [22]. In radar, high resolution is usually achieved by using Synthetic Aperture Radar (SAR) [23], [24], [3], [4], [5] or sensor fusion such as integrating radar and camera/lidar [25], [26], [27], [28]. While SAR is used in mobile contexts such as satellite imagery and automotive [29], inaccurate motion information causes errors in the synthesized image [30]. For more portable applications (e.g: future small robots, light-weight UAVs etc.) that we envision, mm-accurate (on the scale of mmWave wavelength) motion information can be expensive to obtain. Moreover, our applications need high resolution images even when the radar is not being moved/temporarily static. More recently, techniques leverage deep learning [6], [31], [32], [33], [34] to perform radar super resolution. [34] uses deep learning to only keep robust points from the range-doppler spectrum. We instead tackle super

resolution and seek to create *lidar-like* point cloud which not only have true radar points but other synthetically generated points that boost the resolution. HawkEye [6] is the closest related work to RadarHD. However, it relies on input data obtained from mechanically scanning the mmWave radar on a large aperture slider to perform SAR [6], [35]. Other works such as [31], [33] just like [6] deal with static radar platform and single object setup. That is, a static radar is looking at a single object (static car/person) whose 3D point cloud is of interest. In such setups where radar platform and object are static, a ground truth can be obtained from a SAR system [31], and low resolution radar could be trained to generate SAR-like output. In contrast, we want to generate high resolution output just from a single-chip radar when (1) radar is static/arbitrarily moving and (2) in real, complicated environments. This calls for rethinking how we collect data, what our ground truth is and design the entire learning pipeline to deal with radar artifacts that show up due to real, complicated environments.

Radar Odometry and Mapping: State-of-the-art radar odometry techniques [36], [37], [38], [39], [40], [41], [42], [43], [44], [45], [46] mainly rely on scan matching, in which spatial features from the radar images are matched against previous scans or a pre-determined map. However, due to spurious reflections and artifacts in radar data, the accuracy of these methods significantly drop for low resolution radars. Follow-up research in this area relies on more complex modeling such as using velocity estimates from radar scans [47], [48], [49], [50], [51], [52], the fusion of radar and IMU data based on Extended Kalman Filtering [53], [54], [41], [55], or fusion of radar and RGB camera [56], [57], [58] to overcome radar limitations. Few other approaches involve end-to-end deep learning approaches for fusing Constant False Alarm Rate (CFAR) radar point clouds and IMU to obtain either odometry [10] or mapping [11], [59] individually. RadarHD, on the other hand, seeks to solve an orthogonal problem: replacing CFAR points by creating higher resolution point clouds. These point clouds are general-purpose and can be fed into multiple lidar processing pipelines such as odometry, mapping, object detection, classification etc.

Radar-Lidar Datasets: Most publicly available radar and lidar datasets capture outdoor environments when mounted on a car [60], [61], [62]. Some of them use non-compact, bulkier mechanically scanning radar [60]. We want a large repository of single-chip radar and lidar pairs in real, complicated environments for robotic applications. While there have been datasets for radar odometry [10], mapping [11] and activity recognition [17], ground truth lidar point cloud is not commonly exposed. More recently, [63], [64] provide single chip radar datasets with monocular and stereo camera largely for automotive purposes. [65] is the closest that comes to the raw radar-lidar dataset that we desire. However, the indoor data samples (∼15k radar-lidar pairs) are only a fraction of the entire dataset. To allow for larger volume of training data, custom design test cases and test robustly, we collect our own dataset.

[1]Demo Link: https://youtu.be/me8ozpgyy0M

Raw Radar Heatmap | CFAR Thresholding at 8dB

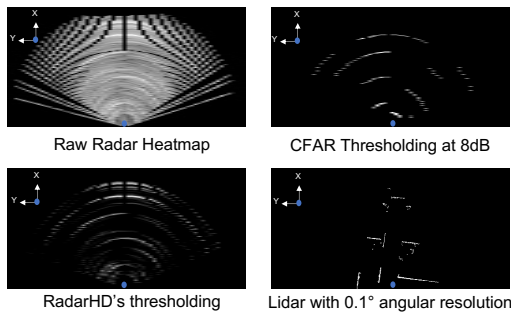RadarHD's thresholding | Lidar with 0.1° angular resolution

Fig. 2. Radar data representation: Raw radar heatmap captures all reflections and radar artifacts but the image is extremely crowded. CFAR thresholding only selects robust strong peaks. RadarHD uses a hybrid capturing a combination of strong/weak peaks and radar artifacts. All radar images are log magnitude normalized to 8 bit image.

III. RadarHD SYSTEM DESIGN

The core objective of RadarHD is to improve the resolution of the radar signals and make them *lidar-like*. At first blush, one may consider simply interpolate the neighborhood and upscaling the radar image (Fig. 2), much like super resolution on camera images [66]. But unlike camera images, low resolution radar images' neighborhoods lack similarity and can often be polluted. One such artifact affecting pixel neighborhoods is the azimuthally spreading patterns that can be seen in Fig. 2. This spreading affects the entire image and is a result of side lobes originating from a few strong reflectors. Tackling this azimuth spreading requires us to go beyond local pixel neighborhood and get a global view of the radar image. On top of this, radar images have "ghost points", i.e. false detections of non-existent objects that also pollute pixel neighborhoods. Thus, we need to first understand what objects are truly present in the real world and weed out these artifacts before upscaling. RadarHD goes beyond classical neighborhood approaches and explores machine learning approaches to obtain global image understanding.

ML Design Choice: Having motivated the need for learning, we now motivate RadarHD's specific choice of ML models. One may simply consider a CNN model that uses convolution layers successively and builds a global view of the input image. However, since we are interested ultimately in the task of super resolution we need to consider global image understanding and upsampling in conjunction. Therefore, RadarHD builds on a U-net based encoder-decoder architecture which is traditionally used on camera image data for segmentation and ports it to address the radar super resolution task. The U-net's general design allows the encoder to understand various radar artifacts and obtain semantically accurate representation of real world objects and the decoder uses this representation to do super resolution.

Design Challenges: The rest of this section describes the key design decisions in RadarHD's ML pipeline that helps U-net to learn effectively. (1) Effectively representing, pre-processing, and inputting radar I/Q data. (2) Designing U-net itself to allow for the elimination of spurious artifacts while preserving data from true objects in the scene. (3) Designing loss functions to ensure that features such as sharp lines in the expected output (see Fig. 2) are preserved.

A. Radar Data Representation

We first consider the problem of representing radar signals as input to the ML architecture. To do so, we briefly describe our radar platform's capabilities and typical sensor output.

Radar Platform: Our mmWave radar [67] provides raw I/Q data streams that can undergo further processing as needed. A typical radar processing pipeline [68] involves a spatial Fourier transform that outputs a 2D heatmap with intensity of the reflected radar signal across range and azimuth. Traditionally, the heatmap gets thresholded to a "point cloud". Constant False Alarm Rate (CFAR) is one such threshold detector that keeps robust objects and eliminate noisy ones. Fig. 2 shows CFAR thresholding applied to radar heatmap.

Choice of Input Representation: At this point, we have a choice – do we feed in raw I/Q inputs directly or should we send processed point cloud data. On one hand, inputting raw I/Q requires the model to understand and learn phase information and to learn some fairly obvious initial steps (e.g. a Fourier Transform). On the other hand, sending in a highly thresholded heatmap may filter out important information that values below the threshold carry, e.g. feeble objects masked by sinc lobes from stronger reflections.

RadarHD instead takes an approach in between these two extremes. Specifically, RadarHD applies a very low threshold to the processed heatmap so that it preserves dominant reflectors, feeble ones and many artifacts. Our objective is to retain a significant portion of the heatmap, including feeble reflectors while leaving it to the ML model to learn and filter out spurious artifacts. RadarHD chooses a threshold such that extremely weak points are omitted, but is still low enough to propagate radar's artifacts, strong and feeble reflectors. For context in Fig. 2, CFAR thresholding has 110 non-zero pixels and RadarHD's thresholding has 1606 non-zero pixels.

Polar vs Cartesian Representation: We note that the thresholded image above is actually in polar format i.e. range-azimuth, but is shown in Fig. 2 in Cartesian format for easy understanding. Given that the output we desire for a point cloud is Cartesian, one may consider the Cartesian representation as the natural choice of representation.

However, radar inherently measures radially and side lobes arising from a strong reflector also spread azimuthally. To capture these radial and azimuthal variations, one would need radial/azimuthal processing. But the primary learning element in convolutional layers in machine learning is a filter that performs 2D correlation across the height and width of the input. To leverage this to our advantage, we choose a polar format so that the filters then naturally traverse along the range and azimuth when they go across height and width, respectively. We thus have thresholded points on the heatmap with range, azimuth, and intensity arranged into an image with range (0-10 m) along rows and azimuth (-90° to 90°) along columns (Fig. 3). The radar images (64×256) are narrower than lidar (512×256) because of the poorer azimuth resolution.
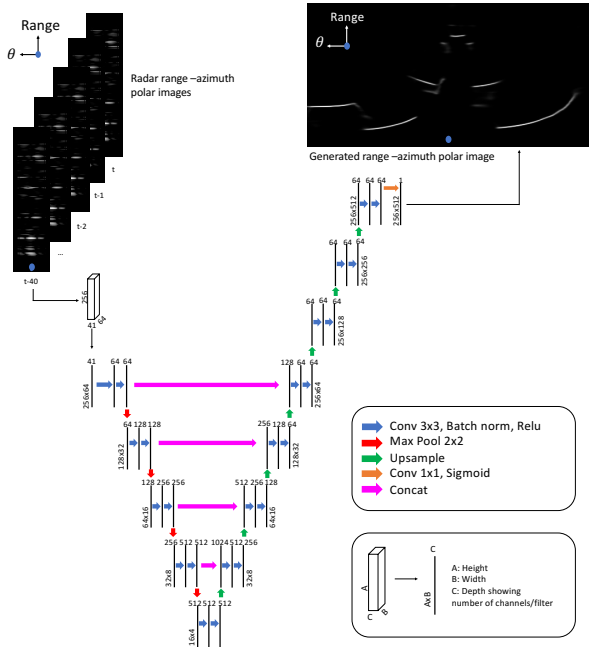
Fig. 3. Assymetric U-net Network architecture: RadarHD adapts an encoder-decoder architecture for radar super resolution. Colored arrows show different ML operations. Black solid lines are 3D inputs and outputs of these operations.

B. Neural network architecture

We choose our base architecture as U-net [69] (see Fig. 3) that allows encoder to denoise and obtain accurate semantic scene understanding and decoder to perform super resolution. U-net captures features at different "resolution levels". From Fig. 3, we can also clearly see that as we move to lower resolutions, the height and width of convolution output decreases (implying capturing more global understanding), and the number of filters increases (to capture richer global understanding). This is crucial to capture the azimuthally spreading artifacts that affects the radar image globally. RadarHD's design has other salient differences from traditional U-nets for the context of radar super resolution.

U-net Asymmetry due to Super Resolution: While base U-net is generally a symmetric network, super resolution problem being fundamentally asymmetric (output image width larger than input), we adapt the base U-net to an asymmetric U-net by adding upsampling and convolutional layers to the decoder to obtain desired resolution (Fig. 3).

Mitigating Specularity: While the input representation allows U-net to capture the azimuthally spreading artifacts, one key challenge that remains is "ghost points". This arises due to specularity [12] of some radar reflections. Specularity is observed when an object, viewed from different orientations, appears and disappears in the radar image. One way to deal with this is to view the scene from multiple viewpoints. The need for multiple viewpoints is further exacerbated when we empirically observed the lack of persistence in the inferred images. That is, when the generated images are viewed as a video, objects would appear and disappear. Thus, it is important to consider using these multi-viewpoint images to tackle specularity and establish a notion of persistence.
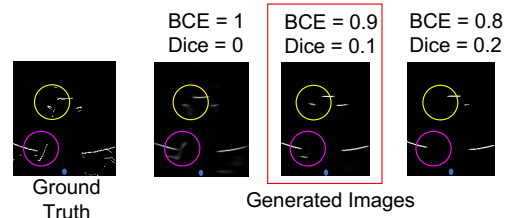

Fig. 4. Qualitative Ablation Study: Dice Loss helps in sharpening lines, but is very aggressive and starts missing out features shown in colored circles. RadarHD chooses a balance between sharpness and accuracy.

Naïvely one could use single frame inference and classical filtering to ensure that objects do not appear and disappear. We perform this filtering through the network by incorporating past radar frames (that offer multiple viewpoints if radar is moving) as input while performing super resolution on the current frame. Our design is to exploit the variable number of input image channels and stack the past frames as input channels to allow for understanding of each radar frame and modeling persistence across frames. We empirically analyze and find 40 past frames (2 seconds history) to be sufficient for enforcing persistence. Even in static cases, using past frames lead to smooth and less jittery output.

C. Neural network training methodology

We carefully select loss functions to preserve features such as sharp lines that appear in lidar images (thin lines of white pixels against black background). We use a combination of various loss functions for achieving different objectives.

Pixel wise loss: To compare two images, one ground-truth label and one output from the network, we first consider the most standard loss function – pixel wise loss. Our ground truth labels are binary lidar images. We compare this binary image against the final sigmoid layer output from the network. We use mean Binary Cross-Entropy (BCE) over all pixels. The objective of this pixel wise loss is to force each pixel to match the expected output. Fig. 4 shows that BCE alone generates an acceptable output, but the lines and boundaries are not as sharp as the ground truth.

Dice loss: To promote crisp and sharp lines in the output image, we draw from Dice loss [70] used in computer vision tasks like boundary detection [71]. For each pixel in ground-truth g_i and network output o_i, Dice loss for N pixels is:

$$D = 1 - \frac{2\sum_{i=1}^{N} o_i g_i}{\sum_{i=1}^{N} o_i^2 + \sum_{i=1}^{N} g_i^2}$$

Here, the numerator finds the loss pixel wise and is maximized when both o and g are identical. The denominator keeps a global view of total number of points that are 1. The loss promotes maximizing the intersection between o and g and penalizes the union of o and g. This forces the network to output 1 exactly where the ground truth is 1, while remaining 0 where the ground truth is 0. This enables a sharper and crisper prediction than pixel wise cross entropy. We analyze and find trade-off between BCE and Dice loss (Fig. 4). More Dice loss leads to eliminating certain important features.
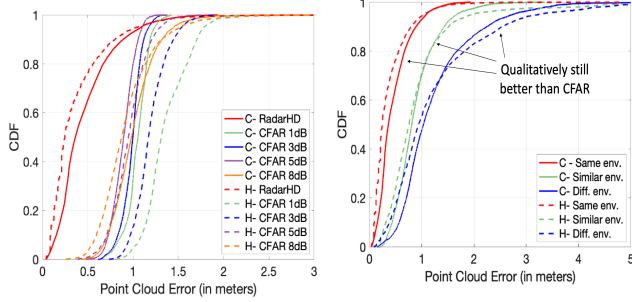
4138

Fig. 5. RadarHD point cloud error.
C- Chamfer H- Mod. Hausdorff

Fig. 6. Contrasting RadarHD error in varying environments.
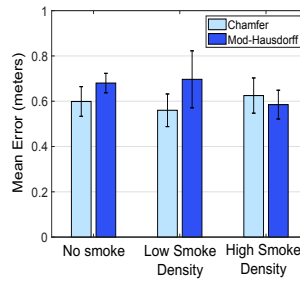C- Chamfer H- Mod. Hausdorff

Fig. 7. RadarHD is unaffected in smoke.

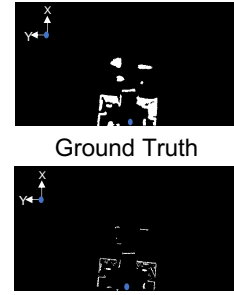Thresholded RadarHD Point Cloud

Ground Truth

Fig. 8. RadarHD thresholds generated images to compare with lidar. ● marks the origin and images show a 10x20m area.
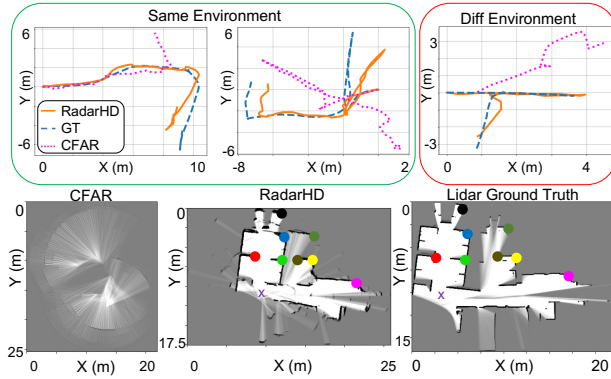

Fig. 9. Odometry and Mapping Examples. Colored points in the maps show physically similar *keypoints*. x is the start of trajectory.

IV. IMPLEMENTATION

System Hardware: RadarHD was implemented using TI mmWave radar AWR1843, a state-of-the-art single-chip radar with a theoretical range resolution of 3.75 cm and azimuth resolution of $15°$. RadarHD's objective is to improve this azimuth resolution. We use the AWR1843 together with DCA1000EVM to collect raw I/Q samples.

Testbed: Our testbed consists of radar, lidar for ground truth and camera for debugging - all time-synced. Our testbed is mounted on a mobile testing rig. Our entire data repository consists of about 200k radar I/Q - lidar pairs collected across a total area of 5147 m^2 which we believe will be extremely useful to the research community.

Ground Truth: We use Ouster OS 0 - 64 beam mechanical scanning lidar for our ground truth. The lidar is configured to work at $0.35°$ azimuth resolution. We only use the forward-facing lidar points for super resolution and we also restrict the lidar's elevation FoV to be within +/-30cm.

Baselines: We use Constant False Alarm Rate (CFAR) based thresholding with different thresholds as our baselines. Today, CFAR is widely used for collecting radar point clouds. CFAR is an ideal baseline as: it is not machine learning based, relies purely on radar signals (no IMU) and works when radar is either static/mobile (unlike SAR). We specifically implement Cell-Averaging CFAR [72].

Train and Test Data: We train our system in a large office space that includes furniture, electronics, walls, conference rooms, and cubicles, with a total area of 839 m^2. All of our tests are performed on unseen data in (1) *same environment* with different trajectories collected across different days, (2) *similar environment* in different office spaces with different cubicle structures, and (3) *different environment*, which is totally new and includes building lobbies and outdoor environments. Across all our data we use 28 trajectories for training (561 meters long with 22784 image pairs in total) and 39 diverse trajectories for testing (714 meters long with 36779 image pairs in total) to allow for testing robustly.

V. RESULTS

A. Point cloud Comparison

Method: After training on rich office space environment, we run RadarHD on all the diverse test samples. To compare against lidar point cloud, we first convert the threshold the range-azimuth output image to obtain a list of points with their (x,y) location. We then compare point cloud error using two popularly used point cloud similarity metrics [73]: (1) Chamfer distance [74]: finds the nearest neighbor for each point in one point cloud to the other, and takes the mean of all these distances to get an error for each point cloud pair. (2) Modified Hausdorff Distance [75]: which also finds the nearest neighbors and obtains the median neighbor distance.

Comparison to baseline: Here, we show our performance in the floor-wide office environment on 19 different trajectories against different CFAR thresholds. This includes 18k radar-lidar point cloud pairs, each over a 10x20 meter area.

As seen in Fig. 5, we obtain a 0.24 m modified-Hausdorff median error and 0.36 m Chamfer median error. CFAR, on the other hand, varies depending on the threshold. A low threshold like 1dB threshold creates point cloud 5x denser than lidar, while a high threshold like 8dB would just have 10% of the number of points captured using lidar. Despite varying density levels across these extremes, none of them have any structural similarity to the ground truth lidar point cloud. So from 1dB to 8dB, as the threshold increases and density decreases, both point cloud error metric Cumulative

ATE (meter)	CFAR		RadarHD		
	3dB	8dB	Same env.	Similar env.	Diff env
Mean	5.02	3.21	1.03	1.30	1.28
RMSE	5.74	3.98	1.35	1.52	1.44
Median	5.70	2.42	0.80	1.22	1.21
Std	2.79	2.35	0.87	0.80	0.68

Map Error (meter)	RadarHD		
	Same env.	Similar env.	Diff env.
Mean	2.03	2.61	3.17
RMSE	2.70	3.58	3.87
Median	1.32	0.91	2.02
Std	1.78	2.44	2.22

Fig. 10. Odometry (Absolute Trajectory Error - ATE) and Mapping quantitative results.

Distribution Functions (CDFs) shift to the left. However, because the highest threshold generates a very sparse point cloud and does not resemble ground truth, the CDFs do not shift any more to the left with an increase in the threshold. In fact, beyond 8dB, the number of points was less than 1% that of lidar. Our generated point cloud not only improves upon CFAR by $3.5\times$ (mod-Hausdorff) and $2.7\times$ (Chamfer), but is also structurally more similar to the ground truth as seen in Fig. 8, unlike Fig. 1.

Generalization: We compare the performance of point cloud generation along these 19 diverse trajectories against those obtained in *similar* and *different* environments having trained only on *same* environments. We use 7.5k point cloud pairs, each for *similar* and *different* environments.

Fig. 6 shows the change in performance in new environments. We see a median error of 0.75 m (mod-Hausdorff) and 0.8 m (Chamfer) for *similar* environments and 0.94 m (mod-Hausdorff) and 1.03 m (Chamfer) for *different* environments. This tells us that these point clouds generated are not quite as accurate as when tested in *same* environment. At first glance, we observe that these medians are similar to the medians for some CFAR CDFs in Fig. 5. However, we would like to point out three important reasons why RadarHD's point clouds are still superior. First, the CFAR CDFs start on the x-axis at 0.4 m; in contrast, even for *similar* and *different* environments, we see that the CDFs start at 0.08 m. This shows that there is a significant fraction of the point clouds that is accurately inferred by our system. Second, because both Chamfer distance and modified Hausdorff distance have a nearest neighbor point association, structural similarity is not entirely captured [76]. However, we can qualitatively see from Fig. 8 that our system indeed generates meaningful points. Third, to quantitatively show the impact of improved accuracy, we compare RadarHD against CFAR in two key applications - odometry and mapping in Sec. V-B.

Smoke: To study the impact of occlusion, we build a smoke chamber around the testing rig and drop smoke pellets in it to create dense smoke. The radar is kept static and same scene is captured without smoke and varying levels of smoke. We use onboard camera to judge the intensity of smoke.

We notice that even with 1 smoke pellet, that generates 500 cubic feet/min, lidar does not receive any points. However, the collected radar signals in smoke are almost identical to

that without smoke even for the densest smoke we could create using 4 pellets. As the collected radar signal remains the same, we expect similar performance as without smoke. Fig. 7 validates this by showing that RadarHD's performance doesn't degrade, up to densities we could create.

B. Odometry and Mapping Comparison

Method: Using the high quality point clouds generated by RadarHD, we next show 2 downstream tasks that RadarHD will enable in scenarios where lidars fail: odometry and mapping. Since our points are lidar-like, we evaluate this by feeding our point clouds, without adding any other sensor (e.g IMU), into existing lidar SLAM frameworks such as Google Cartographer [13]. We obtain the 3-DoF pose in 2D (translation (x,y) and rotation) and map from Cartographer.

Odometry: We evaluate odometry against lidar and benchmark CFAR point clouds using Absolute Trajectory Error - ATE (see Fig. 10). In all cases, including *different* environments, the odometry accuracy of RadarHD outperforms that of CFAR regardless of threshold. Qualitatively, one can clearly see the difference between RadarHD odometry and CFAR odometry in Fig. 9. RadarHD achieves performance comparable to 0.8 m reported in radar+IMU pipelines dedicated for odometry in past work [10].

Mapping: We benchmark the mapping performance by identifying *keypoints* that point to the same physical feature in the real world, such as corners of a room, and then calculate the Euclidean distance error of corresponding *keypoints* between RadarHD and ground truth. Fig. 9 shows a qualitative comparison of a map generated from one trajectory. It is clear that CFAR does not provide any meaningful features to extract *keypoints* while RadarHD achieves a structurally similar map compared to lidar. Fig. 10 shows the Euclidean distance error between *keypoints* across different environments. Good performance on odometry/mapping is possible only because of artifact-free, meaningful point clouds generated by RadarHD. RadarHD also allows for visual debugging of point clouds in case of poor odometry/mapping performance.

VI. CONCLUSION AND FUTURE WORK

RadarHD creates a lidar-like high resolution point cloud from low resolution single-chip mmWave radar input for use in robotic applications where lidar fails. RadarHD designs a machine learning pipeline for this task and overcomes the challenges arising from radar artifacts by choosing design parameters. We show our rich point cloud in a variety of scenes - completely new environments and in occlusions such as smoke. We collect a large dataset of radar-lidar raw data pairs, which is useful for other perception tasks. In the future, we hope to solve other challenges in enabling RadarHD to be an invaluable asset in situations where lidar fails. This includes moving beyond 2D and generating 3D point clouds, tackling highly dynamic scenes, and dealing with 3-dimensional mobile platforms (e.g. UAVs).

REFERENCES

[1] J. W. Starr and B. Lattimer, "Evaluation of navigation sensors in fire smoke environments," *Fire Technology*, vol. 50, no. 6, pp. 1459–1481, 2014.

[2] C. M. Watts, P. Lancaster, A. Pedross-Engel, J. R. Smith, and M. S. Reynolds, "2d and 3d millimeter-wave synthetic aperture radar imaging on a pr2 platform," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4304–4310.

[3] M. E. Yanik and M. Torlak, "Near-field mimo-sar millimeter-wave imaging with sparsely sampled aperture data," *Ieee Access*, vol. 7, pp. 31 801–31 819, 2019.

[4] A. Prabhakara, V. Singh, S. Kumar, and A. Rowe, "Osprey: a mmwave approach to tire wear sensing," in *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, 2020, pp. 28–41.

[5] K. Qian, Z. He, and X. Zhang, "3d point cloud generation with millimeter-wave radar," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 4, no. 4, dec 2020. [Online]. Available: https://doi.org/10.1145/3432221

[6] J. Guan, S. Madani, S. Jog, S. Gupta, and H. Hassanieh, "Through fog high-resolution imaging using millimeter wave radar," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 464–11 473.

[7] R. Nabati and H. Qi, "Centerfusion: Center-based radar and camera fusion for 3d object detection," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 1527–1536.

[8] Y. Long, D. Morris, X. Liu, M. Castro, P. Chakravarty, and P. Narayanan, "Full-velocity radar returns by radar-camera fusion," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 16 198–16 207.

[9] V. Lekic and Z. Babic, "Automotive radar and camera fusion using generative adversarial networks," *Computer Vision and Image Understanding*, vol. 184, pp. 1–8, 2019.

[10] C. X. Lu, M. R. U. Saputra, P. Zhao, Y. Almalioglu, P. P. de Gusmao, C. Chen, K. Sun, N. Trigoni, and A. Markham, "milliego: single-chip mmwave radar aided egomotion estimation via deep sensor fusion," in *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, 2020, pp. 109–122.

[11] C. X. Lu, S. Rosa, P. Zhao, B. Wang, C. Chen, J. A. Stankovic, N. Trigoni, and A. 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*, 2020, pp. 14–27.

[12] K. Bansal, K. Rungta, S. Zhu, and D. Bharadia, "Pointillism: Accurate 3d bounding box estimation with multi-radars," in *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, 2020, pp. 340–353.

[13] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2d lidar slam," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 1271–1278.

[14] F. Guidi, A. Guerra, and D. Dardari, "Personal mobile radars with millimeter-wave massive arrays for indoor mapping," *IEEE Transactions on Mobile Computing*, vol. 15, no. 6, pp. 1471–1484, 2015.

[15] C. X. Lu, S. Rosa, P. Zhao, B. Wang, C. Chen, J. A. Stankovic, N. Trigoni, and A. 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*, 2020, pp. 14–27.

[16] J. Lien, N. Gillian, M. E. Karagozler, P. Amihood, C. Schwesig, E. Olson, H. Raja, and I. Poupyrev, "Soli: Ubiquitous gesture sensing with millimeter wave radar," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, pp. 1–19, 2016.

[17] A. D. Singh, S. S. Sandha, L. Garcia, and M. 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*, 2019, pp. 51–56.

[18] T. Wei and X. Zhang, "mtrack: High-precision passive tracking using millimeter wave radios," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, 2015, pp. 117–129.

[19] M. Zhao, T. Li, M. Abu Alsheikh, Y. Tian, H. Zhao, A. Torralba, and D. Katabi, "Through-wall human pose estimation using radio signals," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7356–7365.

[20] M. Zhao, Y. Tian, H. Zhao, M. A. Alsheikh, T. Li, R. Hristov, Z. Kabelac, D. Katabi, and A. Torralba, "Rf-based 3d skeletons," in *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, 2018, pp. 267–281.

[21] F. Adib, C.-Y. Hsu, H. Mao, D. Katabi, and F. Durand, "Capturing the human figure through a wall," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 6, pp. 1–13, 2015.

[22] C. Wu, F. Zhang, B. Wang, and K. R. Liu, "mmtrack: Passive multi-person localization using commodity millimeter wave radio," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 2400–2409.

[23] B. Mamandipoor, G. Malysa, A. Arbabian, U. Madhow, and K. Noujeim, "60 ghz synthetic aperture radar for short-range imaging: Theory and experiments," in *2014 48th Asilomar Conference on Signals, Systems and Computers*. IEEE, 2014, pp. 553–558.

[24] H. Yamada, T. Kobayashi, Y. Yamaguchi, and Y. Sugiyama, "High-resolution 2d sar imaging by the millimeter-wave automobile radar," in *2017 IEEE Conference on Antenna Measurements & Applications (CAMA)*. IEEE, 2017, pp. 149–150.

[25] Y. Meng, A. Qing, C. Lin, J. Zang, Y. Zhao, and C. Zhang, "Passive millimeter wave imaging system based on helical scanning," *Scientific reports*, vol. 8, no. 1, pp. 1–12, 2018.

[26] M. T. Ghasr, M. J. Horst, M. R. Dvorsky, and R. Zoughi, "Wideband microwave camera for real-time 3-d imaging," *IEEE Transactions on Antennas and Propagation*, vol. 65, no. 1, pp. 258–268, 2016.

[27] R. Appleby and R. N. Anderton, "Millimeter-wave and submillimeter-wave imaging for security and surveillance," *Proceedings of the IEEE*, vol. 95, no. 8, pp. 1683–1690, 2007.

[28] D. M. Sheen, D. L. McMakin, and T. E. Hall, "Near field imaging at microwave and millimeter wave frequencies," in *2007 IEEE/MTT-S International Microwave Symposium*. IEEE, 2007, pp. 1693–1696.

[29] M. Mostajabi, C. M. Wang, D. Ranjan, and G. 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 Workshops*, 2020, pp. 100–101.

[30] G. Fornaro, G. Franceschetti, and S. Perna, "Motion compensation errors: Effects on the accuracy of airborne sar images," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1338–1352, 2005.

[31] S. Fang and S. Nirjon, "Superrf: Enhanced 3d rf representation using stationary low-cost mmwave radar," in *International Conference on Embedded Wireless Systems and Networks (EWSN)...*, vol. 2020. NIH Public Access, 2020, p. 120.

[32] A. Geiss and J. C. Hardin, "Radar super resolution using a deep convolutional neural network," *Journal of Atmospheric and Oceanic Technology*, vol. 37, no. 12, pp. 2197–2207, 2020.

[33] Y. Sun, H. Zhang, Z. Huang, and B. Liu, "Deeppoint: A deep learning model for 3d reconstruction in point clouds via mmwave radar," *arXiv preprint arXiv:2109.09188*, 2021.

[34] Y. Cheng, J. Su, M. Jiang, and Y. Liu, "A novel radar point cloud generation method for robot environment perception," *IEEE Transactions on Robotics*, 2022.

[35] J. Guan, S. Madani, S. Jog, S. Gupta, and H. Hassanieh, "Supplementary materials for through fog high resolution imaging using millimeter wave radar," 2020.

[36] M. Rapp, M. Barjenbruch, K. Dietmayer, M. Hahn, and J. Dickmann, "A fast probabilistic ego-motion estimation framework for radar," in *2015 European Conference on Mobile Robots (ECMR)*. IEEE, 2015, pp. 1–6.

[37] R. Aldera, D. De Martini, M. Gadd, and P. Newman, "Fast radar motion estimation with a learnt focus of attention using weak supervision," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 1190–1196.

[38] ——, "What could go wrong? introspective radar odometry in challenging environments," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 2835–2842.

[39] S. H. Cen and P. Newman, "Precise ego-motion estimation with millimeter-wave radar under diverse and challenging conditions," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 6045–6052.

[40] F. Schuster, C. G. Keller, M. Rapp, M. Haueis, and C. Curio, "Landmark based radar slam using graph optimization," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2016, pp. 2559–2564.

[41] Y. Almalioglu, M. Turan, C. X. Lu, N. Trigoni, and A. Markham, "Milli-rio: Ego-motion estimation with low-cost millimetre-wave radar," *IEEE Sensors Journal*, vol. 21, no. 3, pp. 3314–3323, 2020.

[42] D. Vivet, P. Checchin, and R. Chapuis, "Localization and mapping using only a rotating fmcw radar sensor," *Sensors*, vol. 13, no. 4, pp. 4527–4552, 2013.

[43] F. Schuster, M. Wörner, C. G. Keller, M. Haueis, and C. Curio, "Robust localization based on radar signal clustering," in *2016 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2016, pp. 839–844.

[44] A. F. Scannapieco, A. Renga, G. Fasano, and A. Moccia, "Experimental analysis of radar odometry by commercial ultralight radar sensor for miniaturized uas," *Journal of Intelligent & Robotic Systems*, vol. 90, no. 3, pp. 485–503, 2018.

[45] J. Callmer, D. Törnqvist, F. Gustafsson, H. Svensson, and P. Carlbom, "Radar slam using visual features," *EURASIP Journal on Advances in Signal Processing*, vol. 2011, no. 1, pp. 1–11, 2011.

[46] E. B. Quist and R. W. Beard, "Radar odometry on fixed-wing small unmanned aircraft," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 1, pp. 396–410, 2016.

[47] J. L. Barron, R. E. Mercer, X. Chen, and P. Joe, "3d velocity from 3d doppler radial velocity," *International Journal of Imaging Systems and Technology*, vol. 15, no. 3, pp. 189–198, 2005.

[48] D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer, "Instantaneous ego-motion estimation using doppler radar," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, 2013, pp. 869–874.

[49] ——, "Instantaneous ego-motion estimation using multiple doppler radars," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 1592–1597.

[50] C. Doer and G. F. Trommer, "Radar inertial odometry with online calibration," in *2020 European Navigation Conference (ENC)*. IEEE, 2020, pp. 1–10.

[51] Y. S. Park, Y.-S. Shin, J. Kim, and A. Kim, "3d ego-motion estimation using low-cost mmwave radars via radar velocity factor for pose-graph slam," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7691–7698, 2021.

[52] C. Doer and G. Trommer, "x-rio: Radar inertial odometry with multiple radar sensors and yaw aiding," *Gyroscopy and Navigation*, vol. 12, no. 4, pp. 329–339, 2021.

[53] A. Kramer, C. Stahoviak, A. Santamaria-Navarro, A.-A. Agha-Mohammadi, and C. Heckman, "Radar-inertial ego-velocity estimation for visually degraded environments," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 5739–5746.

[54] A. Scannapieco, A. Renga, G. Fasano, and A. Moccia, "Ultralight radar for small and micro-uav navigation," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 42, p. 333, 2017.

[55] C. Doer and G. F. Trommer, "An ekf based approach to radar inertial odometry," in *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. IEEE, 2020, pp. 152–159.

[56] D. Murdoch, "Augmentation of visual odometry using radar," Master's thesis, University of Waterloo, 2021.

[57] M. Mostafa, S. Zahran, A. Moussa, N. El-Sheimy, and A. Sesay, "Radar and visual odometry integrated system aided navigation for uavs in gnss denied environment," *Sensors*, vol. 18, no. 9, p. 2776, 2018.

[58] A. Ouaknine, A. Newson, J. Rebut, F. Tupin, and P. Perez, "Carrada dataset: Camera and automotive radar with range-angle-doppler annotations," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 5068–5075.

[59] R. Xu, W. Dong, A. Sharma, and M. Kaess, "Learned depth estimation of 3d imaging radar for indoor mapping," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 13 260–13 267.

[60] D. Barnes, M. Gadd, P. Murcutt, P. Newman, and I. Posner, "The oxford radar robotcar dataset: A radar extension to the oxford robotcar dataset," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Paris, 2020. [Online]. Available: https://arxiv.org/abs/1909.01300

[61] M. Meyer and G. Kuschk, "Automotive radar dataset for deep learning based 3d object detection," in *2019 16th european radar conference (EuRAD)*. IEEE, 2019, pp. 129–132.

[62] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 621–11 631.

[63] A. Ouaknine, A. Newson, J. Rebut, F. Tupin, and P. Pérez, "Carrada dataset: Camera and automotive radar with range- angle- doppler annotations," in *2020 25th International Conference on Pattern Recognition (ICPR)*, 2021, pp. 5068–5075.

[64] T. Y. Lim, S. A. Markowitz, and M. Do, "Radical: A synchronized fmcw radar, depth, imu and rgb camera data dataset with low-level fmcw radar signals (ros bag format)," 2021. [Online]. Available: https://doi.org/10.13012/B2IDB-3289560_V1

[65] A. Kramer, K. Harlow, C. Williams, and C. Heckman, "Coloradar: The direct 3d millimeter wave radar dataset," *The International Journal of Robotics Research*, vol. 41, no. 4, pp. 351–360, 2022.

[66] P. Smith, "Bilinear interpolation of digital images," *Ultramicroscopy*, vol. 6, no. 2, pp. 201–204, 1981.

[67] "Awr1843boost," "https://www.ti.com/tool/AWR1843BOOST", 2022.

[68] C. Iovescu and S. Rao, "The fundamentals of millimeter wave sensors," *Texas Instruments*, pp. 1–8, 2017.

[69] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[70] A. Carass, S. Roy, A. Gherman, J. C. Reinhold, A. Jesson, T. Arbel, O. Maier, H. Handels, M. Ghafoorian, B. Platel, *et al.*, "Evaluating white matter lesion segmentations with refined sørensen-dice analysis," *Scientific reports*, vol. 10, no. 1, p. 8242, 2020.

[71] R. Deng, C. Shen, S. Liu, H. Wang, and X. Liu, "Learning to predict crisp boundaries," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 562–578.

[72] M. A. Richards, *Fundamentals of radar signal processing*. Tata McGraw-Hill Education, 2005.

[73] P. Contributors, "Pdal point data abstraction library," Aug. 2022. [Online]. Available: https://doi.org/10.5281/zenodo.2616780

[74] A. Bell, B. Chambers, and H. Butler, "Point data abstraction library," https://pdal.io/apps/chamfer.html, 2022.

[75] ——, "Point data abstraction library," "https://pdal.io/apps/hausdorff.html", 2022.

[76] T. Wu, L. Pan, J. Zhang, T. Wang, Z. Liu, and D. Lin, "Density-aware chamfer distance as a comprehensive metric for point cloud completion," *arXiv preprint arXiv:2111.12702*, 2021.